

**CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS
AVANZADOS DEL INSTITUTO POLITÉCNICO NACIONAL**

UNIDAD ZACATENCO

DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

SECCIÓN DE BIOELECTRÓNICA

**“Procesamiento digital de señales EEG provocadas por
la imaginación de movimiento de los miembros
superiores utilizando análisis no lineal”**

TESIS

Que presenta

Ing. Jesús Alejandro Montalvo Aguilar

Para obtener el grado de

MAESTRO EN CIENCIAS

EN LA ESPECIALIDAD DE

Ingeniería Eléctrica

Director de Tesis: Dr. David Elías Viñas

Dedicatorias

Para mi familia:

A mis padres Melesio Montalvo Elías y Lilia Aguilar Valdez por todo el apoyo y confianza que me han tenido siempre.

A mis hermanos Carlos Alberto Montalvo Aguilar, Iván Melesio Montalvo Aguilar y Lilia Ineri Montalvo Aguilar por su apoyo y compañía.

A mi nueva Familia Fátima López Murillo por su gran apoyo incondicional y a Mini.

Gracias por todo.

Agradecimientos

Expresó mi agradecimiento al pueblo de México, que a través del Consejo Nacional de Ciencia y Tecnología (CONACYT) me apoyo con la ayuda económica para solventar mis estudios de Maestría en estos últimos años.

Agradezco al Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional (CINVESTAV-IPN) por permitirme realizar mis estudios de Maestría en esta Institución.

Al Dr. David Elías Viñas por darme un lugar en su laboratorio para poder realizar mi proyecto de tesis, así como agradecerle por sus consejos, su amabilidad para brindar su conocimiento, apoyo, su gran confianza y amabilidad tanto como Doctor y persona.

A mis compañeros y técnicos del laboratorio 12 por hacer de mi estancia dentro de este, un momento grato y feliz, así como agradecerles por brindarme su apoyo, confianza y conocimiento; un gusto conocerlos Alfonso, Alex, Berno, Emma, Emilio, Esteban, Isis, Jenny, Jessica y Neto.

A mis compañeros de generación por toda su apoyo, confianza y momentos gratos que vivimos en esta etapa de nuestra vida muchas gracias a Fátima, Manu, Tony, Fer, Efrén, Azul, Lara, Melissa, Ever, José, Isma, Luis Karina, Roció, Pedro, Roy y Dani, suerte a todos.

A todos mis amigos que me apoyaron desde que llegue al DF dentro y fuera de ámbito escolar muchas gracias a Álvaro, Artemio, Esmeralda.

Y mi toda familia por su apoyo incondicional siempre, creer y motivarme desde joven a todos ustedes.

GRACIAS

Resumen

El enfoque más prometedor de los sistemas Interfaz Cerebro Computadora (BCI); es ayudar en la rehabilitación de personas con discapacidad motora utilizando la interpretación de la señales EEG durante la fase de imaginación motora para el control de una órtesis o prótesis; y poder brindar movilidad a personas con este tipo de discapacidad. Según datos del INEGI [1], en el año 2010 se registró que 3, 345,994(2.9% de la población total) personas tiene algún tipo de discapacidad motora en el México, las cuales muchas de ellas no pueden tener acceso a las nuevas tecnologías de rehabilitación y no pueden integrarse a las fuerzas laborales del país.

Motivado por los datos anteriores, se presenta la tesis que se basa principalmente en el procesamiento de las señales cerebrales por medio del Análisis Fractal y su clasificación por medio de la inteligencia artificial. El proyecto consistirá en el desarrollo de un programa en Matlab y una tarjeta de adquisición de datos de la Compañía *Measurement Computing*, el cuál analizara las señales EEG en base a la imaginación motora (ritmos μ y β), para procesarlas y poder clasificarlas, y así poder identificar que movimiento se desea realizar; el cual puede ser abrir o cerrar las manos (derecha e izquierda), o quedarse estático esto para poder controlar el sistema Interfaz Cerebro-Computadora (BCI).

Para poder determinar el movimiento; primero separamos la señales de interés; en este caso los electrodos necesarios que se usan en la imaginación motora(IM) que son C3 y C4, después procedemos a extraer las características de la señal por medio de la dimensión fractal de la señal adquirida con diferentes algoritmos(Katz, Higuchi, PSD, Exponente de Hurst, DFA), para después poderla clasificar por medio de un algoritmo de inteligencia artificial como Redes Neuronales Artificiales (RNA) y Maquina de Soporte de Vectores(SVM).

Abstract

The most promising approach for the Brain Computer-Interface (BCI) systems; it is to assist in the habilitation of people with motor disabilities using the interpretation of the EEG signals during motor imagery phase for the control of an orthosis and prosthesis hand; and it can provide mobility for people with this kind of disabilities. According to INEGI [1], in 2010 was recorded that 3,345,994 (2.9%) people have some kind of motor disabilities in the country, which a lot of them may not have access to the new rehabilitation technologies and they cannot integrate the labor forces in the country.

Based on it, the thesis is presented based mainly in the EEG signal processing through Fractal Analysis and their classification base on Artificial Intelligence. The Project involves the development of a Matlab program and using a data acquisition board from Measurement Computing, which it will analyze the EEG signals based on motor imagery task (μ and β rhythms), processing and then it can classify them, so it can be identify which movement is desired; these moves can be stay static, open or close the hands (right or left). This is in order to control a Brain Computer-Interface (BCI) system.

In order to determine the move; first, the signal is separated by to get the interest part; in this case the electrodes that were needed to be used in imagination motor (IM), which are placed on C3 and C4, then proceed to features extraction of the signals through the Fractal Dimension of the signal that was acquired by different algorithms (Katz, Higuchi, PSD, Hurst's Exponent, DFA); finally in order to classify through an Artificial Intelligence algorithm that these will be Neural Network (NN) and Support Vector machine (SVM).

Índice General

Dedicatorias	i
Agradecimientos	ii
Resumen	iii
Abstract	iv

Capítulo 1

1. Introducción	1
1.1. Planteamiento del problema.....	1
1.2. Objetivos	3
1.2.1. Objetivo general.....	3
1.2.2. Objetivos específicos.....	4
1.3. Estructura de la tesis.....	4

Capítulo 2

2. Antecedentes y Estado del Arte	6
2.1. Reseña histórica	6
2.2. Organización anatómica y funcional de la corteza cerebral	8
2.2.1. Origen de las señales cerebrales	10
2.2.2. Imaginación Motora	12
2.2.3. Ritmos μ y β	13
2.2.4. El cerebro, un sistema dinámico complejo	14
2.3. Electroencefalografía (EEG)	16
2.3.1. EEG como serie de tiempo.....	16
2.4. Sistema BCI	17
2.5. Análisis de sistemas no Lineal a señales EEG	20
2.5.1. Teoría del Caos y Fractales.....	20
2.5.1.1. Geometría Fractal.....	22
2.5.1.2. Dimensión Fractal.....	24

2.5.2.	Redes Neuronales	26
2.5.2.1.	Introducción.....	26
2.5.2.2.	Características de las Redes Neuronales Artificiales	27
2.5.2.3.	Redes Neuronales Artificiales.....	28
2.5.2.4.	Investigación al día de hoy	31
2.5.2.5.	Aplicaciones de las Redes Neuronales.....	32
2.5.3.	Maquina de Soporte de Vectores	32

Capítulo 3

3.	Diseño y desarrollo.....	35
3.1.	Solución Propuesta	35
3.2.	Software Propuesto.....	36
3.2.1.	Adquisición de las señales EEG.....	36
3.2.1.1.	Material.....	36
3.2.1.2.	Protocolo para la Adquisición de Datos	37
3.2.1.3.	Imaginación motora.....	43
3.2.2.	Obtención de los ritmos Mu y Beta	44
3.2.2.1.	Filtro Butterworth	45
3.2.3.	Estimación del Exponente de Hurst y la Dimensión Fractal	47
3.2.3.1.	El Exponente de Hurst.....	48
3.2.3.2.	Algoritmo de Higuchi.....	50
3.2.3.3.	Análisis de Fluctuación sin Tendencia (DFA).....	51
3.2.3.4.	Potencia de la Densidad Espectral	53
3.2.4.	Clasificación por Redes Neuronales Artificiales (RNA).....	55
3.2.4.1.	<i>Levenberg-Marquardt</i> con Propagación hacia atrás	55
3.2.4.2.	Gradiente Descendente de impulso y con propagación hacia atrás adaptativo	57
3.2.5.	Clasificación por Maquina de Soporte de Vectores (SVM)	58
3.2.5.1.	El kernel Polinómico	58
3.2.5.2.	El Kernel Gaussiano.....	61
3.2.6.	Simulación de la adquisición de datos en tiempo real	64

Capítulo 4

4. Resultados y discusiones	65
4.1. Pruebas.....	65
4.2. Resultados	66
4.3. Discusión.....	72

Capítulo 5

5. Conclusiones y Trabajo a Futuro.....	75
5.1. Conclusiones.....	75
5.2. Trabajo a Futuro.....	76

REFERENCIAS.....	78
-------------------------	-----------

Anexo A.....	82
---------------------	-----------

Artículos y congresos obtenidos con este trabajo

Anexo B.....	83
---------------------	-----------

Código de Matlab

Capítulo 1

1. Introducción

1.1. Planteamiento del problema

Según el Informe mundial sobre la discapacidad (2011) publicado por la Organización Mundial de la Salud (OMS); se estima que el 15% de la población mundial vive con algún tipo de discapacidad (aproximadamente 1,000 millones de personas) [1]. Las poblaciones más vulnerables son aquellos en los países de ingresos más bajos (países en desarrollo). En América Latina existen alrededor de 85 millones de personas con algún tipo de discapacidad, mientras que en México según datos del Instituto Nacional de Estadística y Geografía (INEGI), en el año 2010 se registro que 3,345,994 (2,9%) de la población tiene algún tipo de discapacidad motora, las cuales muchas de ellas no pueden tener acceso a las nuevas tecnologías de rehabilitación y no pueden integrarse a las fuerzas laborales del país, así como también el mundo no está adaptando para sus necesidades como es el caso en el transporte público, edificios públicos. Algunas de las causas de este tipo de padecimiento pueden ser por accidente, enfermedad neuromuscular, por nacimiento, por edad avanzada [2].

El enfoque más prometedor de los sistemas Interfaz Cerebro Computadora (BCI); es ayudar en la rehabilitación de personas con discapacidad motora; ya que establecen una interacción en tiempo real entre el usuario y su entorno utilizando únicamente las señales cerebrales (como las EEG) en lugar de acciones motoras.

Desde hace poco más de tres décadas ha causado gran interés el desarrollo de sistemas BCI debido a su potencial para construir un canal de comunicación natural para el hombre con la máquina, dado que el BCI está basado en medir la actividad cerebral, se puede pensar en una interfaz que traduce la información cognitiva y emocional del usuario directamente en ordenes hacia dispositivos. El

reto científico y tecnológico está en decodificar esta información de todo el volumen de datos.

La aplicación que desde un inicio ha causado mayor interés es el apoyo de pacientes con una lesión medular grave o una enfermedad neuromuscular crónica que interrumpe el flujo de información neural motora hacia las extremidades del cuerpo, tales como la esclerosis lateral amiotrófica, infarto cerebral, parálisis cerebral o lesiones en la columna vertebral. La potencialidad de las BCI es ofrecer un canal de comunicación directa desde el sistema nervioso central hacia dispositivos, evitando el uso de los caminos neuronales que no pueden ser utilizados normalmente, lo que abre un enorme abanico de aplicaciones que puede proporcionar una mayor independencia a los pacientes [3].

Los sistemas BCI no han logrado promover grandes inversiones por parte de la industria para su investigación y desarrollo debido a su limitado sector de usuarios. Pero nuevos enfoques de las tecnologías BCI, hacia un campo más amplio de usuarios principalmente el área militar y de videojuegos, han conseguido atraer la atención de empresas para invertir en dicha investigación. Este nuevo planteamiento de las BCI anima el desarrollo de tecnología más amigable con el usuario, más compacta, con mayor capacidad de procesamiento, más barata y más accesible al público en general, lo que resulta beneficioso para la investigación con objetivo médico de las BCI.

El reto tecnológico que involucra el desarrollo de BCIs es enorme dado que involucra un conjunto multidisciplinario de conocimientos como son: la intersección de las neurociencias, la ingeniería biomédica, técnicas de procesamiento y análisis de señales, ingeniería de control y diversas ingenierías dependiendo de la aplicación, como la mecatrónica o robótica.

Actualmente la tecnología BCI aún no ha logrado cumplir sus propias

expectativas, es decir, las BCIs no son capaces de deducir la intención del usuario. En el presente desarrollo el usuario no se concentra directamente en la tarea que lleva a cabo, sino que se diseña alguna tarea que utilice procesos cognitivos conocidos y que se saben medir, para que de su identificación se deduzca la intención del usuario. Por eso mismo, las BCIs actuales no son interfaces naturales para los usuarios y el desarrollo de las mismas es el último objetivo de todas las investigaciones que se están llevando a cabo en este campo.

Un aspecto es que todas las personas tenemos una actividad cerebral diferente en los dominios temporales, frecuenciales y espaciales, además de que debido a que la señal EEG es una señal caótica, no estacionario lo cual nos indica que no se puede medir por método lineales. Las herramientas que se utilizan han de tener la capacidad de aprender a identificar los patrones de cada persona en particular basándose en fenómenos de auto calibración. Por esto, esta tesis busca comparar este tipo de procesamiento en base a la dimensión fractal por diferentes métodos para determinar cuál es la intención del usuario y poderla clasificar por medio de Redes Neuronales y Máquina de Soporte de Vectores.

1.2. Objetivos

1.2.1. Objetivo general

El objetivo principal del proyecto es el diseño y desarrollo de un software el cual realice el procesamiento de las señales cerebrales obtenidas previamente con un Electroencefalograma para poder encontrar un patrón en las señales cerebrales de diferentes personas; en donde las señales a analizar se dividen en 3 acciones que se buscan diferenciar:

- a) Permanecer estático.

- b) Imaginación del movimiento de la mano izquierda.
- c) Imaginación del movimiento de la mano derecha.

Se propone encontrar las características temporales de la señal EEG registrada en sujetos sanos al realizar tareas de movimiento y de imaginación de movimiento de la mano con los ojos abiertos. Con este registro se propone probar nuestros algoritmos de clasificación y entrenamiento.

1.2.2. Objetivos específicos

- Diseño y desarrollo de un software el cual puede extraer las características de las señales EEG por medio de la dimensión fractal.
- Encontrar un patrón de las señales en el dominio del tiempo por medio de la dimensión fractal.
- Encontrar un patrón de las señales en el dominio de la frecuencia por medio de la dimensión fractal.
- Comparación de diferentes algoritmos para ver cual tiene mejor exactitud y mejor velocidad.
- Diseño y desarrollo de un software el cual pueda clasificar las características de las señales EEG por medio de un clasificador no lineal.
- Evaluación de los algoritmos para la extracción de características con diferentes métodos de clasificación.

1.3. Estructura de la tesis

La tesis está dividida en cinco capítulos. En el primer capítulo de introducción se explica en el planteamiento del problema como ayuda un sistema BCI a las personas con enfermedades neuromusculares y sus retos en la actualidad; además de los objetivos que se abarcan en la tesis. Dentro del segundo capítulo se exponen los antecedentes y estado del arte, donde se explica

que es un sistema BCI, su historia, así como los últimos hallazgos de esta tecnología, también se explica cómo se genera las señales EEG, las diferentes partes del cerebro y su función, que es la imaginación motora, el porqué se considera al cerebro un sistema caótico y no estacionario, así como algunos métodos matemáticos no lineales para poder procesar las señales EEG. En el tercer capítulo se da a conocer la solución propuesta donde se explican la aplicación de los filtros en el sistema, los diferentes métodos para obtener la dimensión fractal de una señal; así como una breve reseña de que son las Redes Neuronales y las Maquinas de Soporte de Vectores. Dentro del cuarto capítulo de Resultados y discusiones se explicara cómo se obtuvieron las señales EEG analizar; así como se realizaron las pruebas de este software tiempo real, los resultados obtenidos; En el quinto capítulo de Conclusiones y trabajo a futuro en donde se describirán las evaluaciones obtenidas de estas, aparte de ver si es un buen método para implementarse; además se trata una discusión de los resultados comparándolos con otros métodos similares visto en publicaciones para poder obtener una conclusión acerca de nuestro sistema en comparación con los ya existente. Así como un planteamiento de trabajos que se pueden derivar a partir de este proyecto de tesis.

Capítulo 2

2. Antecedentes y Estado del Arte

2.1. Reseña histórica

Desde 1929, Hans Berger (Neurólogo Alemán), considerado como el padre de la electroencefalografía, utilizó esta técnica por primera vez, esta ha sido usada por médicos y científicos para investigar el funcionamiento del cerebro. Así como se propuso la posibilidad de usar el Electroencefalograma (EEG) para descifrar intenciones, de forma en que una persona pudiera controlar determinados dispositivos a partir del análisis de su señal EEG, debida a su actividad cerebral.

Así, se define *Brain-Computer Interface* (BCI) [4], o Interfaz Cerebro-Computadora, como un sistema de comunicación que monitorea la actividad cerebral y traduce determinadas características, de acuerdo a las intenciones del usuario en comandos de control de uno o varios dispositivos.

El primer antecedente que se tiene que se logró establecer la comunicación directa entre el cerebro y el mundo exterior fue el de Dewan, quien en 1967 experimentó la utilización del ritmo alfa para producir caracteres en código Morse [5]. Este tipo de interfaz es simple, robusta, y normalmente no requiere entrenamiento específico para su utilización, por esa razón aun hoy son objeto importante de investigación a pesar de sus limitadas aplicaciones y avances.

En 1988 L.A. *Farwell and Donchin* de la *University of Illinois* implementaron un sistema a partir del potencial P300, el cual se manifiesta como una onda positiva en el potencial medido del EEG en la zona parietal a una latencia de aproximadamente 300ms tras la aparición del estímulo deseado, dentro de una

secuencia de estímulos aleatorios. En este caso, los estímulos corresponden a la presentación de letras en la pantalla. El usuario observa la pantalla y se concentra en la letra que quiere escribir. El sistema presenta las letras en orden aleatorio mientras monitoriza el EEG, y cuando se detecta el potencial P300, el sistema obtiene la letra en la que el usuario está pensando [6].

Alrededor de 1994, los grupos de *Jonathan R. Wolpaw* y *Dennis McFarland* del *Wadsworth Center for Laboratories and Researchen New York, USA* y el de *Gert de Pfurtscheller* en la *University of Technology of Graz, Austria* comenzaron a trabajar con BCI. Estos grupos históricamente se dedicaron al estudio de los ritmos cerebrales, fundamentalmente los relacionados con actividades motoras y posteriormente comenzaron a aplicarlo en la implementación de BCI [7,8].

Los trabajos de estos laboratorios fueron muy significativos para darle un empuje de crecimiento a la tecnología BCI y actualmente son de los grupos de investigación más importantes en el mundo ya que cuenta con una gran diversidad de proyectos en su experiencia. Los grupos más importantes en la investigación de trabajos BCI son: “*Laboratory of Brain-Computer Interfaces*” en Graz, Austria (como se mencionó anteriormente) [10], el “*Bernstein Center of Computational Neuroscience*” en Berlín, Alemania [11] y el “*Institute of Neural Engineering*”, de la Universidad Tsinghua en Beijing, China [12].

Un aspecto fundamental para entender y controlar los sistemas BCI; es conocer la generación del proceso de la actividad cerebral y conocer la modalidad para medirla. En este capítulo se explicará la organización anatómica y funcional de la corteza cerebral especialmente en la imaginación de movimientos. Se explicará porque se considera al cerebro como un sistema complejo o caótico; el funcionamiento de la tecnología de registro EEG, las partes esenciales de un sistema BCI, así como su evolución y las características extraíbles de la señal EEG y los tipos de procesamiento más comunes que se están utilizando en la

actualidad como son el Análisis no lineal.

2.2. Organización anatómica y funcional de la corteza cerebral

Los procesos intrínsecos de la vida corporal están regulados en las partes subcorticales del cerebro. En cambio, en la corteza cerebral se procesan y almacena la información proveniente de los sensores del propio organismo. La corteza cerebral se divide en dos partes más o menos simétricas denominadas hemisferios. Cada hemisferio puede dividirse principalmente en cuatro lóbulos que son: frontal, temporal, occipital y parietal (como se puede observar en la figura 2.1), a los cuales se les atribuyen capacidades funcionales específicas del sistema nervioso [13].

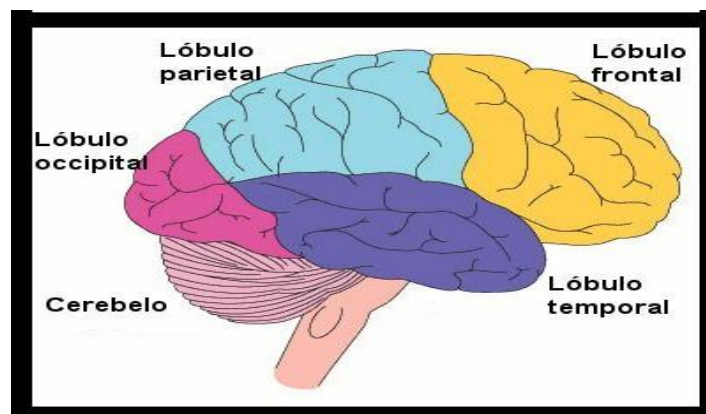


Fig. 2.1 Estructura del cerebro

En los lóbulos occipitales se encuentran la corteza visual, la cual participa en la integración de la respuesta visual, permiten analizar los patrones y combinar imágenes con otros estímulos sensoriales.

Los lóbulos parietales se asocian a estímulos sensoriales como la temperatura, el dolor y el tacto. Cuenta con regiones asociativas que intervienen en la interpretación de la información sensorial y participa en la selección de

palabras para expresar pensamientos y sentimientos.

En el lóbulo frontal se relaciona con las funciones motrices primarias (corteza motor primaria). Aunque también en este lóbulo se localizan varias áreas asociativas que integran procesos intelectuales de alto nivel, como son la concentración, la solución de problemas complejos, la conducta emocional.

En el lóbulo temporal se encuentran las funciones relacionadas con la interpretación de estímulos sensoriales complejos como la memoria, la comprensión del lenguaje, la lectura, la memoria visual y musical.

Los estudios de Penfield en humanos y de Woolsey en animales pusieron en manifiesto que la estimulación eléctrica de la corteza motora primaria producía movimientos de diferentes partes en el lado contra lateral del cuerpo. Existen regiones específicas del córtex motor que influyen sobre la actividad de grupos de músculos específicos (figura 2.2). Es importante destacar que la organización somatotópica del córtex motor no es fija, sino que se puede alterar durante un aprendizaje motor, o después de una lesión [14].

En el mapa motor representado en la corteza motora la musculatura utilizada en tareas que requieren un control muy fino (como la de la cara o manos) ocupa mucho más espacio que la representación de la musculatura que requiere un control motor relativamente menos preciso, como se observa en el homúnculo de la figura 2.2. Estudios indican que hay músculos que están representados en más de un lugar de la corteza.

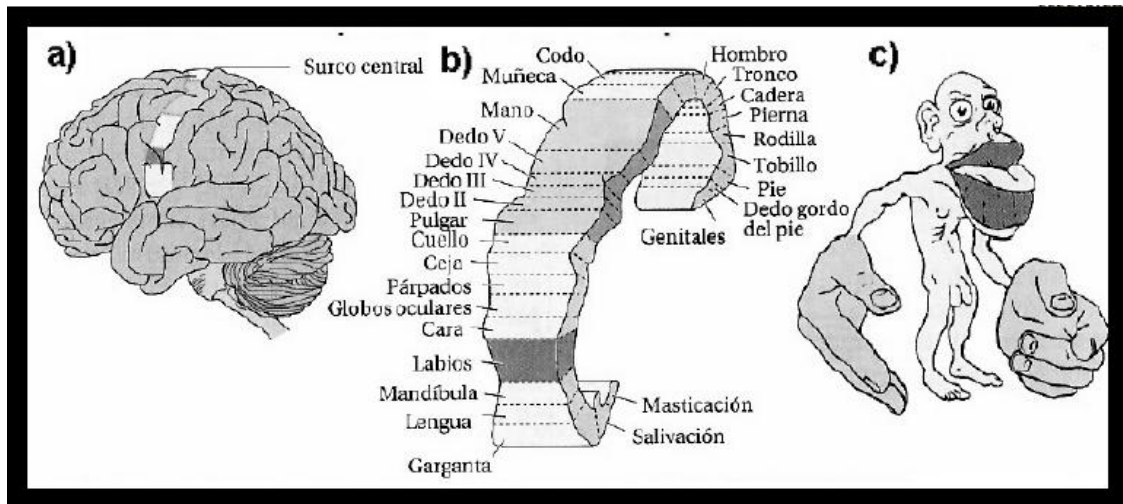


Fig 2.2 Organización somatotópica de la corteza motora primaria humana.

2.2.1. Origen de las señales cerebrales

Los registros electrográficos de la superficie del cerebro y de la superficie exterior de la cabeza revelan que el cerebro tiene una actividad eléctrica continua. Tanto la intensidad como los patrones de esta actividad, están determinados en gran medida por el nivel de excitación neuronal resultante de estados fisiológicos como pueden ser: pensar, mover una parte del cuerpo, el sueño, enfermedades como la Epilepsia, Alzheimer o algunas psicosis.

Las ondulaciones de los potenciales eléctricos del cerebro se denominan ondas cerebrales y a su registro se le conoce como electroencefalograma (EEG), y la técnica con la que se realiza el registro es electroencefalografía.

Según Welding [2007]; un EEG es “una señal estacionaria con eventos transitorios y rupturas en su dinámica, más o menos abruptas”.

Las ondas o señales cerebrales se caracterizan por su frecuencia y amplitud particulares. La intensidad de las ondas sobre la superficie del cuero

cabelludo varía entre 0 y 200 μV y su frecuencia varía de 0.5 hasta 50 Hz. Cuando los registros se realizan directamente sobre la superficie de la corteza cerebral (EEG intracraneal), se puede presentar amplitudes más altas [15]. Las ondas cerebrales se clasifican en 4 bandas de frecuencia que corresponde a distintos tipos de ritmos cerebrales, de acuerdo a la actividad en el cerebro del sujeto, tal como se muestra en la Tabla 2.1.


Designación	Frecuencia	Región cerebral	Ocurrencia	Ejemplo de su registro
Ondas Beta (β)	13-30 Hz	Parietal y frontal.	Adultos en estado de vigilia.	
Ondas Alfa (α)	8-13 Hz	Occipital.	Relajación y concentración mental.	
Ondas Teta (θ)	4-8 Hz	Regiones temporales.	Niños y adultos dormidos.	
Ondas Delta (δ)	0.5-3 Hz	Varias regiones.	Bebes. Estados profundos de sueño.	

Tabla 2.1 Principales Ritmos Cerebrales

También se encuentra el ritmo μ ; es el menos frecuente de los ritmos en un registro normal, estando tan solo en un 10% de los individuos normales; localizado en las regiones centrales. Se identifica por su morfología típica en “arcos” y por ser suprimido si se mueve la extremidad superior contra lateral. Está vinculado a los sistemas sensorial motor, de forma contra lateral. Sin relación con lo visual ni con la actividad mental [15].

La mayor parte del tiempo las ondas cerebrales son irregulares y en el EEG no se encuentra un patrón general, mientras que en otros momentos aparecen patrones característicos, algunos respecto anomalías específicas del cerebro y otros presentes también en personas sanas.

Es importante destacar que la descarga de una sola neurona no se puede registrar desde la superficie de la cabeza, debido a que su magnitud es demasiado pequeña. Para registrar la actividad eléctrica cerebral, se requiere de una descarga sincrónica de millones de neuronas; de esta manera se suman los potenciales de las neuronas individuales y esta magnitud se puede registrar a través del cráneo. Por lo tanto, la intensidad de las ondas cerebrales registradas en el cuero cabelludo depende principalmente del número de neuronas que descarguen sincronizadas entre sí, no del nivel total de actividad eléctrica del cerebro. Incluso algunas señales nerviosas fuertes y asíncronas se pueden anular en el registro debido a su polaridad opuesta. [Martínez, 1998].

2.2.2. Imaginación Motora

Basado en estudios que describen a detalle el proceso de imaginar un movimiento, [16] [17] [18]; se puede decir que la imaginación motora es una parte del fenómeno relativo a la preparación de un movimiento. Es decir; imaginar un movimiento genera el mismo proceso mental e incluso físico, a excepción de que el movimiento queda bloqueado.

Como se demuestra en trabajos citados anteriormente, estas afirmaciones se comprueban observando diversos efectos físicos del hecho de estar imaginando un movimiento. En primer lugar durante el tiempo que se realiza la simulación, la excitabilidad de la espina dorsal se incrementa, aumentando los reflejos mono sinápticos espinales, tal y como lo haría si realmente el movimiento se fuera a ejecutar. Además, la respiración y el pulso se aceleran de forma similar tanto cuando se está pensando en la acción como cuando se está realizando el ejercicio. Por último, la duración de la acción real y su simulación son similares. Se puede concluir; que la imaginación de un movimiento puede ser una tarea mental claramente diferenciable del estado de reposo.

De hecho, como se comprobó en algunos de los experimentos realizados

[19], para identificar el estado mental en el que se encuentra el sujeto de estudio bastará con tomar aquellas señales obtenidas a partir de los electrodos colocados en las cercanías del área motora del cerebro.

Es importante determinar cuál o cuáles serán las tareas mentales que se desean reconocer para poder establecer un canal de comunicación adecuado entre la persona y la computadora; es decir, que tareas serán reconocidas para poderlas traducir a comandos que el sistema puede reconocer. En este caso será la imaginación de abrir o cerrar las manos.

2.2.3. Ritmos μ y β

Las ondas cerebrales asociadas a la actividad motora son los ritmos μ y β que se registran en la corteza sensorial motora. Estos ritmos se pueden observar como un cambio en la energía en el análisis en el dominio de la frecuencia de la señal EEG.

Los ritmos se atenúan cuando se producen eventos que implican que el área cortical involucrada procese o se prepare a procesar información. Este fenómeno se denomina desincronización asociada al evento ERD (*Event Related Desynchronization*). Finalizado el evento que produjo la ERD el ritmo suele reaparecer intensificado. Este fenómeno se denomina Sincronización Relacionada al Evento: ERS (*Event Related Synchronization*).

Los conceptos ERD y ERS solo son significativos si se toma en cuenta el registro del EEG en estado basal, entonces los ritmos ERD o ERS se presentan alrededor del instante de planificación y ejecución del movimiento.

El ritmo μ se manifiesta como una ERD en la banda α (7-12Hz) (de amplitud entre 20-60 μ V) que se inicia alrededor de 2 segundos antes de la ejecución del

movimiento y dura hasta uno o dos segundos posteriores al movimiento. El ritmo β (13-30Hz) (de amplitud entre 5-10 μV) se observa como una ERD en el entorno del instante de la ejecución del movimiento y una ERS post-movimiento que inicia alrededor de 0.8 segundos luego de la ejecución del movimiento y puede durar hasta unos 6 segundos. Puede llegar a tener un valor del 400% del valor de referencia. Si durante la ERS se realiza un nuevo movimiento, durante la preparación y ejecución del mismo esta es bloqueada (ERD). Esto puede observarse con claridad en la Figura 2.3, que corresponde a un ensayo con movimientos relativamente próximos entre sí.

Se ha reportado en diversos trabajos que estos ritmos no solamente existen durante movimientos activos de los sujetos, sino que también mientras los sujetos imaginan realizar algún movimiento [13].

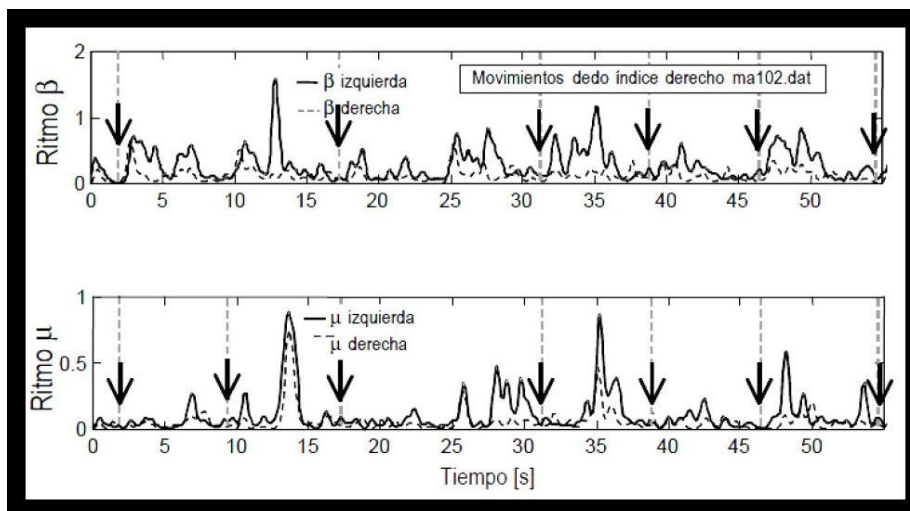


Fig 2.3 Evolución de los ritmos y al efectuar movimientos con el dedo índice derecho.

2.2.4. El cerebro, un sistema dinámico complejo

Los sistemas complejos están compuestos por elementos simples que modifican su estructura de acuerdo al intercambio que reciben del entorno; y de sus propias interacciones, son inestables en su mayoría y dado cualquier

variación de algunos de sus elementos puede ocasionar un cambio global en todo el sistema, se consideran impredecibles.

Una variación provoca el cambio permanente entre un estado de orden y desorden o caos. Por lo tanto, cuando un sistema tiene un comportamiento impredecible, se le llama sistema caótico.

Un sistema dinámico es aquel cuyo estado varía en el tiempo, determinística o estocásticamente, es decir, en el primero, su resultado siempre es el mismo con las mismas condiciones iniciales, mientras que en el segundo está basado en función de probabilidades [Clinton, 2004].

El cerebro es un sistema dinámico complejo; está formado por millones de neuronas, que forman redes y cuyas interacciones posibilitan las distintas funciones cerebrales como la percepción, el aprendizaje, la memoria, los movimientos y la atención. En él se encuentran numerosas variables, muchas de las cuales aún en estos días permanecen sin poder explicarse del todo. Este desconocimiento produce que, al ocurrir anomalías, como en el caso de los padecimientos neurológicos, su evolución en el tiempo sea impredecible.

El funcionamiento general de algunas propiedades del cerebro se ha monitoreado mediante distintos métodos; uno de ellos es el estudio de las señales cerebrales, a partir del cual se ha podido describir la dinámica cerebral en distintos estados de este, demostrando así sus características como sistema complejo, dinámico y no lineal. Así, las señales cerebrales pueden presentar distintos estados diferenciados al estado normal de funcionamiento del cerebro; por una alteración como puede ser una enfermedad, un movimiento o un pensamiento [14].

2.3. Electroencefalografía (EEG)

El primer registro de señal de actividad cerebral desde la superficie cortical se realizó en animales por Richard Caton en 1875 y en personas en 1929 por Hans Berger. En 1925 Nemiski introdujo la palabra electroencefalograma (EEG) para definir el registro de la actividad del encéfalo desde la superficie del cerebro, es decir en el cuero cabelludo.

Para registrar un EEG se colocan electrodos en la superficie del cuero cabelludo, los cuales captan la señal proveniente de la suma de los cambios de potenciales en las neuronas que forman la corteza cerebral y la registra como una señal eléctrica que posteriormente es amplificada y digitalizada para poder registrar. Debido a que los electrodos están separados de la corteza cerebral por el cráneo y las capas cutáneas que la protegen, la señal capturada está deformada y solamente proporciona una idea parcial de la actividad cerebral.

Los cambios de potencial se presentan por los intercambios de iones (Na^+ , Cl^- , K^+ , etc.) que ocurren en la membrana de las neuronas debido a los impulsos, excitatorios o inhibitorios, que llegan a la célula procedentes de otras neuronas, pasando de una a otra a través de cientos de puntos de unión llamados sinapsis. Los electrodos que se colocan sobre el cuero cabelludo registran la suma de las pequeñas corrientes que producen los potenciales de las células más superficiales de la corteza cerebral, las que a su vez llevan información de las neuronas que se encuentran situadas más profundamente. Los cambios de potencial que se registra en el EEG varían de acuerdo a la extensión que cubra un par de electrodos y al tiempo que trascorra durante su registro [20].

2.3.1. EEG como serie de tiempo

Un EEG (electroencefalograma) representa los cambios en el campo eléctrico originados en el cerebro y registra las diferencias de voltaje entre pares

de electrodos en relación a un electrodo de referencia. Estas diferencias se amplifican, se transforman por medio de un convertidor analógico-digital y finalmente se muestran en un monitor, se imprimen o se almacenan con un formato digital, en ocasiones paralelamente a un registro en video de la actividad física de la persona, lo que permite correlacionarla con la señal registrada. Los datos almacenados digitalmente son magnitudes de voltaje, con una tasa de muestreo determinado por el usuario.

Un registro Electroencefalográfico (EEG) es una *serie de tiempo*, es estocástica, no determinística, es decir, no siempre se obtiene la misma actividad con las mismas condiciones iniciales [14].

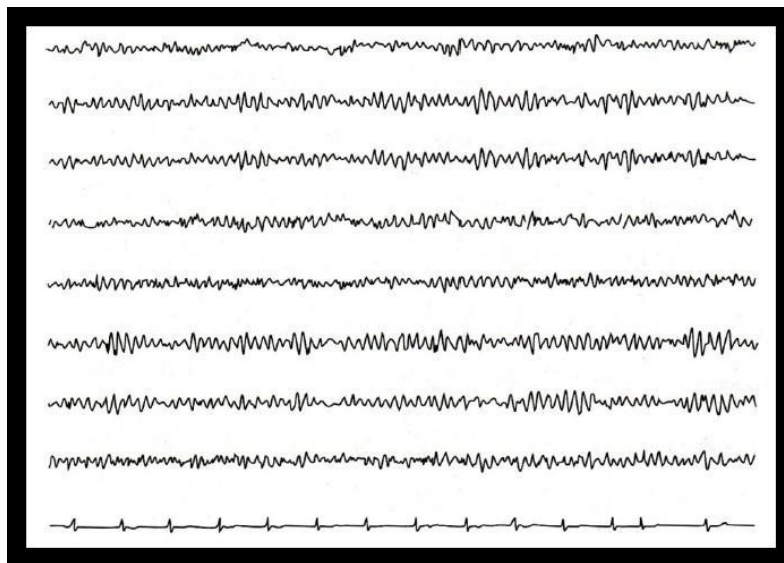


Fig 2.4 EEG es una serie de tiempo estocástica

Fuente: Franco, 2006

2.4. Sistema BCI

Como se mencionó anteriormente los sistemas BCI establecen una interacción en tiempo real entre el usuario y su entorno utilizando únicamente las señales cerebrales (como las EEG) en lugar de acciones motoras.

Los sistemas BCI no han logrado promover grandes inversiones por parte de la industria para su investigación y desarrollo debido a su limitado sector de usuarios. Pero nuevos enfoques de las tecnologías BCI, hacia un campo más amplio de usuarios principalmente el área militar y de videojuegos, han conseguido atraer la atención de empresas para invertir en dicha investigación. Este nuevo planteamiento de las BCI anima el desarrollo de tecnología más amigable con el usuario, más compacta, con mayor capacidad de procesamiento, más barata y más accesible al público en general, lo que resulta beneficioso para la investigación con objetivo médico de las BCI.

Actualmente la tecnología BCI aún no ha logrado cumplir sus propias expectativas, es decir, los BCIs no son capaces de deducir la intención del usuario. En el desarrollo presente el usuario no se concentra directamente en la tarea que lleva a cabo, sino que se diseña alguna tarea que utilice procesos cognitivos conocidos y que se saben medir, para que de su identificación se deduzca la intención del usuario. Por eso mismo, las BCIs actuales no son interfaces naturales para los usuarios y el desarrollo de las mismas es el último objetivo de todas las investigaciones que se están llevando a cabo en este campo.

Algunos de los principales problemas a los que se enfrentan los actuales sistemas BCI son: en primer lugar los artefactos, que es actividad eléctrica presente en el EEG pero que no tiene su origen en el cerebro. El movimiento de los ojos, de la lengua, de los músculos del cuerpo, etc; crean una actividad que puede aparecer en el registro EEG. Su presencia tiene dos dificultades. La primera es que los procesos cerebrales a decodificar pueden verse afectados o incluso eliminados por la actividad de los artefactos; y la segunda es que puede darse el caso en el que las herramientas de procesamiento aprendan a identificar la actividad de los artefactos como el objetivo a decodificar.

En segundo lugar la actividad medida en el EEG tiene una naturaleza no estacionaria, lo que hace que varíe en el tiempo para una misma persona durante el uso de la BCI y entre diferentes sesiones. Esto dificulta enormemente las estrategias de procesamiento de señal, las cuales deben ser capaces de adaptarse en línea durante cada sesión, y recalibrarse antes de empezar una nueva sesión.

El tercer aspecto es que todas las personas tenemos una actividad cerebral diferente en los dominios temporales, frecuenciales y espaciales. Las herramientas que se utilizan han de tener la capacidad de aprender a identificar los patrones de cada persona en particular basándose en fenómenos de auto calibración.

El grueso de investigación que se está realizando en BCIs no invasivas está centrado en alguna de estas tres áreas, con el objetivo de filtrar mejor los artefactos, desarrollar sistemas que se adapten automáticamente durante la ejecución, y en disminuir el tiempo de calibración de las herramientas para adaptarlas a cada individuo en particular.

Actualmente se han diseñado sistemas BCI para detectar el estado de ánimo, Neuro-prótesis en la medula alta de la espina dorsal lastimada, con potenciales de campo local y electro cortical; entre otras [16]. Además de diversos métodos se han aplicado para estos sistemas y poder entender la señal EEG; diferentes tipos de filtros entre los más importantes los “Filtros Laplacianos Espaciales” [21][22]. Para la extracción de características es uno de los pasos más importantes ya que esto define la velocidad y precisión del sistema entre los métodos más importantes actualmente están: la dimensión fractal, el exponente *wavelet*, el exponente de *lyapunov*, potencia de la banda (*Band power*), Densidad espectral de Potencia (*Power Spectral Density*), Parámetros Auto regresivos Adaptativos, características en tiempo y frecuencia (Fourier) entre otras técnicas

de análisis no lineal. Para la clasificación las técnicas más usadas actualmente son Redes Neuronales Artificiales, Bayes Cuadrática, Modelo Oculto de Markov, Análisis de Discriminación Lineal, Maquina de Soporte de Vectores, Logica Difusa, Sistemas ANFIS, *Nearest Neighbor Classifiers* (NNC), *K Nearest Neighbor* (KNN) [23].

2.5. Análisis de sistemas no Lineal a señales EEG

Como se mencionó anteriormente el funcionamiento general de algunas propiedades del cerebro se ha monitoreado mediante distintos métodos; pudiendo conocer que las señales cerebrales, han podido describir la dinámica cerebral en distintos estados de este, demostrando así sus características como sistema complejo, dinámico y no lineal. Debido a esto se ha optado por realizar diversas técnicas para el procesamiento no lineal tanto para la clasificación, extracción de características como son Redes Neuronales Artificiales (RNA), Lógica difusa, Maquinas de Soporte de Vectores, Análisis de Frecuencia (Transformada de Fourier o Análisis Wavelet) entre otras; obteniendo los mejores resultados para la extracción de características con Análisis Wavelet y con la geometría fractal; así como con Maquinas de Soporte de Vectores con múltiples núcleos y RNA para su clasificación.

2.5.1. Teoría del Caos y Fractales

Debido a que no existe una definición universal para caos, se le considera que es el comportamiento aperiódico a largo plazo de un sistema determinístico limitado que presenta dependencia sensible a las condiciones iniciales.

La teoría del caos tuvo sus orígenes en los años 60; Edward Lorenz, un meteorólogo del MIT estudiaba un modelo con ecuaciones diferenciales para predecir los estados del tiempo. Al realizar un nuevo análisis previo, introdujo sin darse cuenta, cantidades con menos datos significativos y descubrió así la dependencia del comportamiento del sistema a las condiciones iniciales (un mínimo cambio produce cambio drásticos en los resultados del sistema) y la imposibilidad de predecir el clima, esto es conocido como el efecto mariposa: la

divergencia entre dos trayectorias inicialmente cercanas en el espacio de fase se va magnificando exponencialmente, lo que termina generando patrones climáticos radicalmente distintos [Clinton 2003].

Por tanto, cuando un sistema dinámico no lineal tiene un comportamiento impredecible, se le llama sistema caótico. También es altamente sensible a los cambios en las condiciones iniciales.

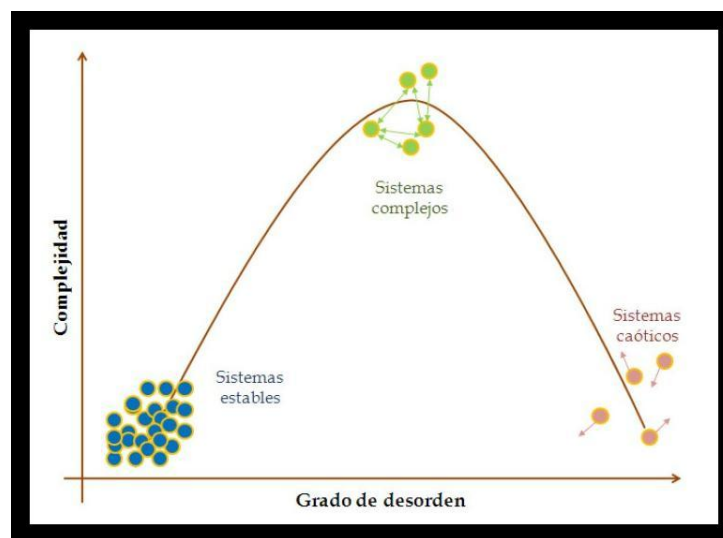


Fig. 2.5. Ubicación de los diferentes tipos de sistemas según su complejidad

El caos estudia el comportamiento cualitativo a largo plazo de un sistema dinámico, principalmente para describir la estabilidad del sistema a través de los atractores: las manifestaciones geométricas del caos, puntos constantes en el tiempo hacia los que tiende a converger el sistema [14].

El atractor de Lorenz es una figura geométrica en forma de mariposa, que para ser contenida necesita más de dos dimensiones y menos de tres (2.06 dimensiones), por lo que es considerado un fractal (Figura 2.6)

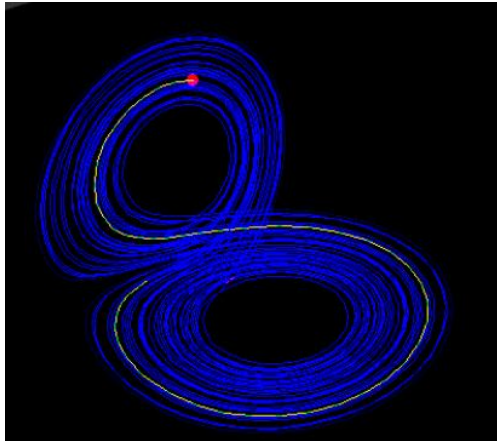


Fig 2.6 Atractor de Lorenz

Fuente: Scientific/Educational Matlab Database

2.5.1.1. Geometría Fractal

Análisis fractal es un nuevo paradigma científico que ha sido usado satisfactoriamente en muchos campos incluidos la biología y Fisiología. La descripción Fractal de una señal EEG puede ser una herramienta útil para la extracción de características y entender su comportamiento.

Benoît Mandelbrot, un matemático polaco, concibió y desarrolló en 1977 una nueva geometría que describe los patrones irregulares y fragmentados encontrados en la naturaleza, la geometría fractal o teoría de fractales: un lenguaje matemático utilizado para caracterizar los fenómenos que presentan invariancia de escala, que es su propiedad principal; el mismo modelo es válido para todas las escalas.

Mientras trabajaba en un laboratorio de investigación en IBM, Mandelbrot realizó uno de los primeros programas gráficos por computadora, utilizando el conjunto de Julia, cuya teoría sobre las iteraciones sobre el plano complejo había estudiado anteriormente. Gaston Julia fue uno de los padres de la teoría de

sistemas dinámicos y el conjunto que lleva su nombre es uno de los fractales más conocidos (Figura 2.7).

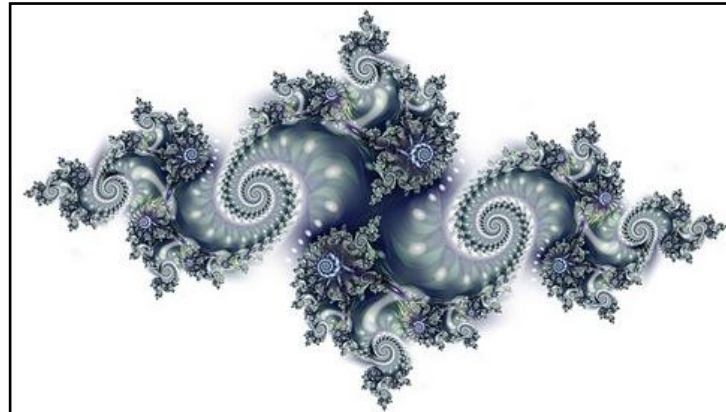


Fig. 2.7 *Conjunto de Julia*

Fuente: Scientific/Educational Matlab Database

Comprendió que muchas de los objetos encontrados en la naturaleza (nubes, árboles, galaxias, montañas, flores, rocas, etc.) presentan formas irregulares que no pueden representarse por medio de la geometría tradicional. A estas formas les llamó fractales.

Más tarde, Mandelbrot formuló el ahora llamado conjunto de Mandelbrot, un cálculo basado en la iteración de números complejos que inicia desde cero. Después de millones de iteraciones mediante un programa computacional, es posible observar la representación de esta generación de números, que se presenta en una figura recursiva auto similar, en una escala infinita (Figura 2.8).

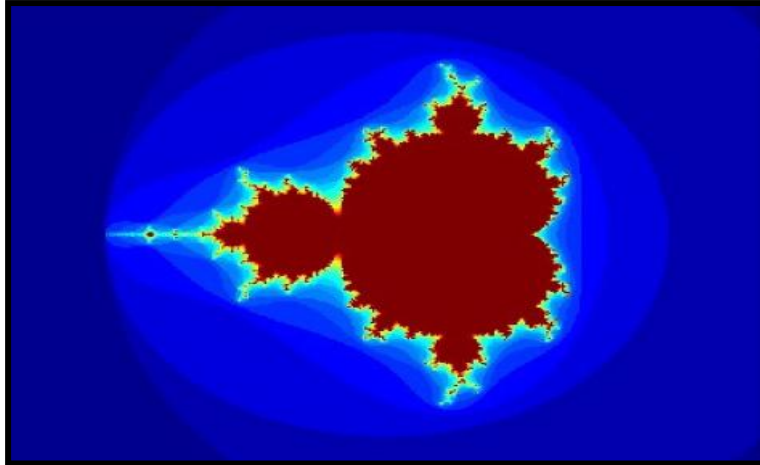


Fig. 2.8 Conjunto de Mandelbrot

Fuente: Scientific/Educational Matlab Database

La palabra fractal, a la vez verbo y adjetivo, acuñado por Mandelbrot, tiene su origen en el Latín *fractus* que significa “irregular” y del verbo *frangere*, “romper en pedazos irregulares” [Mandelbrot 1997]

La Geometría Fractal ha estado en los escritorios de los matemáticos puros desde hace muchos años, que en un principio los catalogaban muchos de estos y otros fenómenos como “monstros” sin solución. Es ahora y después del perfeccionamiento de las computadoras que podemos modelarlos inclusive algunos sistemas más complejos y encontrar soluciones útiles, y esto con la capacidad de procesamiento de una computadora sencilla [14].

2.5.1.2. Dimensión Fractal

Para comprender la fractalidad, es necesario recordar el concepto de dimensiones. La dimensión topológica (DT) se asigna a un número natural a la dimensión de los objetos geométricos que pueden definirse en un espacio dado.

Las dimensiones topológicas conocidas son: 0 si es un punto aislado o un

número finito de puntos, 1 si es una recta o cualquier curva estándar, 2 si es un plano y cualquier otra superficie ordinaria, 3 si es un espacio o un objeto con volumen (Figura 2.9.a). De manera intuitiva se entiende la dimensión de un objeto como el número de parámetros necesarios para definirlos (Figura 2.9.b).

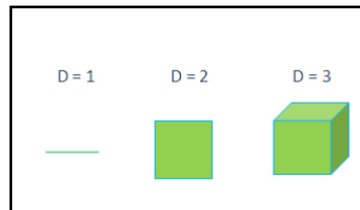


Fig. 2.9.a Dimensiones topológicas conocidas


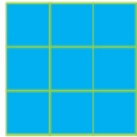

	N	Factor de escala	
	3	3	$3^1 = 3$
	9	3	$3^2 = 9$
	27	3	$3^3 = 27$

Fig. 2.9.b Cálculo de la dimensión topológica

Como aparece en la Figura 8.b, la dimensión euclidiana está dada por un exponente entero. Sin embargo, algunos objetos encontrados en la naturaleza se encuentran en una dimensión no entera: su dimensión es un número decimal mayor que la dimensión topológica.

Bajo la ley de escala $N = s^D$, con s como factor de escala, puede observarse la dimensión de un objeto De acuerdo a esta ley:

$$D = \frac{\log N}{\log s}$$

Félix Hausdorff demostró en 1919 que la dimensión de un objeto podría calcularse al encontrar el factor de escala por el que matemáticamente el objeto se puede reproducir, por lo que para ciertos objetos, la dimensión es una fracción. Este concepto es conocido como la dimensión de Hausdorff-Besicovitch (D) o dimensión fractal y fue expuesta por Hausdorff y reformulada por Besicovitch [Mandelbrot B., 1997].

La dimensión de Hausdorff; nos permite cuantificar la dimensión de cualquier forma geométrica y es una medida de la complejidad y rigurosidad del cuerpo geométrico y nos da una idea de su extensión real en el espacio.

Para calcular la dimensión fractal (dimensión de Hausdorff) se usa la siguiente fórmula [24]:

$$d = \frac{\log N}{\log \frac{L}{l}}$$

2.5.2. Redes Neuronales

2.5.2.1. Introducción

El cerebro humano es el sistema de cálculo más complejo que conoce el hombre. El ordenador y el hombre realizan bien diferentes clases de tareas; así la operación de reconocer el rostro de una persona resulta una tarea relativamente sencilla para el hombre y difícil para el ordenador, mientras que la contabilidad de una empresa es tarea costosa para un experto contable y una sencilla rutina para un ordenador básico.

La capacidad del cerebro humano de pensar, recordar y resolver problemas ha inspirado a muchos científicos intentar o procurar modelar en el ordenador el funcionamiento del cerebro humano.

Los profesionales de diferentes campos como la ingeniería, filosofía, fisiología y psicología han unido sus esfuerzos debido al potencial que ofrece esta tecnología y están encontrando diferentes aplicaciones en sus respectivas áreas.

Un grupo de investigadores ha perseguido la creación de un modelo en el ordenador que iguale o adopte las distintas funciones básicas del cerebro. El resultado ha sido una nueva tecnología llamada Computación Neuronal o también Redes Neuronales Artificiales.

El resurgimiento del interés en esta nueva forma de realizar los cálculos tras dos décadas de olvido se debe al extraordinario avance y éxito tanto en el aspecto teórico como de aplicación que se está obteniendo en estos últimos años [25].

2.5.2.2. Características de las Redes Neuronales Artificiales

Las Redes Neuronales Artificiales, *ANN (Artificial Neural Networks)* están inspiradas en las redes neuronales biológicas del cerebro humano. Están constituidas por elementos que se comportan de forma similar a la neurona biológica en sus funciones más comunes. Estos elementos están organizados de una forma parecida a la que presenta el cerebro humano.

Las *ANN* al margen de "parecerse" al cerebro presentan una serie de características propias del cerebro. Por ejemplo las *ANN* aprenden de la

experiencia, generalizan de ejemplos previos a ejemplos nuevos y abstraen las características principales de una serie de datos.

Aprender: Adquirir el conocimiento de una cosa por medio del estudio, ejercicio o experiencia. Las *ANN* pueden cambiar su comportamiento en función del entorno. Se les muestra un conjunto de entradas y ellas mismas se ajustan para producir unas salidas consistentes.

Generalizar: Extender o ampliar una cosa. Las *ANN* generalizan automáticamente debido a su propia estructura y naturaleza. Estas redes pueden ofrecer, dentro de un margen, respuestas correctas a entradas que presentan pequeñas variaciones debido a los efectos de ruido o distorsión.

Abstraer: Aislar mentalmente o considerar por separado las cualidades de un objeto. Algunas *ANN* son capaces de abstraer la esencia de un conjunto de entradas que aparentemente no presentan aspectos comunes o relativos [25].

2.5.2.3. Redes Neuronales Artificiales

La neurona es la unidad fundamental del sistema nervioso y en particular del cerebro. Cada neurona es una simple unidad procesadora que recibe y combina señales desde y hacia otras neuronas. Si la combinación de entradas es suficientemente fuerte la salida de la neurona se activa. La Figura 2.10 muestra las partes que constituyen una neurona.

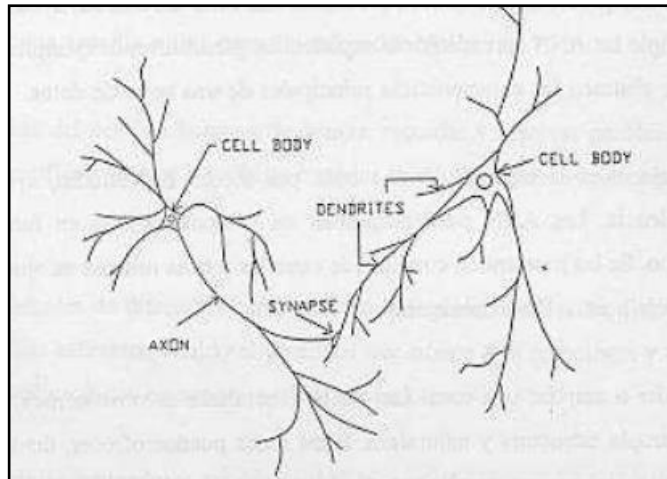


Fig. 2.10. Componentes de una Neurona

El cerebro consiste en uno o varios billones de neuronas densamente interconectadas. El axón (salida) de la neurona se ramifica y está conectada a las dendritas (entradas) de otras neuronas a través de uniones llamadas sinapsis. La eficacia de la sinapsis es modificable durante el proceso de aprendizaje de la red.

En las Redes Neuronales Artificiales, *ANN*, la unidad análoga a la neurona biológica (como en el caso del cerebro humano) es el elemento procesador, *PE* (*process element*). Un elemento procesador tiene varias entradas y las combina, normalmente con una suma básica. La suma de las entradas es modificada por una función de transferencia y el valor de la salida de esta función de transferencia se pasa directamente a la salida del elemento procesador.

La salida del *PE* se puede conectar a las entradas de otras neuronas artificiales (*PE*) mediante conexiones ponderadas correspondientes a la eficacia de la sinapsis de las conexiones neuronales.

La Figura 2.11 representa un elemento procesador de una red neuronal artificial implementada en un ordenador.

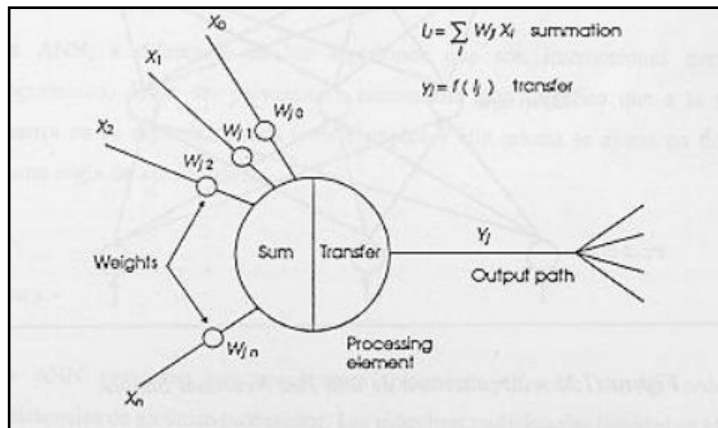


Fig. 2.11. Diagrama de una Neurona Artificial (PE)

Una red neuronal consiste en un conjunto de unidades elementales PE conectadas de una forma concreta. El interés de las ANN no reside solamente en el modelo del elemento PE sino en las formas en que se conectan estos elementos procesadores. Generalmente los elementos PE están organizados en grupos llamados niveles o capas. Una red típica consiste en una secuencia de capas con conexiones entre capas adyacentes consecutivas.

Existen dos capas con conexiones con el mundo exterior. Una capa de entrada, buffer de entrada, donde se presentan los datos a la red, y una capa buffer de salida que mantiene la respuesta de la red a una entrada. El resto de las capas reciben el nombre de capas ocultas. La Figura 2.12 muestra el aspecto de una Red Neuronal Artificial [25].

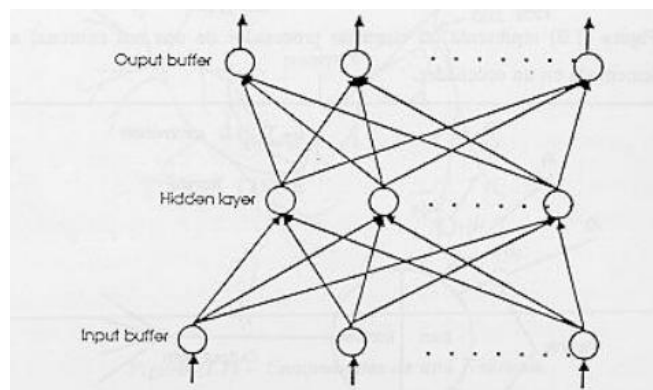


Fig. 2.12. Arquitectura de una Red Neuronal Simple

2.5.2.4. Investigación al día de hoy

Existen muchos grupos con sede en diferentes universidades de todo el mundo que están realizando trabajos de investigación en el área de las redes neuronales artificiales. Cada grupo tiene diferente énfasis y motivación resultado de las colaboraciones con los neurólogos, psicólogos del conocimiento, físicos, programadores y matemáticos. Todos ellos ofrecen nuevos puntos de vista e intuiciones en esta área de la técnica.

Grossberg continua trabajando en compañía de Carpenter en la Universidad de Boston, mientras Teuvo Kohonen está en la Universidad de Helsinki. Uno de los mayores grupos de investigación de los últimos años ha sido el grupo PDP (*Parallel Distributed Processing*) formado por Rumelhart, McClelland y Hinton.

Rumelhart de la Universidad de Stanford es uno de los principales impulsores de la red más utilizada en la mayoría de las aplicaciones actuales, la famosa red neuronal *Backpropagation*. En la Universidad de Carnegie-Mellon, el grupo de investigación a la cabeza con McClelland destaca por el estudio de las posibles aplicaciones de la *Backpropagation*. Y en la Universidad de Toronto, Hinton y Sejnowski han desarrollado una máquina llamada Boltzman que consiste en la red de Hopfield con dos modificaciones significativas.

Bart Kosko ha diseñado una red llamada BAM (*Bidirectional Associate Memory*) basado en la red de Grossberg. Por último indicar la existencia de grandes grupos de investigación como los de California *Institute of Technology*, *Massachusetts Institute of Technology*, *University of California Berkeley* y *University of California San Diego*.

Conviene no olvidar el esfuerzo económico y técnico que están realizando las empresas privadas tanto en Estados Unidos como en Japón y en la Comunidad Económica Europea. Como botón de muestra de las inversiones en estos países basta conocer que sólo en USA se gasta más de 100 millones de dólares al año [25].

2.5.2.5. Aplicaciones de las Redes Neuronales

Las características especiales de los sistemas de computación neuronal permiten que sea utilizada esta nueva técnica de cálculo en una extensa variedad de aplicaciones.

La computación neuronal provee un acercamiento mayor al reconocimiento y percepción humana que los métodos tradicionales de cálculo. Las redes neuronales artificiales presentan resultados razonables en aplicaciones donde las entradas presentan ruido o las entradas están incompletas. Algunas de las áreas de aplicación de las ANN son las siguientes:

- Análisis y procesado de señales
- Control de procesos
- Robótica
- Diagnóstico médicos
- Reconocimiento de Imágenes
- Filtrado de ruido
- Procesado de Lenguaje

2.5.3. Máquina de Soporte de Vectores

Una Máquina de Soporte de Vectores (SVM) usa un hiperplano de discriminación para identificar clases. Sin embargo, para la SVM, la selección del hiperplano es el que maximiza los márgenes, por ejemplo, la distancia del punto más cercano de entrenamiento (Figura 2.13), la maximización de los márgenes es

conocido como el incremento de la capacidad de generalización de los elementos a clasificar. Una SVM usa un parámetro de regularización C que permite el alojamiento de los valores atípicos y permite errores en el conjunto de entrenamiento.

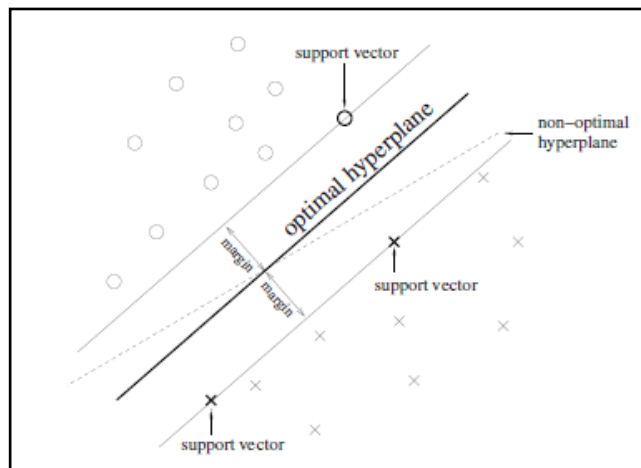


Fig. 2.13. Una SVM encontrara el hiperplano óptimo para la generalización

La SVM permite su clasificación utilizando límite de decisiones lineales; y se conoce como SVM lineal. Esta clasificación ha sido aplicada satisfactoriamente, a un gran número relativamente grande de problemas BCI. Sin embargo, es posible crear límites de decisiones no lineales, con solo un bajo aumento en la complejidad del clasificador; usando el conocido “truco del Núcleo”. Consiste en la cartografía de los datos implícitamente en otro espacio, generalmente de mucho mayor dimensión. Usando una función de kernel “K(x,y)”. El núcleo generalmente usado en investigaciones BCI es el Kernel Gaussiano o el Kernel de la Función de base Radial (RBF).

$$K(x, y) = \exp\left(\frac{-\|x - y\|^2}{2\sigma^2}\right)$$

La correspondiente SVM se conoce como SVM Gaussiana o SVM de base radial, ambos métodos nos arrojan muy buenos resultados para aplicaciones BCI.

Los sistemas LDA, SVM han sido aplicados a problemas BCI multiclases usando una estrategia de OVR.

Las SVM tienen varias ventajas. Actualmente, gracias a la maximización de los márgenes y la regularización de los términos, se sabe que las SVM tiene buenas propiedades de generalización, son insensibles al sobre entrenamiento, y se pueden acomodar gracias a la dimensionalidad. Finalmente, las SVM tiene unos pocos hiper parámetros que se tiene que definir a mano, llamados, el parámetro de regularización "C" y el ancho del RFB " σ "; si se usa un kernel mayor a 2. Estas ventajas se obtienen a expensas de una baja velocidad de ejecución [23].

Capítulo 3

3. Diseño y desarrollo

3.1. Solución Propuesta

Para poder analizar la señal EEG se debe de emplear un análisis no lineal; debido a que el cerebro es un sistema dinámico complejo o caótico y debe tratarse como tal para poder tener resultados satisfactorios.

En este trabajo se pretende hacer el análisis de la señal EEG por medio de técnicas de análisis no lineal como son el Análisis Fractal que nos ayuda a modelar un sistema caótico en base a una serie de tiempo y un clasificador multiclases como los son las Redes Neuronales Artificiales y SVM las cuales nos ayudan a poder procesar y clasificar los datos de estudios.

Para el Análisis Fractal nos basaremos en obtener su dimensión fractal. La dimensión fractal nos permite encontrar que tan caótico es una serie de tiempo y poder saber cómo se comporta con el paso del tiempo. Para obtener su dimensión fractal usaremos los algoritmos de Higuchi, DFA, y PSD; los dos primeros se basan en obtener la dimensión en el dominio del tiempo mientras que el último la obtiene en el dominio de la frecuencia.

Por medio de la Inteligencia Artificial como los son las Redes Neuronales Artificiales y las SVM; se procesara la serie para determinar el movimiento que se desea realice el sistema BCI; el cual puede ser mover la mano derecha, izquierda, o no hacer ningún movimiento.

El análisis fractal junto con la Inteligencia Artificial (IA), nos proporcionan una herramienta computacional útil y de bajo costo para poder caracterizar las

señales aleatorias (en este caso las señales EEG); y poderlas definir las.

3.2. Software Propuesto

El software propuesto para este trabajo se ha desarrollado en forma modular para poder facilitar su desarrollo e ir revisando que va pasando en cada etapa del software; además de facilitar mantenimiento, mejoras y velocidad.

Cada etapa será descrita en los siguientes subsecciones; explicando el porqué se uso, como funciona y que resultado es el esperando en cada módulo.

3.2.1. Adquisición de las señales EEG

3.2.1.1. Material

Las señales EEG utilizadas para este proyecto fueron adquiridas previamente en el Instituto Nacional de Rehabilitación (INR) de la Ciudad de México por medio del equipo de adquisición Ig.EEG32sys (ver Figura 3.1).



Fig. 3.1 Sistema de Adquisición de bioseñales g.EEG32sys

El sistema cuenta con los siguientes dispositivos:

- Electroencefalograma g.TEC en forma de gorro.
- Amplificador g.USBamp de 32 canales.
- Software g.Recorder para adquisición de datos.
- Software g.BSanalyze para análisis en línea
- PC con el software necesario para la adquisición.

Los sistemas nos ayudan a adquirir la señal, configurar el amplificador, visualización y almacenamiento de datos en la PC. Tiene la posibilidad de asignar y registrar disparos “*Triggers*” y marcadores. Se puede usar con electrodos activos y pasivos, el sistema se puede configurar para una adquisición en tiempo real.

Además del sistema para la adquisición de datos se usó una tarjeta de Adquisición de datos de *Measurement Computer* (MC) modelo USB-1608GX-2AO y el software de Matlab R2012a utilizado para el procesamiento de los datos que se explicara posteriormente.

3.2.1.2. Protocolo para la Adquisición de Datos

Para la selección de los sujetos que participaron en los registros, se definieron los criterios de inclusión, exclusión y eliminación; de acuerdo con un médico especialista en electrofisiología. Los criterios son los siguientes:

Criterios de Inclusión:

Personas sanas entre 20 a 30 años de edad

- Con mano derecha dominante
- Sin antecedentes de enfermedades neurológicas
- Sin antecedentes de enfermedades psiquiátricas
- Sin antecedentes de enfermedades tóxico-metabólicas

- Sin antecedentes de traumatismos craneoencefálicos
- Sin antecedentes de familiares con epilepsia o alguna otra enfermedad heredo-degenerativa.

Criterios de Exclusión:

Se excluyeron del protocolo aquellas personas con:

- Antecedentes de enfermedades neurológicas
- Antecedentes de enfermedades psiquiátricas
- Antecedentes de enfermedades tóxico-metabólicas
- Antecedentes de traumatismos craneoencefálicos
- Antecedentes de familiares con epilepsia o alguna otra enfermedad heredo-degenerativa.

Criterios de Eliminación:

Los registros de EEG se eliminaron en caso de existir alguno de los siguientes casos:

- Retiro Voluntario del protocolo
- Imposibilidad para realizar el estudio (vista cansada, excesivo movimiento palpebral con ojos abiertos e incluso ojos cerrados).
- Presencia de electro morfogramas anormales.

Los sujetos se presentaron al registro con el cabello limpio, seco y sin gel o fijador, sin maquillaje ni crema en el rostro. A cada voluntario se le explicó en qué consistía su participación y estando de acuerdo en su participación, se procedió a la firma del consentimiento informado, y se le pidió que contestara un cuestionario con sus datos personales para estadística.

Para el registro de la actividad cerebral se utilizó el amplificador comercial marca Nicolet, modelo NicOne de 32 canales. Se colocaron 22 canales

referenciales en el cuero cabelludo correspondientes al sistema internacional 10-20 (ver Figura 3.2). La tierra y la referencia se colocaron en la línea central de la frente. Además de colocaron electrodos en el borde palpebral de ambos ojos para registrar los movimientos oculares; así como en los maseteros derecho e izquierdo para detectar los artefactos por movimientos de la boca.

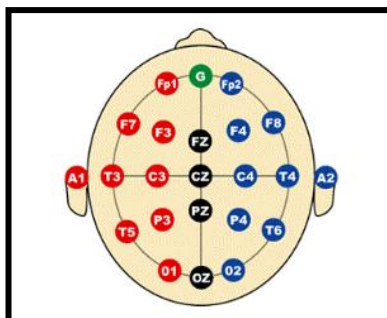


Fig. 3.2 Colocación de los electrodos de acuerdo al sistema Internacional 10-20

Para definir el protocolo de registro se tomó como base el método propuesto por Pfurtscheller y colaboradores para la discriminación de dos estados mentales. La tarea del paciente fue imaginar y ejecutar movimientos de la mano derecha e izquierda (abrir y cerrar) dependiendo de una guía visual presentada en la pantalla de una computadora. Después de realizar las pruebas preliminares se decidió modificar y complementar el método de Graz como se explica en los siguientes párrafos.

Se le pidió al sujeto que se sentará en una posición cómoda en un sillón con descansa brazos. Se le indicó que se mantuviera despierto y relajado. Frente a él se colocó una pantalla de computadora. Esta permaneció cubierta, excepto cuando se utilizaba la interfaz visual. Primero se realizó el registro de la actividad basal del sujeto (ojos cerrados y ojos abiertos), con el objetivo de evaluar el estado fisiológico del paciente. Después se le instruyó verbalmente para que realizara movimiento continuo de abrir y cerrar de mano derecha e izquierda, y finalmente se le solicitó que imaginara el mismo movimiento de las manos, estos tiempos se plantearon como un corto entrenamiento de las actividades que realizarían posteriormente en las pruebas del paradigma de Graz. Se definieron

intervalos intermedios de ojos cerrados entre cada actividad para que el sujeto pudiera descansar la vista y pudiera realizar las pruebas con ojos abiertos sin parpadeos excesivos (ver Tabla 3.1).

No.	Estado	Duración
1.	Ojos cerrados	3 minutos
2.	Ojos abiertos	1 minuto
3.	Ojos cerrados	1 minuto
4.	Ojos abiertos	1 minuto
5.	Ojos cerrados	1 minuto
6.	Movimiento de mano derecha (abrir/cerrar) con OA	1 minuto
7.	Movimiento de mano izquierda (abrir/cerrar) con OA	1 minuto
8.	Ojos cerrados	1 minuto
9.	Imaginar movimiento de mano derecha (abrir/cerrar) con OA	1 minuto
10.	Imaginar movimiento de mano izquierda (abrir/cerrar) con OA	1 minuto
11.	Ojos cerrados	1 minuto
12.	Pruebas con movimiento real de manos.	4-5 minutos
13.	Ojos cerrados	1 minuto
14.	Pruebas con imaginación de movimiento de manos.	4-5 minutos
15.	Ojos cerrados	1 minuto

Tabla 3.1 Estados del sujeto y los tiempos designados para cada uno de ellos

Ya en las pruebas con la interfaz visual, el paciente permanecía con los ojos abiertos observando la pantalla de la computadora colocada a 150 cm de distancia de sus ojos. Cada prueba fue de 8 segundos de duración e inicia con la presentación de una cruz en el centro del monitor, seguido de un tono auditivo de corta duración (*beep*) a los 2 s. Posteriormente a los 3 s, la cruz desaparece y se muestra una flecha en el centro del monitor apuntado hacia la derecha o la izquierda durante 1.5seg. Dependiendo de la dirección de la flecha, el sujeto fue instruido para ejecutar o imaginar un movimiento de la mano derecha o izquierda. El paciente debe de ejecutar el movimiento real o mantener la imaginación hasta los 8 s, después se deja un periodo aleatorio de relajación y descanso entre cada prueba con la pantalla de la computadora sin ninguna guía visual. La pantalla de descanso es una pantalla azul (no indicada en el paradigma de Graz) para indicar al sujeto que había finalizado la prueba y se le indicó que podía parpadear para descansar la vista (ver Figura 3.3). La prueba de 8 s se repite 20 veces,

apareciendo de manera aleatoria 10 veces de flecha derecha y 10 de izquierda, esto con el fin de evitar la habituación del sujeto.

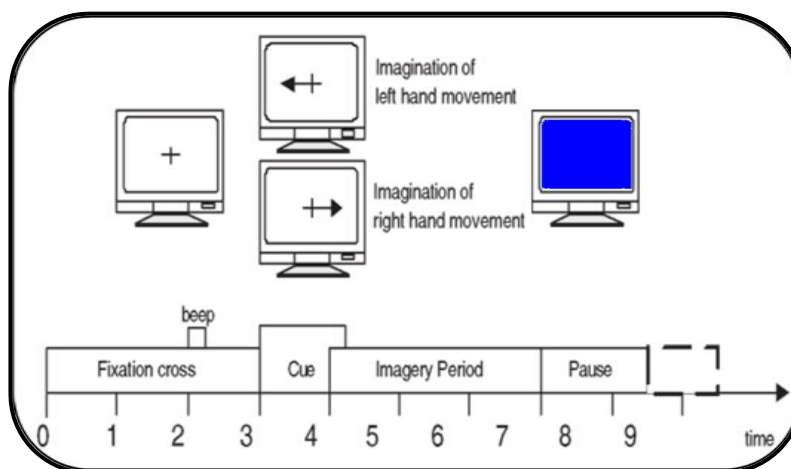


Fig. 3.3 Protocolo de Pfurtscheller

Al final de cada registro se tienen 20 intervalos de 5 s de movimiento de mano izquierda y derecha, y 20 intervalos de imaginación de movimiento de mano izquierda y derecha. Para verificar cuando el paciente estaba realizando el movimiento de las manos se colocaron electrodos en los brazos para registrar su actividad mioeléctrica.

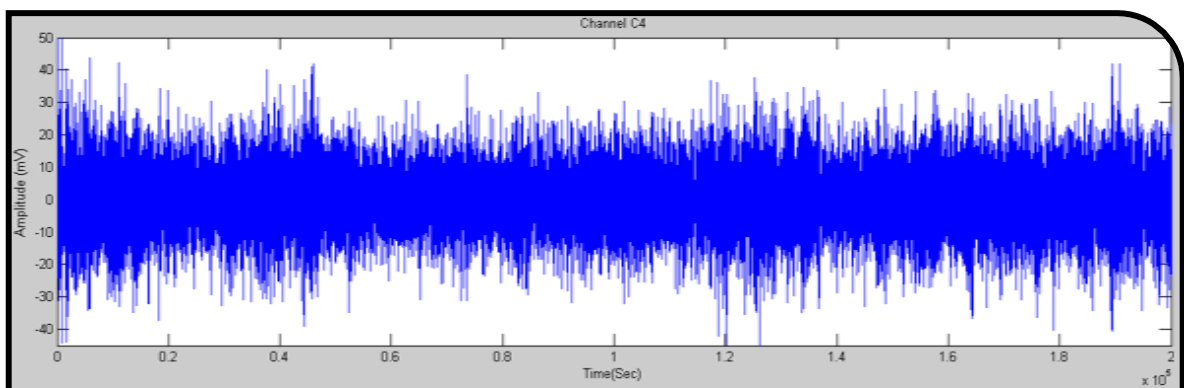
También se registró EMG de los ambos brazos, para lo cual se colocaron electrodos desechables de tamaño pediátrico en la parte interna. Los electrodos activos se colocaron en la región donde se localiza el músculo flexor superficial de las falanges y el músculo flexor profundo de las falanges, un electrodo cerca del origen de los músculos y el otro a cinco centímetros. El electrodo de referencia se colocó en la apófisis estiloides del hueso cúbito de la extremidad superior. La Figura 3.4 muestra la colocación utilizada en los sujetos participantes en las mediciones.



Fig. 3.4 Colocación de los electrodos para registrar los movimientos de la mano

Una vez adquiridos los datos de cada sesión se guarda en un archivo HDF5 [26], el cual es compatible con el software de Matlab para proceder a su pre-procesado, procesado, y clasificación de la señal. Este archivo guarda la información de 12 canales EEG y 2 canales EMG; además de que en este archivo se almacena la amplificación de cada canal, tipo de electrodo, se puede saber en qué momento se indica la tarea de mover la mano (izquierda o derecha) o la imaginación de estas por medio de la señalización por pulsos *Triggers*. En este caso se guardaron los *Triggers* para indicar cada señalización del protocolo de Pfurtscheller (flecha, beep, mover, imaginar, descansar).

En la figura 3.5, se muestra dos graficas obtenidas del archivo HDF5 en las cuales muestran el comportamiento de las señales obtenidas de los electrodos ubicados en C4 y C3; lo cuales están colocados en el área sensorio motor de las manos.



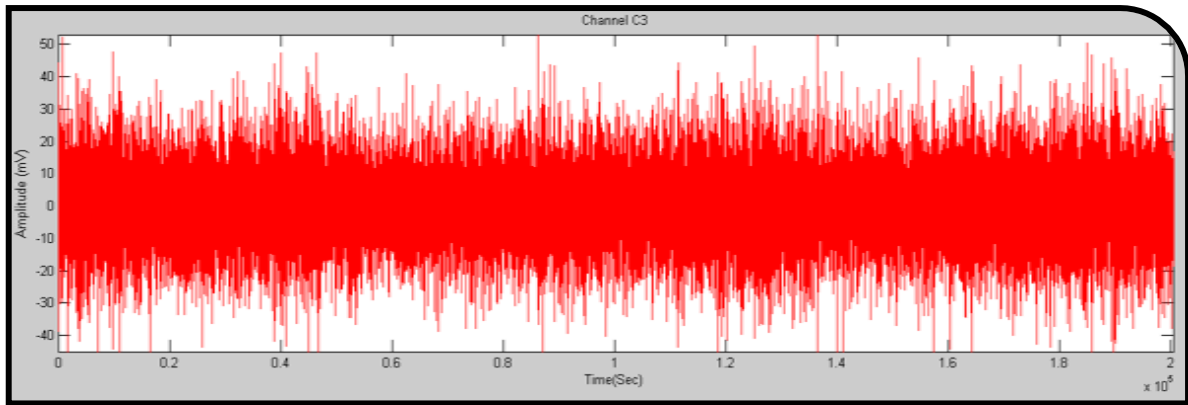


Fig. 3.5 Señales provenientes de los electrodos C3 y C4 (área sensorio motor)

3.2.1.3. Imaginación motora

Estudios realizados por Dennis McFarland, Laurier A. Miner [15], han demostrado que la imaginación motora puede ser detectada únicamente por medio de dos electrodos que son los que detectan los movimientos o imaginación de ambas manos; estos electrodos basándose en el Sistema Internacional 10-20 son los ubicados en la posición C3 (actividad con la mano derecha) y C4(actividad con la mano izquierda) referencia a Figura 3.2.

Como se mencionó en capítulos anteriores (2.2.2), basado en estudios que describen a detalle el proceso de imaginar un movimiento, [16] [17] [18]; se puede decir que la imaginación motora es una parte del fenómeno relativo a la preparación de un movimiento. Es decir; imaginar un movimiento genera el mismo proceso mental e incluso físico, a excepción de que el movimiento queda bloqueado. Por lo tanto en base a estos dos estudios por medio de dos electrodos se puede detectar cuando una persona que alguna vez tuvo la posibilidad de mover los brazos; puede generar la misma actividad cerebral en estas zonas al momento de imaginar el movimiento. A este fenómeno se le conoce como Imaginación Motora (IM).

3.2.2. Obtención de los ritmos Mu y Beta

Se sabe de trabajos citados anteriormente [15],[16],[17] y [18]; que los ritmos Mu (8-12Hz) y Beta (18-26Hz) nos sirven para la detección de movimientos o la imaginación de movimientos. En este sistema se pretende utilizar los ritmos Mu y Beta para poder determinar qué movimiento desea realizar el usuario; entre los que puede ser mover la mano derecha o izquierda, todo esto también dependerá del lado de activación cerebral. Es decir, se analizará los ritmos Mu y Beta de los electrodos C3 y C4, colocados según el sistema Internacional 10-20 (Figura 3.2), los cuales están ubicados en el área motora cerebral de las manos (donde se genera la actividad neuronal del movimiento de las manos). El electrodo ubicado en la posición C3 tendrá una supresión cuando se mueva o se imagine mover la mano derecha, mientras que el electrodo ubicado en la posición C4 detectará una supresión al mover o imaginar mover la mano izquierda.

Para la obtención del ritmo Mu, se sabe de los trabajos citados anteriormente que la información que nos brinda se encuentra en el intervalo de 8Hz a 12Hz; mientras que para la obtención del ritmo Beta se encuentre dentro del intervalo 18Hz-26Hz. Una vez adquirido un tramo de la señal (256 muestras), se crea una ventana hamming del mismo tamaño para analizar a mayor detalle el tramo. La ventana hamming se muestra en la figura 3.6. La cuál posee las siguientes ventajas mejora el decaimiento de altas frecuencias, el ancho de pulso es bueno, genera poco rizado, menos región de transición, pero al final en las discontinuidades decae (atenúan) mucho los datos de la señal en los extremos. Teniendo en cuenta estas características se optó por un *overlapping* de la señal para resaltar los extremos de estas, pero viendo que los resultados eran lentos y no varía mucho se optó por remover el *overlapping* de la señal.

El ventaneo se utiliza con el fin de evitar que las discontinuidades introducidas al analizar solo una fracción de la señal o al introducir muestras con un valor cero introduzcan componentes de alta frecuencia en el espectro, que son más bien artefactos de las discontinuidades introducidas y no pertenecen a la señal.

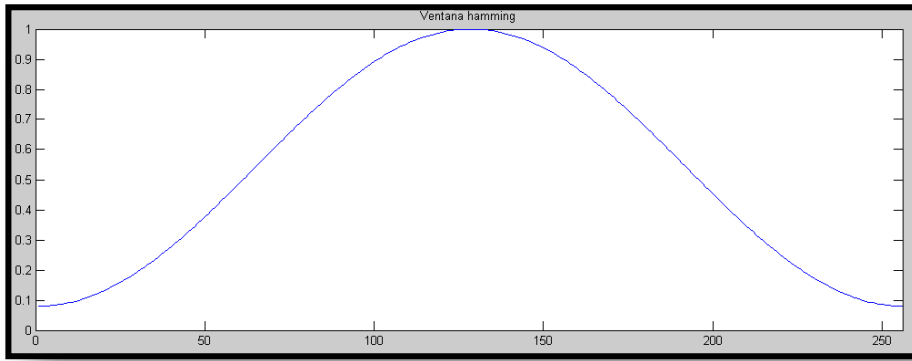


Fig. 3.6 Ventana Hamming utilizada para el ventaneo de la señal

3.2.2.1. Filtro Butterworth

Un filtro digital, es un filtro que opera sobre señales digitales. Es una operación matemática que toma una secuencia de números (la señal de entrada) y la modifica produciendo otra secuencia de números (la señal de salida) con el objetivo de resaltar o atenuar ciertas características de la señal. Puede existir como una fórmula en un papel, un *loop* en un programa de computadora, como un circuito integrado en un chip.

El filtro Butterworth, es un filtro básico, con una respuesta más plana en la banda de paso y caída aguda en la frecuencia de corte a razón de $20n$ (dB/dec); donde 'n' indica el orden del filtro en nuestro caso $n=6$. El orden del filtro tiene que ver con el número de polos de la función de transferencia o con el número de redes presentes en la estructura. Mientras mayor sea el orden del filtro más aproximada será la respuesta del filtro. La función de transferencia del filtro es (fórmula 1):

$$H(j\omega) = \frac{K_{pb}}{\sqrt{1 + \left(\frac{\omega}{\omega_c}\right)^{2n}}} \quad (1)$$

$n=1,2,3\dots$ orden del filtro

Fórmula 1. Función de transferencia del filtro Butterworth

Para la creación del filtro en Matlab se usan los comandos “buttord, Butter y filtfilt”; donde buttord nos da el orden del filtro en ‘N’ y ‘Wn’ la cual indica la frecuencia natural para que el filtro alcance las especificaciones dadas (frecuencia de paso, frecuencia de corte, atenuación por década). Después, se comprobó que el filtro funciona muy bien dejándolo con un orden de 6 teniendo una buena respuesta a nuestra necesidad. Ya que tenemos el orden y la frecuencia natural del filtro, con el comando butter se crea los coeficientes del filtro; debido a que la frecuencia natural en nuestro caso es de dos valores el filtro que se creara será pasa banda. Finalmente se usa el comando ‘filtfilt’, el cual con los coeficientes y la señal dada crea el filtro con fase cero hacia adelante y con respuesta al impulso infinito (salida con un número infinito de términos nulos); como se indica en fórmula 2:

$$a(1) * y(n) = b(1) * x(n) + b(2) * x(n - 1) + \dots + b(nb + 1) * x(n - nb) - a(2) * y(n - 1) - \dots - a(na + 1) * y(n - na)$$

Donde

a,b: Coeficientes del filtro

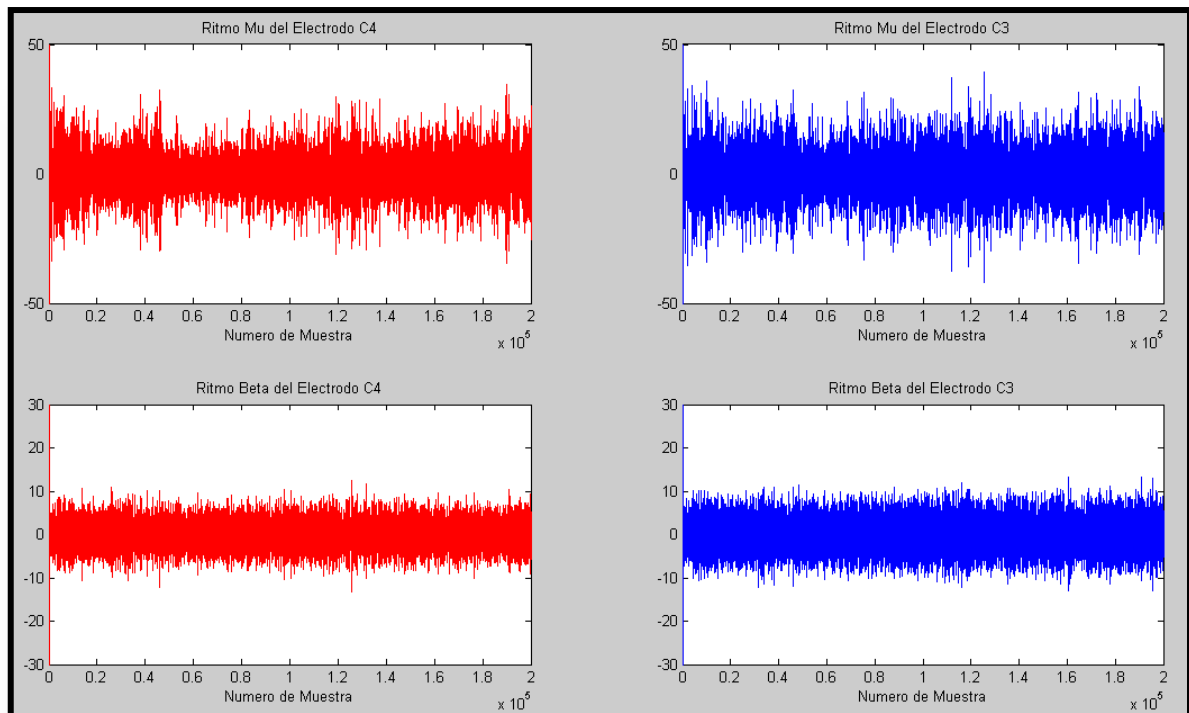
(2)

x: señal a filtrar

n: número de iteración

Fórmula 2. Ecuación de diferencia del filtro IIR con fase cero

En la figura 3.7 se muestra las señales de salida después de pasar por la etapa de filtrado para la obtención de los Ritmos Mu y Beta de cada canal; después de obtener estos ritmos se procede a obtener la dimensión fractal de cada trama (256 muestras) por medio de tres métodos que se mencionaran más adelante.



. Fig. 3.7 *Salidas del filtro digital de los Ritmos Cerebrales para cada canal*

3.2.3. Estimación del Exponente de Hurst y la Dimensión Fractal

A principios del siglo XX, con el surgimiento de la geometría fractal o geometría moderna, la cual estudia los aspectos geométricos que son invariantes con el cambio de escala, se origina una nueva herramienta para la representación de las diversas formas presentes en la naturaleza.

Para encontrar una respuesta sólida, que solucione el problema de representar elementos de la superficie de forma más precisa, se presenta la forma de evaluar dos atributos de gran relevancia en el estudio de la geometría fractal, como son el exponente de Hurst y la dimensión fractal; los cuales en señales se relaciona el grado de aleatoriedad de una señal [27].

3.2.3.1. El Exponente de Hurst

El coeficiente o exponente de Hurst es una medida de independencia de las series de tiempo que fue estudiada inicialmente por el científico británico Harold Edwin Hurst, como elemento para distinguir series fractales. Hurst descubrió que muchos fenómenos naturales exhiben un comportamiento que puede ser caracterizado por un proceso aleatorio sesgado, en el cual existe “memoria de largo plazo” entre las observaciones, es decir, que los eventos de un periodo influyen en todos los siguientes.

Posteriormente, las ideas de Hurst fueron retomadas por Benoit B. Mandelbrot quien generalizó su trabajo y lo llamó análisis de rango reescalado (R/S), definido como un método estadístico utilizado para evaluar la ocurrencia de eventos poco comunes, dando origen a una herramienta ideal para procesos físicos y financieros, aunque no se limita solamente a este tipo de eventos, por lo que puede ser usado en cualquier serie de tiempo (Gutiérrez, 2008).

Si $H=0.5$ (ruido blanco) implica un proceso completamente aleatorio e independiente, con ausencia de correlación entre los incrementos de la señal. Correlación = 0. Si $0.5 < H \leq 1$ implica series de tiempo que muestran procesos persistentes o correlacionados (un periodo de crecimiento es seguido de otro análogo) y presentan un aspecto suave. Una serie persistente está caracterizada por efectos de memoria de largo plazo. Teóricamente lo que suceda hoy impactara en el futuro por siempre, todos los cambios diarios están correlacionados con todos los cambios futuros. Se ha encontrado que las series persistentes son las más comunes en la naturaleza. Un $H=1$ indicaría un comportamiento determinístico. A este comportamiento también se le atribuye el nombre de *Ruido Negro*, ya que se relaciona con procesos cíclicos de largo plazo, como niveles de ríos, cambios de precios en la bolsa de valores, entre otros. Si $0 \leq H < 0.5$ corresponde a un comportamiento de anti-persistencia o anti-correlacional en la serie de tiempo (un periodo de crecimiento es seguido de otro de decrecimiento) que se caracteriza por un mayor contenido de alta frecuencia.

Un incremento en los sucesos del pasado, supone un descenso en los sucesos futuros y viceversa. Un sistema anti-persistente tiende a regresar constantemente al lugar de procedencia y tienen la particularidad de ser señales muy irregulares. A este tipo de comportamientos se le conoce también como *Ruido Rosa* y se caracterizan por que abundan en la naturaleza y se encuentran relacionados con procesos de turbulencia [27].

En 1975 Mandelbrot demostró que el valor del exponente de Hurst (H) se encuentra acotado entre 0 y 1, explicando que este fenómeno ocurría debido a un síntoma de invariancia al cambio de escala, que es la característica clave de las series de tiempo fractales. (Figura 3.8)

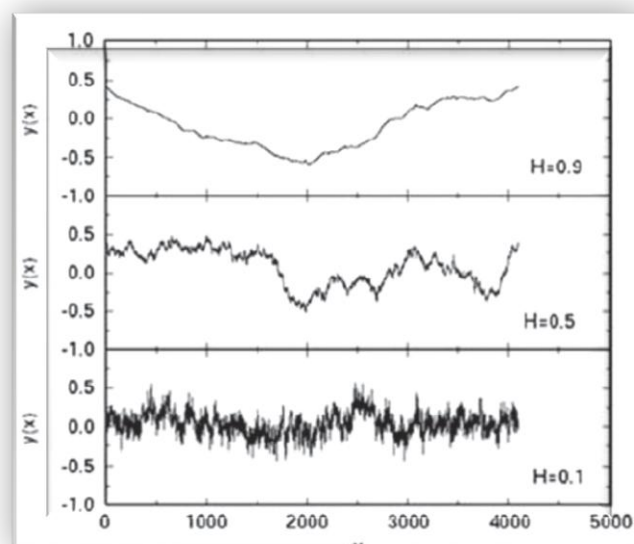


Fig. 3.8 Valores del exponente de Hurst (H) en señales: $H = 0,9$, comportamiento persistente; $H = 0,5$, comportamiento aleatorio; $H = 0,1$, comportamiento anti-persistente.

Para obtenerla dimensión fractal a partir del Exponente de Hurst (o viceversa) se aplica la siguiente fórmula:

$$D = 2 - H \quad (3)$$

Donde D: Dimensión Fractal

H: Exponente de Hurst

3.2.3.2. Algoritmo de Higuchi

Higuchi propone un algoritmo para estimar la Dimensión Fractal directamente en el dominio del tiempo sin la necesidad de reconstruir un atractor para el caso de la mayoría de los algoritmos en el dominio de la frecuencia. Este algoritmo da una estimación razonable de la FD en el caso de una trama de datos pequeños; se basa en que dada una serie de tiempo $y = \{y(1), y(2), \dots, y(N)\}$, una nueva serie y_m^k , es reconstruida por medio de la siguiente ecuación:

$$y_m^k = \left\{ y(m), y(m+k), y(m+2k), \dots, y\left(m + \left\lceil \frac{N-m}{k} \right\rceil * k\right) \right\} \quad (4)$$

para $m=1,2,\dots,k$

donde m: tiempo de inicio

k: duración del intervalo

La longitud de $L_m(k)$, de cada curva se obtiene por la siguiente fórmula:

$$L_m(k) = \frac{1}{k} \left\{ \left(\sum_{i=1}^{\left\lceil \frac{N-m}{k} \right\rceil} |y(m+i*k) - y(m+(i-1)*k)| \right) * \frac{N-1}{\left\lceil \frac{N-m}{k} \right\rceil * k} \right\} \quad (5)$$

El largo de la curva para el tiempo del intervalo k, $L_m(k)$ es considerada como el promedio de la curva m. La relación de este algoritmo es $L_m(k) \propto k^{-D_{HM}}$, por medio de un ajuste de mínimos cuadrados de $\log_2(L(k))$ versus $\log_2 k$, obtenemos la pendiente de esta relación la cual nos indica la estimación de la dimensión fractal [28].

Al aplicar este algoritmo a los ritmos Mu y Beta; obtenemos la siguientes gráficas, donde se puede observar los decrementos muy marcados cuando se imagina el movimiento de mover alguna mano, cuando se imagina mover alguna mano; la señal de C3 es marcada que la de C4 pero esto solo se puede apreciar a detalle cuando se analizan los números. A continuación se mostraran las gráficas de la dimensión fractal obtenida por medio del Algoritmo de Higuchi (Figura 3.9).

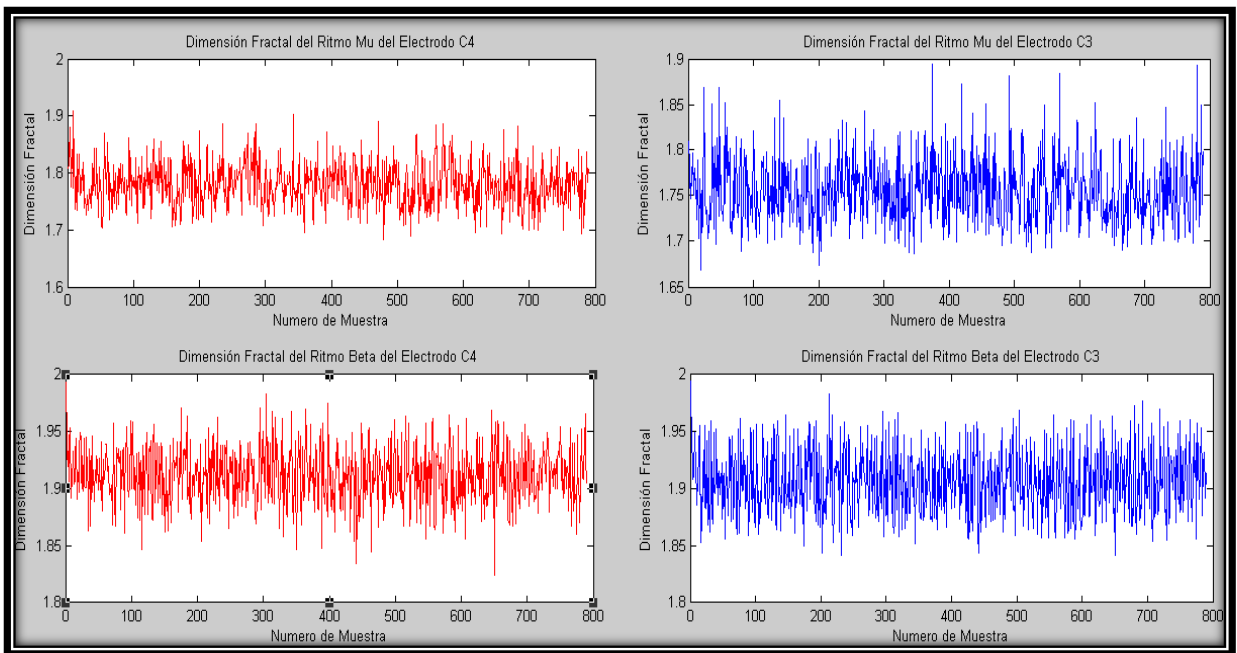


Fig. 3.9 *Dimensión Fractal por medio del Algoritmo de Higuchi*

3.2.3.3. Análisis de Fluctuación sin Tendencia (DFA)

DFA es un método de análisis de escalamiento que proporciona un parámetro cuantitativo sencillo para representar las propiedades de auto-correlación de una señal y ver su comportamiento a largo plazo. También es conocida por su robustez frente a las señales no estacionarias. Cuantifica la complejidad de las señales que utilizan la propiedad fractal. Este método es una modificación de la raíz cuadrada media para el paseo aleatorio. El promedio de distancia cuadrada de la señal de la línea de tendencia local, se analiza en función del parámetro de escala.

Por lo general, la dependencia de poder de la ley y el parámetro interesante es el exponente. En muchos casos la escala exponente DFA se puede utilizar para discriminar los datos sanos y patológicos

El algoritmo para obtener la dimensión fractal por este método es el siguiente. Primero, se calcula la señal integrada en base a la siguiente fórmula:

$$y(k) = \sum_{i=1}^k (B(i) - B_{avg}) \quad (6)$$

Donde B_{avg} : Valor promedio

Después, se dividen los datos en segmentos de tamaño 'n' y se procede a encontrar una aproximación lineal de Y_n usando un ajuste por mínimos cuadrados en cada segmento. La fluctuación promedio de $F(n)$ de la señal alrededor de la tendencia está dada por:

$$F(n) = \sqrt{\left(\frac{1}{N} \sum_{k=1}^N (y(k) - y_n(k))^2\right)} \quad (7)$$

Los cálculos se repiten para todos los n. Nuestro interés gira en torno a la relación de $F(n)$ y el tamaño del segmento n. En general $F(n)$ incrementará con el tamaño del segmento n. Con una gráfica logarítmica de $\log F(n)$ vs $\log n$, la pendiente de la gráfica nos indica el exponente de escala α .

$$F(n) \sim n^\alpha \quad (8)$$

El parámetro α representa las propiedades de auto-correlación de la señal o su exponente de Hurst; para obtener su dimensión fractal basta con usar la fórmula 3 [29].

En la Figura 3.10 se muestra la salida resultante al aplicarle a los ritmos cerebrales el algoritmo de DFA, en el cual para poder analizar la señal también es necesaria revisar los números debido a que en la gráfica no se puede apreciar los resultados a simple vista.

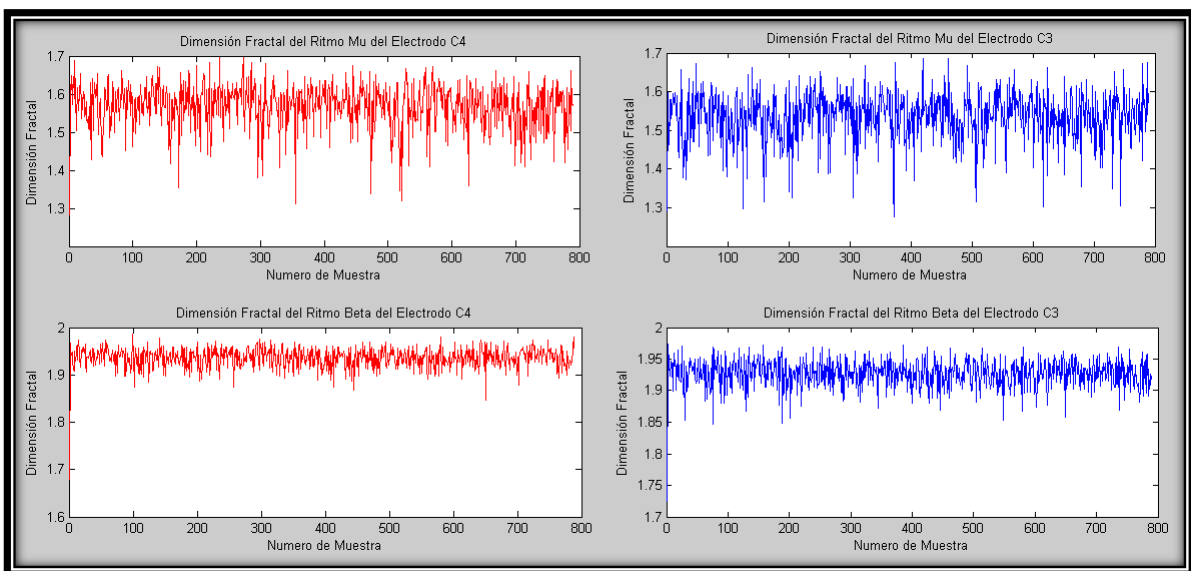


Fig. 3.10 Dimensión Fractal por medio del Algoritmo de Análisis de Fluctuación sin tendencia

3.2.3.4. Potencia de la Densidad Espectral

Este método es ampliamente utilizado para la evaluación de las propiedades fractales de series de tiempo, el método basado en el análisis de dominio de frecuencia, y también trabaja sobre la base de la escala de potencia $\frac{1}{f}$, el cual puede ser obtenido por medio de la transformada rápida de Fourier (fft). Esto es conocido como la relación de ley de potencia de Mandelbrot y Van Ness y se puede expresar de la siguiente manera:

$$f(s) \propto \frac{1}{s^{-\beta}} \quad (9)$$

Donde s es la frecuencia, $f(s)$ es la potencia correspondiente a la frecuencia dada, y β es el exponente de escala el cual es estimado calculando la pendiente negativa de la relación lineal de $\log f(s)$ vs $\log s$, por último el exponente de Hurst puede ser determinado por medio de $H = \frac{(\beta-1)}{2}$, donde $1 < \beta < 3$; y para estimar la dimensión fractal se aplica la siguiente fórmula:

$$FD_{PSD} = 2 + \frac{(1+\beta)}{2} \quad (10)$$

En la figura 3.11, se muestran las gráficas de la dimensión fractal obtenida por medio de la Densidad de Potencia Espectral.

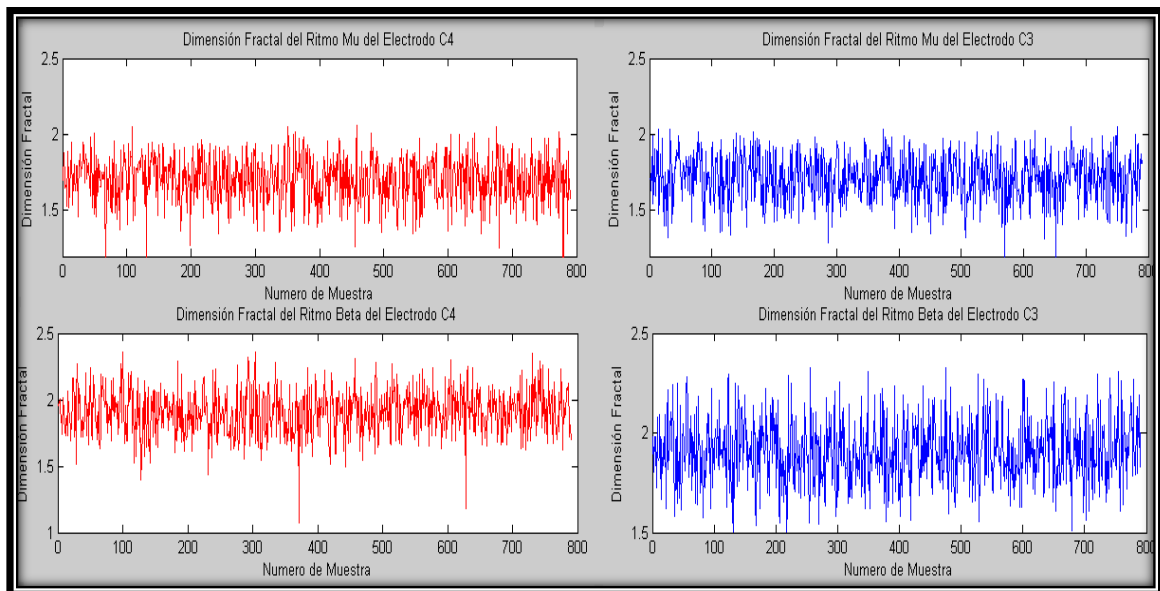


Fig. 3.11 Dimensión Fractal por medio del Algoritmo de la Densidad de Potencia Espectral

Como se puede observar en los tres métodos se obtuvieron 4 diferentes series de tiempo caóticas por algoritmo; estas muestras representativas de las primeras señales y con las cuales podemos determinar por medio de la Inteligencia Artificial que movimiento está imaginando el usuario realizar que es lo que se explicará en la siguiente subsección.

3.2.4. Clasificación por Redes Neuronales Artificiales (RNA)

Para el entrenamiento de la señal primero se buscó una dimensión que fuera la dominante (umbral) cuando se movía cada mano, después se procedió a ver si concordaba con un umbral cuando se imaginaba el movimiento. Ya teniendo estos datos empezó a buscar en que momentos la dimensión fractal coincidía al caso en que se imaginara mover alguna mano; una vez que se encontró el umbral para cada persona se procedió a entrenar a la red con los datos obtenidos.

Para entrenar la red se probaron varios algoritmos de aprendizaje contenidos en el *tool box* de Matlab, pero al final se escogieron los que nos dieron mejor resultado (más preciso) estos fueron *Levenberg-Marquardt* con Propagación hacia atrás (*trainlm*) y el Gradiente Descendente de impulso y con propagación hacia atrás adaptiva (*traingdx*). Los parámetros que se escogieron para tener un buen entrenamiento fueron los siguientes:

- Número de épocas antes de mostrar el resultado: 10
- Máximo número de época para el entrenamiento: 1000
- Límite de error para detener el entrenamiento: 0
- Tasa de aprendizaje: 0.001

3.2.4.1. *Levenberg-Marquardt* con Propagación hacia atrás

Este algoritmo es una modificación del método de Newton, que fue

diseñado para minimizar funciones que sean la suma de los cuadrados de otras funciones no lineales; es por ello que el algoritmo de Levenberg-Marquardt, tiene un excelente desempeño en el entrenamiento de redes neuronales donde el rendimiento de la red está determinado por el error cuadrático medio. El algoritmo de Levenberg-Marquardt se puede resumir de la siguiente manera:

1. Se presentan todas las entradas a la red (a_q^M), se calculan las correspondientes salidas t_q , y cada uno de los errores según la fórmula 11:

$$e_q = t_q - a_q^M \quad (11)$$

Se calcula después, la suma de los errores cuadrados para cada entrada $e(x)$

2. Se calculan las sensibilidades individuales y la matriz sensitiva total y con estas, se calculan los elementos de la matriz Jacobiana (Fórmula 12).

$$[\mathbf{J}]_{h,l} = \frac{\partial e_{k,q}}{\partial w_{i,j}^m} = \frac{\partial e_{k,q}}{\partial n_{i,q}^m} * \frac{\partial n_{i,q}^m}{\partial w_{i,j}^m} = s_{i,h}^m * \frac{\partial n_{i,q}^m}{\partial w_{i,j}^m} = s_{i,h}^m * a_{j,q}^{m-1} \quad (12)$$

3. Se obtiene Δx_k
4. Se recalcula la suma de los errores cuadrados usando $x_k + \Delta x_k$. Si esta suma es más pequeña que el valor calculado en el paso 1, entonces se divide por ' μ ' por ' σ ', se calcula $x_{k+1} = x_k + \Delta x_k$ y se regresa al paso 1. Si la suma no se reduce entonces se multiplica ' μ ' por ' σ ' y se regresa al paso 3.

El algoritmo debe alcanzar convergencia cuando la norma gradiente de:

$$\nabla e(x) = 2J^T(x)v(x) \quad (13)$$

Sea menor que algún valor predeterminado, o cuando la suma de los errores cuadrados ha sido reducida a un error que se haya trazado como meta. Como puede verse, este algoritmo converge en pocas iteraciones, requiere mucha más computación por iteración, debido a que implica el cálculo de matrices inversas. A pesar de su gran esfuerzo computacional sigue siendo el algoritmo de entrenamiento más rápido para redes neuronales cuando se trabaja con un moderado número de parámetros en la red, si el número de parámetros es muy grande utilizarlo resulta poco práctico por el tiempo de entrenamiento [30].

3.2.4.2. Gradiente Descendente de impulso y con propagación hacia atrás adaptativo

Este método combina el aprendizaje adaptativo con el entrenamiento de momento. Estos métodos se explicaran brevemente por separado.

Gradiente descendente con momento, permite a una red responder no sólo al gradiente local, sino también a las tendencias recientes en la superficie de error. Actuando como un filtro de pasa bajo, el momento de la red permite hacer caso omiso de pequeñas características en la superficie de error. Sin momento una red puede quedarse atrapada en un mínimo local. Con momento la red puede desplazarse a través de tal mínimo y salir de él. El parámetro MC es el momento constante que define la cantidad de momento. MC se encuentra entre 0 (sin movimiento) y valores cercanos a 1 (mucho movimiento). Con un momento constante de 1 la red es totalmente insensible a los gradientes locales y por lo tanto la red no aprende correctamente.

La tasa de aprendizaje adaptativa trata de mantener el tamaño del paso de

aprendizaje tan grande como sea posible, manteniendo el aprendizaje estable. La tasa de aprendizaje se hace sensible a la complejidad de la superficie del error local. Un aprendizaje adaptativo requiere algunos cambios en el procedimiento de entrenamiento. Primero, se calcula la salida inicial de la red así como el error. En cada época los pesos y las ganancias son calculados usando la tasa de aprendizaje actual, después se calculan las salidas y los errores. Igual que en el caso en el que se tiene momento, si el nuevo error excede el anterior por más que una relación predefinida (típicamente 1.04), los nuevos pesos y ganancias son descartados. En adición, la tasa de aprendizaje es reducida. De otra manera los valores de las variables son mantenidos. Si el nuevo error es menor que el anterior, la tasa de aprendizaje es incrementado.

La mezcla de estos dos métodos; así como la retro propagación forman el método de Gradiente Descendente de impulso y con propagación hacia atrás adaptiva; un método muy usado en la Inteligencia Artificial [31].

3.2.5. Clasificación por Maquina de Soporte de Vectores (SVM)

Para la clasificación por medio de Maquinas de Soporte se usaron dos de los Kernel más comunes que son el “Gaussiano” y “Polinomial”; se escogieron estos debido a que en artículos leídos son de los más usados en aplicaciones BCI y parte del objetivo es ver con que método nos arroja mejores resultados[30].

3.2.5.1. El kernel Polinómico

Se ha demostrado que si $k(x, z)$ es un kernel válido y $p(\cdot)$ un polinomio con coeficientes positivos, entonces $p[k(x, z)]$ es un kernel válido al que hemos llamado kernel polinómico.

No obstante, cuando se habla del kernel polinómico, es habitual que se

esté haciendo referencia a un caso particular de este kernel polinómico en sentido amplio. En concreto, dado una entrada X contenido en R^n definiremos el kernel polinómico de grado d y offset R del siguiente modo:

$$k_d(x, z) = (\langle x, z \rangle + R)^d \quad \forall x, z \in X \quad (14)$$

Es evidente que este kernel polinómico así definido encaja plenamente en la definición de kernel polinómico más amplio. En efecto, el kernel válido inicial sería $\langle x, z \rangle$ (lo que exige que la entrada X sea un espacio con producto interno) y el polinomio con coeficientes positivos sería $(x + R)^d$.

Vamos a determinar cuál es el feature space y el embedding que resultan de la elección de un kernel polinómico como el que se acaba de definir. Para ello debemos tomar en consideración que $x, z \in X \subseteq R^n$, por lo que:

$$\langle x, z \rangle = \sum_{i=1}^n x_i z_i \quad (15)$$

Y

$$(\langle x, z \rangle + R)^2 = (\sum_{i=1}^n x_i z_i + R)^d \quad (16)$$

Desarrollando la expresión anterior tenemos que:

$$(\langle x, z \rangle + R)^d = \sum_{i_0 + \dots + i_n = d} \frac{d!}{i_0! \dots i_n!} R^{i_0} x_1^{i_1} z_1^{i_1} \dots x_n^{i_n} z_n^{i_n} \quad (17)$$

$$\text{Con } i_j \text{ entero tal que } 0 \leq i_j \leq d \quad \forall j \in \{0, \dots, n\}$$

Para conocer cuál es el número de sumandos de la expresión anterior tenemos que responder a la siguiente pregunta: ¿de cuántas formas se puede obtener el resultado d mediante la suma de $n + 1$ enteros no negativos? La respuesta es:

$$\binom{n+d}{d} = \frac{(n+d)!}{d!n!} \quad (18)$$

Por otra parte, consideremos la función

$$\emptyset: X \rightarrow R^{\binom{n+d}{d}} \quad (19)$$

Que a cada elemento $x \in X \subseteq R^n$ le hace corresponder $\emptyset(x)$, que es un vector cuyos $\binom{n+d}{d}$ componentes son de la forma:

$$\left(\frac{d!}{i_0! \dots i_n!} R^{i_0} \right)^{\frac{1}{2}} x_1^{i_1} \dots x_n^{i_n} \quad (29)$$

Siendo $\{i_0, i_1, \dots, i_n\}$ uno de los $\binom{n+d}{d}$ conjuntos de $n+1$ enteros no negativos que cumplen que $0 \leq i_j \leq d \quad \forall j \in \{0, \dots, n\}$ y que $i_0 + i_1 + \dots + i_n = d$. A partir de estas definiciones resulta que:

$$(\langle x, z \rangle + R)^d = \sum_{i_0 + \dots + i_n = d} \left[\left(\sqrt{\frac{d!}{i_0! \dots i_n!}} R^{i_0} \right) x_1^{i_1} \dots x_n^{i_n} \right] \left[\left(\sqrt{\frac{d!}{i_0! \dots i_n!}} R^{i_0} \right) z_1^{i_1} \dots z_n^{i_n} \right] \quad (20)$$

En definitiva, el kernel polinómico $k_d(x, z) = (\langle x, z \rangle + R)^d$ induce:

- Una función de embedding \emptyset de $X \subseteq R^n$ en $R^{\binom{n+d}{d}}$ que a cada elemento $x \in X$ le hace corresponder un vector $\emptyset(x)$ cuyos componentes son de la forma:

$$\left(\frac{d!}{i_0! \dots i_n!} R^{i_0} \right)^{\frac{1}{2}} x_1^{i_1} x_n^{i_n} \quad (21)$$

- Un feature space $R^{\binom{n+d}{d}}$ con el producto escalar estándar y cuyas dimensiones están indexadas por todas las expresiones de la forma:

$$x_1^{i_1} x_n^{i_n}$$

Es decir, por todos los monomios de grado menor o igual que d (tégase en cuenta que en la expresión anterior se ha omitido el índice i_0), que constituyen los patrones que cabe para localizar en el espacio de entrada X .

Finalmente, conviene destacar que el parámetro R (llamado offset) permite ajustar el peso relativo de cada uno de los monomios. En efecto, a partir de la expresión

$$\left(\frac{d!}{i_0! \dots i_n!} R^{i_0}\right)^{\frac{1}{2}} x_1^{i_1} \dots x_n^{i_n} \quad (22)$$

Vemos que cuanto mayor sea i_0 y, en consecuencia, cuanto menor sea el grado del monomio $x_1^{i_1} \dots x_n^{i_n}$ mayor será la influencia de R en el valor de la expresión anterior. En definitiva, un valor elevado de R proporciona un elevado peso relativo a los monomios de grado reducido y, por tanto, un menor peso relativo a los monomios de grado elevado.

3.2.5.2. El Kernel Gaussiano

En la sección dedicada a la presentación y caracterización de las funciones kernel en general se presentó el caso del siguiente kernel (al que se denominó kernel gaussiano):

$$k(x, z) = e^{\left[-\frac{\|x-z\|^2}{2\theta^2}\right]} \quad (23)$$

Su validez como función kernel se justificó a partir de tres hechos:

- El desarrollo en serie de Taylor de la función exponencial
- La conservación por el paso al límite de la propiedad de una función simétrica de ser finitamente semi-definida positiva.
- La normalización de una función kernel

En este apartado pasaremos revista a algunas de las propiedades más destacables del kernel gaussiano que, por su flexibilidad, es uno de los más frecuentemente utilizados:

- Las imágenes por el embedding inducido por el kernel gaussiano son vectores unitarios. En efecto:

$$\|\phi(x)\|^2 = \langle \phi(x), \phi(x) \rangle = k(x, x) = e^0 = 1 \quad (24)$$

- Dados dos elementos x, z distintos de X , el ángulo que forman sus imágenes por el embedding en el feature space es un ángulo agudo. Efectivamente, por una parte:

$$k(x, z) = e^{\left[\frac{-\|x-z\|^2}{2\sigma^2}\right]} \leq 1 \quad (25)$$

y por otra parte

$$k(x, x) = k(z, z) = 1$$

En consecuencia,

$$\cos(\phi(x), \phi(z)) = \frac{\langle \phi(x), \phi(z) \rangle}{\sqrt{\langle \phi(x), \phi(x) \rangle \langle \phi(z), \phi(z) \rangle}} = \frac{k(x, z)}{\sqrt{k(x, x)k(z, z)}} \leq 1 \quad (26)$$

- Esto implica que el feature space puede elegirse de manera tal que todas las imágenes de los elementos de la entrada se encuentren en un hiperoctante.

- El parámetro σ del kernel gaussiano permite controlar la flexibilidad del kernel. Si el parámetro σ es muy reducido, la función kernel asignará valores muy cercanos

a cero a las parejas formadas por elementos distintos de la entrada (aunque su diferencia sea muy pequeña). De esta forma, la matriz kernel será muy parecida a la matriz identidad y se caerá en el problema de overfitting. En el otro extremo, valores muy elevados de σ tendrán como consecuencia que los valores asignados por la función kernel a las parejas de elementos serán muy cercanos a 1 (sean o no parecidos). La función kernel será, en estos casos, una función prácticamente constante y no será de utilidad para la detección de patrones.

- No es sencillo visualizar cuál es el feature space y el embedding inducidos por el kernel gaussiano aunque siempre cabe utilizar el teorema que afirma que dado cualquier kernel válido podemos definir un feature space y un embedding asociados con él. Aplicando dicho teorema tenemos que el feature space asociado al kernel gaussiano está formado por todas las funciones de la forma.

$$f = \sum_{i=1}^l \alpha_i k(x_i, \cdot) = \sum_{i=1}^l \alpha_i e^{\left(\frac{-\|x_i - \cdot\|^2}{2\sigma^2}\right)} \quad (27)$$

Donde $\{x_1, x_2, \dots, x_l\}$, es un conjunto finito de elementos del espacio de entrada X y los $\alpha_1 \dots \dots \alpha_l$, son números reales. En cuanto al embedding resulta ser:

$$\phi(x) = e^{\left(\frac{-\|x - \cdot\|^2}{2\sigma^2}\right)} \quad (28)$$

Puede comprobarse que la dimensión del feature space asociado al kernel gaussiano es infinita. Una vez revisadas las características básicas de las funciones kernel, establecida su caracterización como funciones reales simétricas y finitamente semi-definidas positivas y presentadas las características básicas de los kernels polinómico y gaussiano, dedicaremos la sección siguiente a la presentación de una batería de procedimientos para extraer información de la matriz kernel.

3.2.6. Simulación de la adquisición de datos en tiempo real

Para la simulación de este proyecto se utilizó una PC con Matlab en donde están guardados los archivos HDF5 mencionados anteriormente; en los cuales se guarda la información recolectada previamente del EEG. En esta PC se lee la señal guardada (archivo HDF5), se adquieren los canales necesarios para el procesamiento de la imaginación motora de las manos (C4 y C3), se le aplica un filtro '*notch*' para evitar ruido y se divide en tramos de 256 muestras. Cada tramo se envía por medio de una tarjeta de Measurement Computing (MC) USB-1608GX-2AO; que se comunica a la PC por medio del puerto USB. En mismo programa recibimos los datos simulando que se están adquiriendo los datos en el momento y a cada tramo que se recibe se procede a obtener sus ritmos Mu y Beta por medio de los filtros mencionados en el apartado 3.2.2; después de obtener los ritmos cerebrales se obtiene la dimensión fractal de la señal para después poderla clasificar por medio de Redes Neuronales Artificiales o Maquinas de Soporte de Vectores. Con esto podemos saber en cada tramo de la señal que movimiento deseó (imagino) realizar el usuario.

Capítulo 4

4. Resultados y discusiones

En este capítulo de la tesis, se incluyen los resultados de los diferentes algoritmos de procesamiento para demostrar que por medio de la dimensión fractal y la ayuda de métodos de Inteligencia Artificial se puede diferenciar estados mentales como en este caso la imaginación de los movimientos de las manos.

En base a los resultados obtenidos se toma una conclusión del cuál es el mejor algoritmo para la aplicación de este trabajo; el cuál es mover una mano robótica que abra y cierra. Esta aplicación no requiere mucha precisión pero si una respuesta rápida.

4.1. Pruebas

Para las pruebas se obtuvieron las señales de 5 personas de ambos sexos en edades de 20 a 30 años; para cada persona se registraron 12 sesiones de duración aproximada de 12 a 15 minutos cada una; en base a los criterios mencionados en la sección anterior (3.2.1.2). Estas sesiones se tomaron en base al protocolo de Pfurtscheller que también se mencionó anteriormente.

En este protocolo en cada sesión se obtiene 10 registros de cada brazo en movimiento y luego imaginando el movimiento de las manos. En base a esto primero se obtuvo la dimensión fractal de la serie de tiempo (de las señales registradas); una vez obtenida la serie de tiempo fractal de las señales se creó un algoritmo para determinar en qué momento se indica mover la mano derecha y cuando la izquierda; esto por medio de marcadores que se guardan en el archivo HDF5.

Una vez conocido en que tiempo se realiza cada acción se empezó buscar reconocer un patrón dominante en el momento de la ejecución del movimiento para cada brazo; basándose que cuando se mueve el brazo derecho la actividad cerebral se genera en el lado izquierdo (electrodo c3); en cambio cuando se mueve el izquierdo se detecta en el lado derecho (electrodo c4) [15]; se buscó una dimensión fractal umbral para detectar este movimiento en las señales cerebrales. Una vez reconocido el patrón cuando se movía el brazo se revisó en el momento de la imaginación motora para corroborar ambas señales.

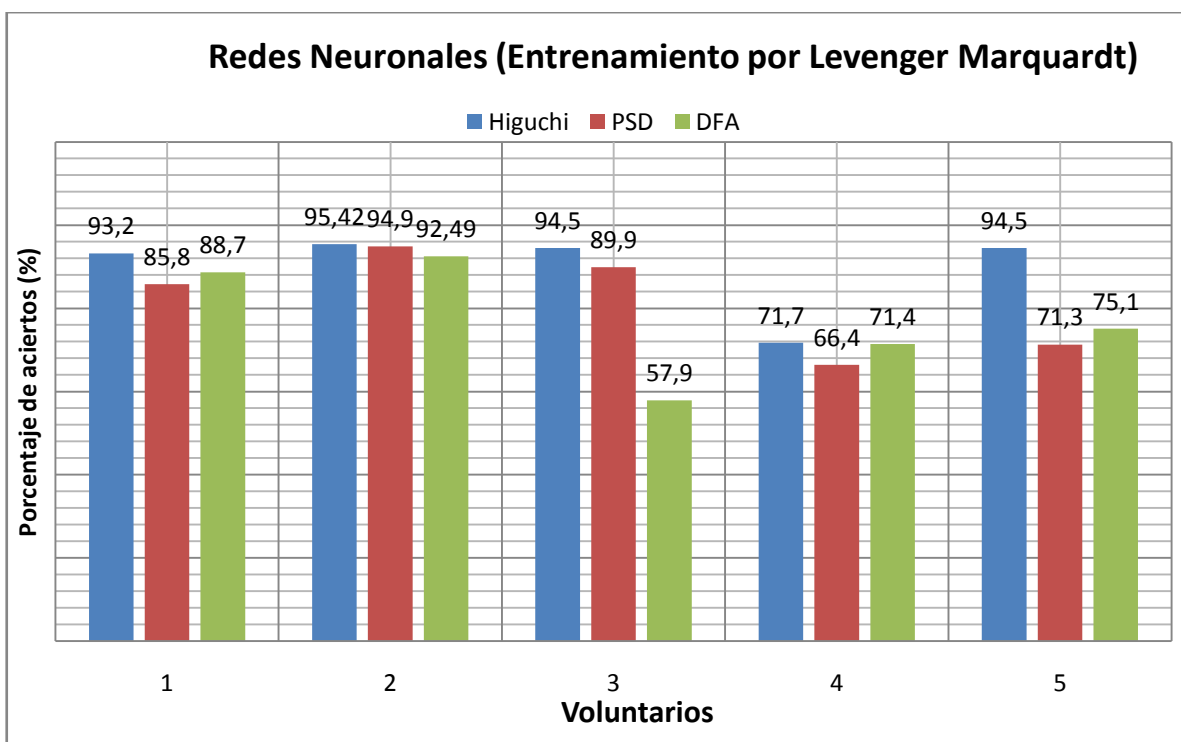
Este procedimiento se realizó para cada persona debido a que la actividad cerebral es diferente en cada persona, así que se buscó un umbral para cada una; además de que para cada método de extracción de la dimensión fractal también es diferente la dimensión. Una vez conociendo cada umbral se entrenó la Red Neuronal y la Máquina de Soporte de Vectores para probarlos con las señales adquiridas; y los resultados expresados en porcentaje se muestran en las siguientes graficas que serán mostradas y explicadas.

4.2. Resultados

En este apartado se muestran las gráficas obtenidas de la evaluación de nuestro algoritmo entrenando el sistema por medio de los resultados obtenidos con la imaginación motora y los resultados son la clasificación de la imaginación motora; en los cuales los resultados de aciertos se localizan entre 60% y 95%; aunque dependiendo del tipo de entrenamiento y el método de extracción de características la velocidad del algoritmo cambia; debido a esto probamos varios métodos.

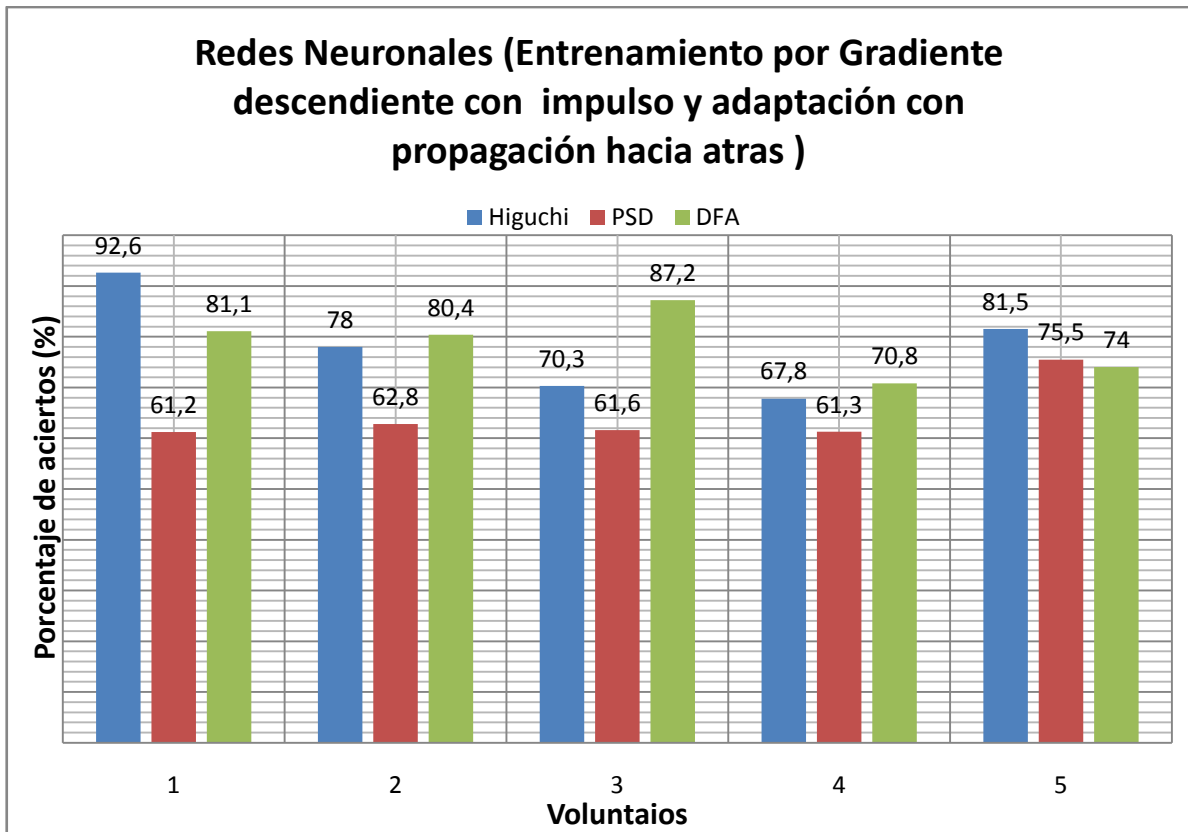
En las siguientes gráficas se muestran los resultados obtenidos por medio de Redes Neuronales; en ambos casos se usaron una capa de entrada con 4 neuronas de entrada, dos capas escondidas con 20 neuronas cada una y una capa de salida con 3 neuronas. Las funciones de transferencia de la red fueron

para las escondidas (la función logarítmica sigmoidea y la función tangente sigmoidea), y para la capa de salida la función tangente sigmoidea. Estas fueron probadas con diferentes neuronas y capas; pero los resultados del entrenamiento resultaron buenos con estas características en la red. A continuación se muestran unas gráficas que indican el porcentaje de aciertos totales usando estas redes. Cada color representa un algoritmo de extracción de características (Higuchi, PSD y DFA); como lo indica en la gráfica y debajo de cada grupo de gráfica se coloca un número para cada paciente para mantener el anonimato de su nombre.

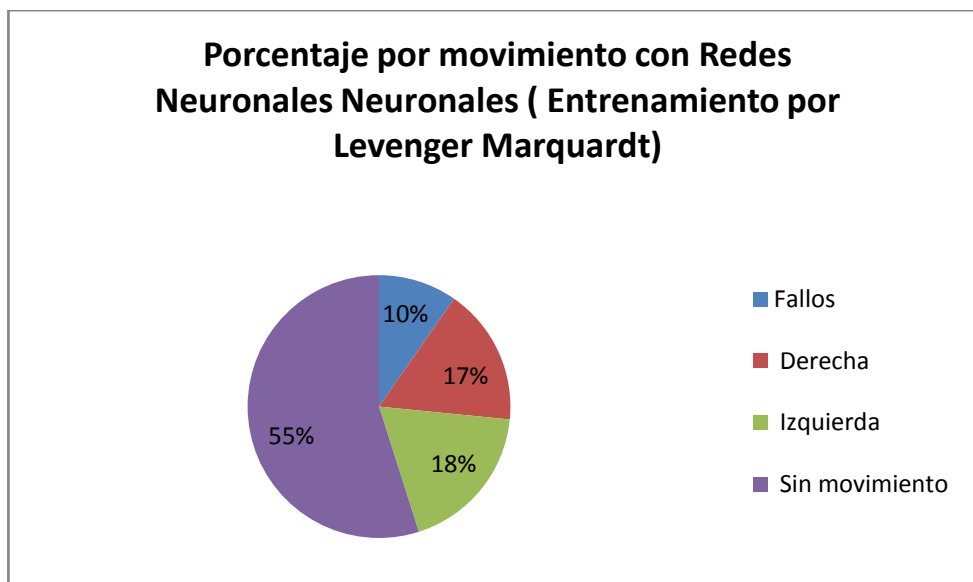


Grafica 4.1 Resultados del entrenamiento por Redes Neuronales por el entrenamiento LM.

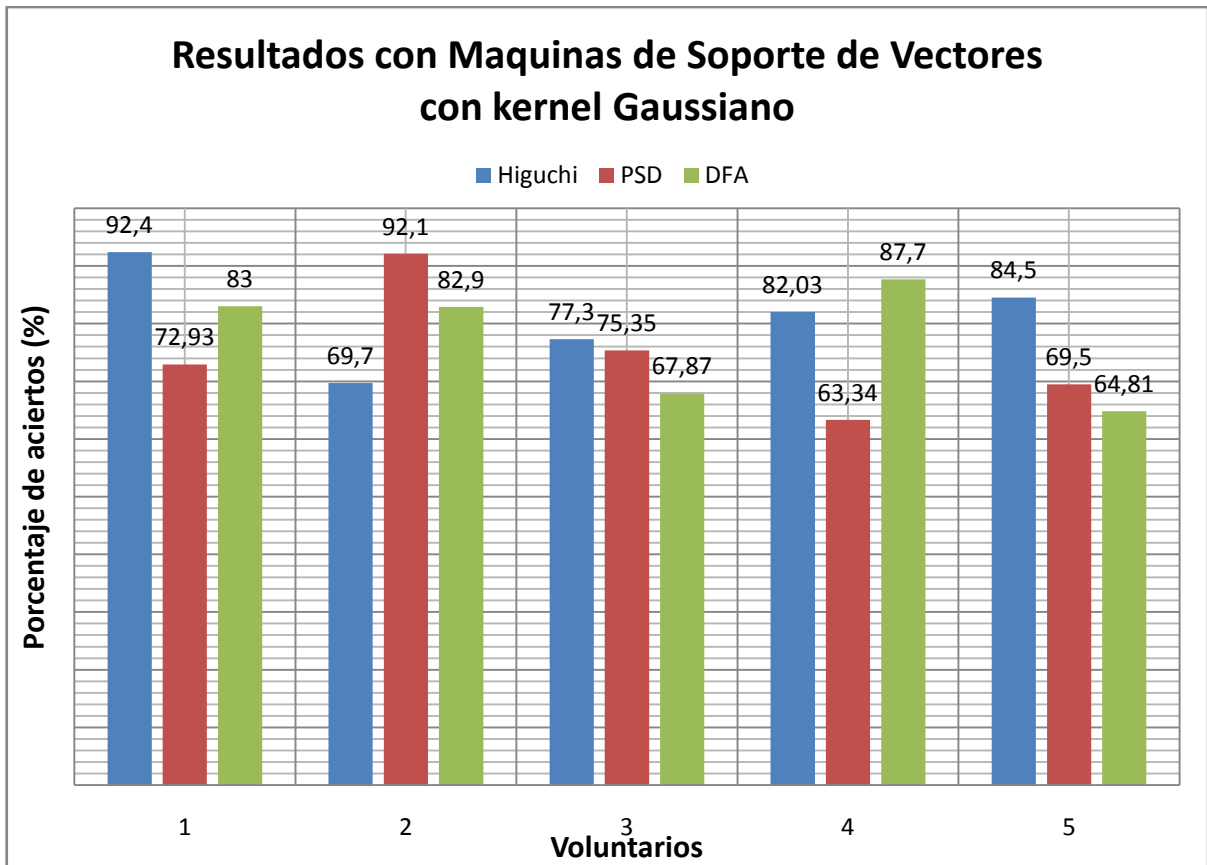
También se agregó una gráfica de tipo pastel (gráfica 4.3), en la cual se escogieron los datos del algoritmo que nos arrojó los mejores resultados para ver detalladamente los aciertos en cada evaluación (sin movimiento, derecha, izquierda, fallos).



Grafica 4.2 Resultados del entrenamiento por Redes Neuronales por el entrenamiento de gradiente descendiente con impulso y adaptación con propagación hacia atrás.



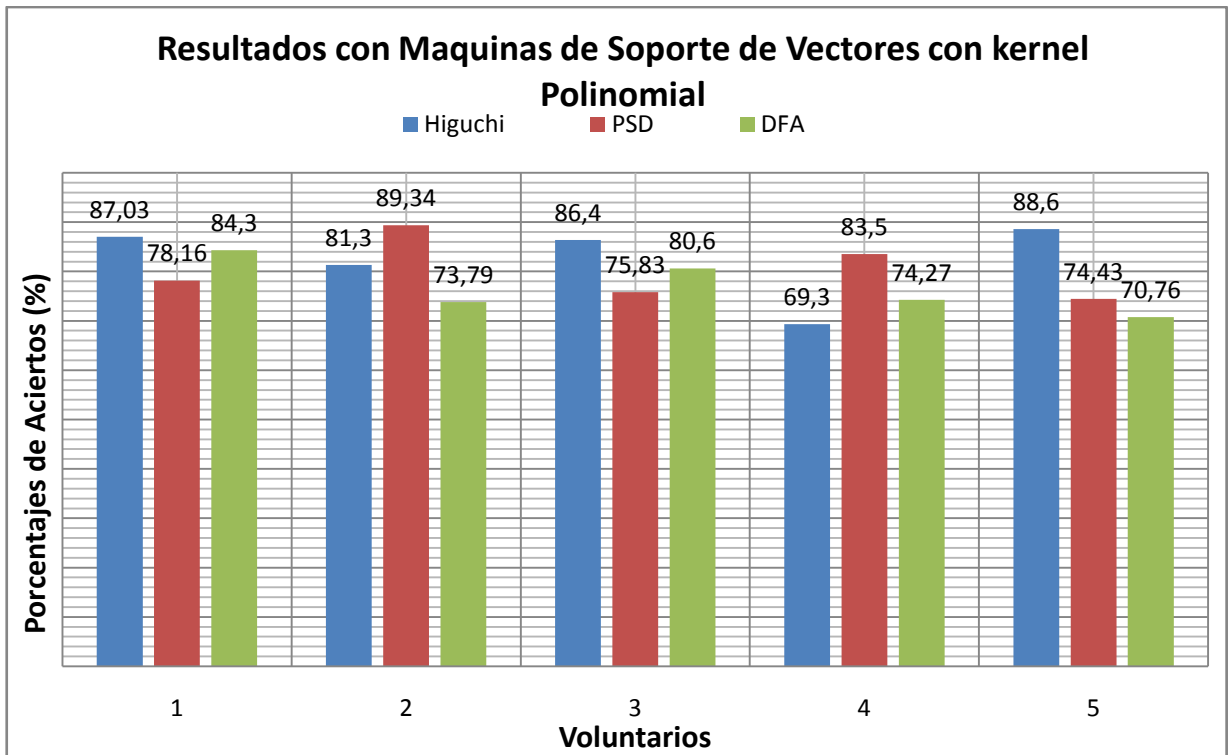
Gráfica 4.3 Gráfica donde se desglosa el resultado de aciertos y fallos para cada imaginación de movimientos de un total de 545 datos.



Grafica 4.4 Resultados del entrenamiento con Maquinas de Soporte de Vectores usando un kernel Gaussiano .

En las gráficas 4.4 y 4.5 se muestran los resultados obtenidos al aplicar el algoritmo con un entrenamiento por medio de Maquina de Soporte de Vectores, para cada método de extracción de características.

Con estas gráficas se puede observar que estos algoritmos son buenos para la detección de la imaginación de movimientos de los miembros superiores (las manos en este caso). Además al final se muestran unas tablas con los tiempos para estos algoritmos en tiempo real; lo cual nos ayuda a escoger cuál es el mejor método según la aplicación deseada.



Grafica 4.5 Resultados del entrenamiento con Maquinas de Soporte de Vectores usando un kernel Polinomial .

Totales (Tiempo en minutos) usando Redes Neuronales
(Entrenamiento por Gradiente descendiente con impulso y adaptación con propagación hacia atrás)

Personas	Higuchi	PSD	DFA
1	27,5	23,2	22,46
2	26,55	22,47	25,4
3	27,32	24,52	23,45
4	27,08	21,5	21,55
5	27,55	23,06	23,48

Tabla 4.1 Tiempo totales de los algoritmos en minutos

Las tablas de la 4.1 a la 4.4 muestran los tiempos promedios de los algoritmos propuestos; en donde se recalca que parte de este tiempo se lleva a cabo la comunicación por medio de la tarjeta de adquisición.

Totales (Tiempo en minutos) usando Redes Neuronales (Entrenamiento por <i>Levenberg Marquardt</i>)			
Personas	Higuchi	PSD	DFA
1	29.13	22.1	19.5
2	25.56	23.4	24.3
3	22.38	27.4	24.5
4	23.52	21.56	27.11
5	27.58	26,13	24,2

Tabla 4.2 Tiempos totales de los algoritmos en minutos

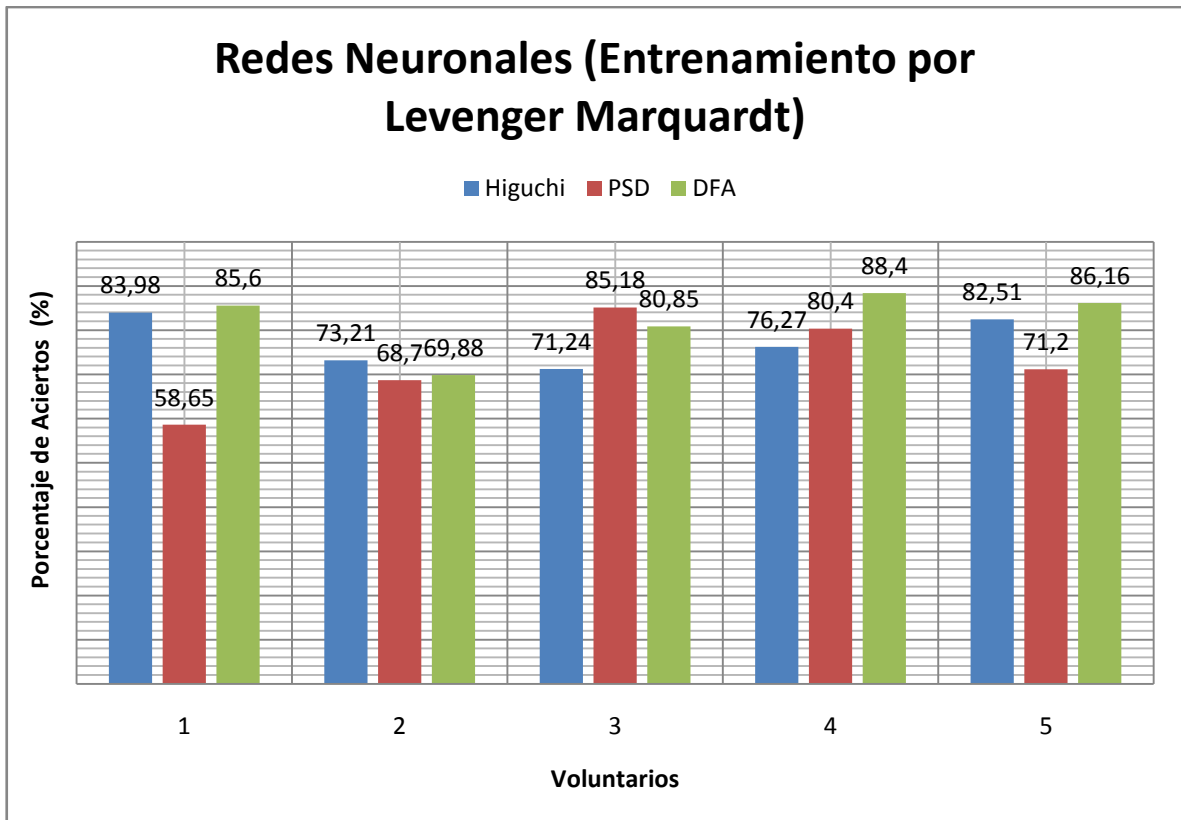
Totales (Tiempo en minutos) usando Maquinas de Soporte de Vectores con Kernel Polinomial			
Personas	Higuchi	PSD	DFA
1	27,3	26,48	25,4
2	28,14	27,42	28,34
3	27,23	28,14	27,54
4	26,55	24,52	24,32
5	26,13	27,33	26,59

Tabla 4.3 Tiempos totales de los algoritmo en minutos

Totales (Tiempo en minutos) usando Maquinas de Soporte de Vectores con Kernel Gaussiano			
Personas	Higuchi	PSD	DFA
1	26,48	27,22	25,48
2	28,24	26,32	26,47
3	27,49	26,17	26,2
4	28,07	25,2	27,31
5	27,23	26,34	26,45

Tabla 4.4 Tiempos totales de los algoritmo en minutos

Además el algoritmo fue probado entrenándose por medio de la imaginación motora y clasificando los resultados obtenidos al realizar el movimiento real; obteniendo los siguientes resultados de la gráfica 4.6.



Grafica 4.6 Resultados por Redes Neuronales por el entrenamiento LM para la clasificación del movimiento real.

4.3. Discusión

Por medio de las gráficas podemos concluir que en primera instancia existen varios métodos que son buenos para la extracción de características de una señal; así como varios métodos de clasificación muy efectivos. En este caso escogimos probar la característica de la Dimensión Fractal por medio del Algoritmos de Higuchi, la Densidad del Espectro de Potencia (PSD), el Algoritmo de Katz, Conteo de cajas y el Análisis de Fluctuación sin Tendencia (DFA). Al ver que no eran buenos los resultados con los algoritmos de Katz y Conteo de cajas se optó por no continuar con su uso y buscar otros métodos, ya que como se mencionó en el segundo capítulo existen muchos métodos de obtener la dimensión fractal pero no todos son buenos para analizar señales EEG.

Para la clasificación escogimos Maquinas de Soporte de Vectores (SVM) y

Redes Neuronales Artificiales (RNA) ambos con buenos resultados; el método de SVM es muy eficiente debido a que arroja resultados más exactos; en cambio con las redes neuronales una misma red puede tener diferentes resultados en cada entrenamiento aunque una vez encontrados los pesos adecuados a la red nos brinda resultados más precisos que SVM.

El mejor resultado se obtuvo con el algoritmo de Higuchi en conjunto con una Red Neuronal Artificial entrenada por el método *Levenberg-Marquardt* obteniendo un porcentaje de aciertos en promedio de 89,86% del total de pruebas. El inconveniente con esta opción es su tiempo de procesamiento ya que es un algoritmo muy lento pero muy preciso, así que para su uso de debe de considerar que es más necesario, si mejor precisión o menor tiempo de respuesta; el más rápido fue usando el algoritmo de DFA en conjunto con Redes Neuronales entrenado por el método de gradiente descendiente.

En promedio el método más preciso es en base a la extracción de características es el método de Higuchi con 82,90% de aciertos, siguiéndole DFA con 77,45% y por último PSD con 75,25%; en base al método de clasificación el mejor es el método con Redes Neuronales entrenado por *Levenberg-Marquardt* con 82,88%, seguido del método de SVM con kernel Polinomial con 79,89%, después el de SVM con kernel Gaussiano con 77,69% y por último el método de Redes Neuronales entrenada por Gradiente descendiente con 73,74%.

En cuanto a la velocidad el más rápido en promedio en base a la extracción de características es con PSD con 25,07 minutos, seguido del DFA con 25,29 minutos y por último como se comento es el de Higuchi con 26,40 minutos; en cuanto al método de clasificación fue con Redes Neuronales entrenadas por *Levenberg-Marquardt* con 24,47 minutos, después en segundo lugar fue por Redes Neuronales entrenada por Gradiente descendiente con 24,55 minutos, en tercer con SVM con kernel Gaussiano con 27,11 y por último con kernel

Polinomial con 27,16 minutos.

En base a estos datos se procede a tener una conclusión de cuál método es mejor para nuestra aplicación que se verá en el último capítulo.

Capítulo 5

5. Conclusiones y Trabajo a Futuro

5.1. Conclusiones

En este trabajo de tesis se desarrollo un programa el cuál pueda analizar y procesar señales EEG por medio de análisis no lineal para determinar qué mano esta imaginado mover una persona, teniendo en cuanto que primero se debe analizar la señal y entrenar al sujeto para ver como trabajo su señal cerebral. Este programa tiene una buena clasificación como se mostró en las gráficas del apartado de resultados (4.2) y el cuál puede ser implementado para un sistema BCI en tiempo real. Ya que como se mostró este sistema puede clasificar tanto cuando se realiza la imaginación motora; así como también cuando se mueve la mano ambas con buena clasificación.

En base a los objetivos se logró el desarrollo de un programa el cual puede:

- Extraer las características de señales EEG por medio de diferentes algoritmos de análisis fractal.
- Se encontró un patrón en las series fractales individuales a analizar en el dominio del tiempo y en el dominio de la frecuencia, para poder determinar en qué momento imagina mover la mano derecha e izquierda.
- Se desarrollaron varios algoritmos para la extracción de características y para la clasificación para comparar cuáles son las diferencias entre los diferentes métodos y determinar cuáles son más efectivos dependiendo de la aplicación que se necesite implementar; ya sea que se prefiera una buena velocidad o se busqué la precisión del tipo de movimiento.

De estas evaluaciones y resultados obtenidos se puede concluir que para nuestro trabajo de tesis; lo primordial es la rapidez ya que el sistema por el momento sus requisitos no se necesitan mucha precisión y es preferible la rapidez

del sistema para lograr nuestro objetivo que será abrir y cerrar las manos derecha e izquierda. Se logró el objetivo de ver la utilidad del análisis fractal para la extracción de características en las señales EEG en el dominio del tiempo y frecuencia; encontrando que es una buena herramienta dentro del análisis no lineal, en las señales caóticas como es el caso de las señales EEG.

Además se comprobó que por medio de solo 2 canales de EEG se puede distinguir cuando se imagina mover los miembros superiores (ambas manos) esto a través del registro y análisis de los electrodos C3 y C4; con los cuales se detecta un decremento en su actividad cerebral, en nuestro caso cuando se imagina mover alguna mano la señal EEG se vuelve menos caótica y esto es lo que mide la dimensión fractal la caosidad de una señal, cuando una persona se concentra su señales se vuelven menos caóticas y más fáciles de predecir que pasara en el futuro.

También se comprobó que el imaginar una señal puede generar la misma actividad cerebral en el EEG que si se realiza el movimiento de la misma; esto gracias de los análisis de los ritmos cerebrales Mu y Beta en los electrodos adecuados.

5.2. Trabajo a Futuro

Este trabajo puede continuar con un sin fin de aplicaciones y trabajos a futuro, empezando con la implementación de este método en un sistema BCI con este tipo de procesamiento o en unión con otras técnicas de procesado de señales para el control en tiempo real del sistema esto se puede lograr con la ayuda de un programa llamado BCI2000 el cuál es compatible con Matlab y con los dispositivos usados para la adquisición de la señal.

Otro trabajos que se pueden derivar es el análisis de imaginación de otras partes del cuerpo piernas, lengua que son las partes del cuerpo humano que tiene más espacio en la corteza motora del cerebro, para poder brindar otras técnicas de rehabilitación a personas con otros tipos de enfermedades motoras.

Por último con el análisis de señales cerebrales crece el abanico de opciones para analizar el cerebro en un sinfín de aplicaciones, existen investigaciones en el área de los sueños por medio de análisis fractal, detección de estados de ánimos, actualmente en la TU Graz [10], Austria están analizando por medio de análisis el reconocimiento de imaginar letras, esto con el fin que las personas puede escribir sin el uso de las manos, entre otras aplicaciones a nivel mundial.

REFERENCIAS

- [1] Discapacidad motora en el Mundo 2011, Organización Mundial de la Salud (OMS),2011
<http://www.citamericas.org/archivos/8eb24c8f67879c1b5b9bfdef68ba48f2.pdf>
- [2] INEGI. Censo de Población y Vivienda 2010, *Cuestionario ampliado*. Estados Unidos Mexicanos/Población con discapacidad/Población total y su distribución porcentual según condición y causa de limitación en la actividad para cada tamaño de localidad y sexo.
- [3] <http://cuentame.inegi.org.mx/poblacion/discapacidad.aspx?tema=P>
- [4] Castellanos, Pilar, et al. *Electrofisiología humana. Un enfoque para Ingenieros*. México: UAM, 1997.
- [5] VV.AA. [ROBERTO HORNERO, REBECA CORRALEJO Y DANIEL ALVAREZ]. Grupo de Ingeniería Biomédica (GIB). Universidad de Valladolid
http://www.fgcsic.es/lychnos/es_ES/articulos/Brain-Computer-Interface-aplicado-al-entrenamiento-cognitivo
- [6] Dewan, Edmond, 1967, "Occipital Alpha Rhythm Eye Position and Lens Accommodation," *Nature*, No. 214, pp. 975-977, Jun 1967.
- [7] L.A. Farwell and Donchin, 1988, "*Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials*", *Electroencephalography and clinical Neurophysiology* , 1988, 70:510-523
- [8] Wolpaw, J. and McFarland, D., 1994, "*Multichannel EEG-based braincomputer communication*", *Electroencephalography & Clinical Neurophysiology*,vol 90, pp. 444 449.

- [9] Kalcher, J. et al., 1996 *“Graz Brain-Computer Interface II: towards communication between humans and computers based on online classification of three different EEG patterns”*, *Medical & Biological Engineering & Computing*. Vol 34, pp. 382-388.
- [10] <http://bci.tugraz.at/>
- [11] <https://www.bccn-berlin.de/Home/>
- [12] <http://sccn.ucsd.edu/~yijun/>
- [13] Maribel Lima; “Sistema Electroencefalográfico (EEG) de 4 canales utilizado en la activación de un mecanismo que induce el movimiento de las manos”, México, BioelectrónicaCinvestav, 2013
- [14] Claudia Martínez, “Aproximación por Escalamiento Dinámico de Electroencefalogramas: Epilepsia en Modelo Animal”; México, ESIME IPN, 2009
- [15] Dennis J. McFarland*, Laurie A. Miner*, Theresa M. Vaughan*, and Jonathan R. Wolpaw* *“Mu and Beta Rhythm Topographies During Motor Imagery and Actual Movements” Brain Topography, Volume 12. Number 3, 2000 pp. 177-186*
- [16] J. Annett, Motor imagery: Perception or action, *Neuropsychologia*, 33, 11,1995, pp. 1395-1417.
- [17] Decety, J., Jeannerod, M., y Prablanc, C. “The timing of mentally represented actions”. *Behav. Brain Res.* 1989, 34, 35-42.
- [18] Decety, J. Lindgren, M. “Sensation of effort and duration of mentally executed actions”. *Scand. J. Psychol.* 1991, 32, 97-104.

- [19] José L. Martín, Sira E. Palazuelos, Luciano Boquete, Manuel Mazo “Estudio Comparativo de dos técnicas de procesado de señales EEG para discriminación de estados mentales”, España, Universidad de Alcala
- [20] *Webster, J. et al, 1992, “Medical Instrumentation: Application and Design”. Houghton Mifflin Company, ISBN 0-395-59492.*
- [21] *Dennis McFarland, Lynn McCane, Jonathan Wolpaw, 1997 “Spatial filter selection for EEG-based communication” Electroencephalography and clinical Neurophysiology 103, p.p. 386-394*
- [22] *Herbert Ramoser, Johannes Müller-Gerking, and Gert Pfurtscheller “Optimal Spatial Filtering of Single Trial EEG During Imagined Hand Movement” IEEE TRANSACTIONS ON REHABILITATION ENGINEERING, VOL. 8, NO. 4, DECEMBER 2000 ,p.p. 441-446*
- [23] *F. Lotte, M Congedo, B Arnaldi “A Review of Classification Algorithms for EEG-based Brain Computer-Interface”, ISRN Signal Processing, 2013, June 2013*
- [24] Carlos Paredes Bartolome, “Aplicación de la Geometría fractal en ciencias de la tierra”, Tesis del Departamento de Matemáticas aplicada y métodos informáticos de Universidad Politécnica de Madrid, 1995.
- [25] Xabier Basogain Olabe, “Redes Neuronales Artificiales y sus Aplicaciones”, Escuela Superior de Bilbao, EHU
- [26] <http://www.hdfgroup.org/HDF5/>

- [27] Oscar Yesid Quintero Delgado y Jonathan Ruiz Delgado “Estimación del Exponente de Hurst y la dimensión fractal de una superficie topográfica a través de la extracción de perfiles”. Universidad Distrital “Francisco José de Caldas”, Universidad Distrital “Francisco José de Caldas” Mayo 2011, Revista Geomática, pp. 84-91
- [28] “Algoritmos de para calcular la dimensión fractal de señales”, Universidad de Mendoza. <http://www.um.edu.ar/catedras/ANASEN/document/fractal/waveform.pdf>
- [29] Agnieszka Kitlas Golinska “Detrended Fluctuation Analysis (DFA) in biomedical signal processing: selected examples”, Department of Medical Informatics, Institute of Computer Science, University of Bialystok, Poland
- [30] <http://proton.ucting.udg.mx/posgrado/cursos/idc/neuronales2/LevMarqB.htm>
- [31] Ing. Eulalio Torres García “Comparación de las Redes Neuronales Artificiales aplicadas al diagnóstico de Turbinas de Gas”, ESIME, IPN. Septiembre 2010
- [32] Juan José Gibaja Martíns “Aprendizaje Estadístico con Funciones Kernel”, Universidad Nacional de Educación a Distancia, Facultad de Ciencias 2010

Anexo A.

Artículos y congresos obtenidos con este trabajo

Durante la elaboración de este trabajo se logró la aceptación en dos ocasiones de la presentación de poster en congresos, donde se presentaron avances de este. Los cuales son:

- Publicación del Artículo *“EEG Signals Processing based on Fractal Dimension Features and classified by Neuronal Network and Support Vector Machine in Motor Imagery for a BCI”* en la revista *IFMBE Proceeding* y poster expuesto en el congreso CLAIB 2014 realizado en Paraná, Entre Ríos, Argentina.
- Poster titulado *“Evaluation of Fractal Dimension estimation of EEG signals for Features extraction in motor Imagery for BCI”*, a exponerse en el Congreso de *Neuroscience* 2014 a realizarse en Washington DC, E.U.A.

Anexo B

Código de Matlab

En este anexo se encuentra el código utilizado para la elaboración de este trabajo de tesis. Este programa se desarrollo en Matlab 2012a, en un PC con Windows XP sp2 de 32btis.

El siguiente código se encarga de detectar en qué momento se indica mover la mano derecha, izquierda, sin movimiento, relajado; esto por medio de un conteo de pulsos.

```
%Programa para calcular en que tiempo se efectuan las acciones en el
%istema; estan pueden ser:
%Abrir Ojos ID=37(w)          Cerrar ojos ID=36(q)
%ID=41
%Aparacion de la cruz(Un pulso):Durante 2seg despues un Beep(dos pulsos)
por 1seg
%Mover mano izquierda(Tres pulsos) Mover mano derecha(Cuatro pulsos)
%Pantalla negra(5 pulsos) y Pantalla azul(6 pulsos)
%Frecuencia de Muestreo=256Hz
clc
clear all
%%
data = h5read('M5_S2_4.hdf5','/RawData/Samples');
%%
%Indica cuanto tiempo dura el pulso
tiempo = h5read('M5_S2_4.hdf5','/AsynchronData/Time');
%%
l_tiempo=length(tiempo);
intervalo=zeros(1,l_tiempo-1);
contiempo=0;
beep=0;
dere=0;
izqui=0;
%Indica que tipo de evento sucedió
tipo = h5read('M5_S2_4.hdf5','/AsynchronData/TypeID');

for i=2:l_tiempo
    intervalo(i)=tiempo(i)-tiempo(i-1);
end
l_intervalo=length(intervalo);
for j=1:l_intervalo
    if intervalo(j)<=12
        contiempo=contiempo+1;
    else
        if contiempo==1
            disp('Beep');
            despliegal=[num2str(ceil((tiempo(j-1))/256)) ' hasta '
num2str(ceil((tiempo(j))/256))];
```

```

        disp(despliega1);
    elseif contienpo==2
        disp('Flecha Izquierda');
        despliega2=[num2str(floor((tiempo(j-3))/256)) ' hasta '
num2str(ceil((tiempo(j))/256))];
        disp(despliega2);
        izqui=izqui+1;
    elseif contienpo==3
        disp('Flecha Derecho');
        despliega3=[num2str(floor((tiempo(j-4))/256)) ' hasta '
num2str(ceil((tiempo(j))/256))];
        disp(despliega3);
        dere=dere+1;
    elseif contienpo==4
        disp('Desaparece');
        despliega4=[num2str(ceil((tiempo(j-4))/256)) ' hasta '
num2str(ceil((tiempo(j-1))/256))];
        disp(despliega4);
    elseif contienpo==5
        disp('Pantalla azul');
        despliega5=[num2str(ceil((tiempo(j-5))/256)) ' hasta '
num2str(ceil((tiempo(j))/256))];
        disp(despliega5);
    elseif contienpo>5
        disp('Algo raro');
    end
    contienpo=0;
end
end
end

```

El siguiente programa es el principal el cuál se encarga de cargar la señal HDF5 a analizar; después separa los canales necesarios para el procesamiento y los envía por medio de la tarjeta de adquisición en tramos de 255 datos, para que cada trama se analice por separado y obtener la dimensión fractal de cada trama; para finalmente clasificarla por medio de AI.

```

% Adquirir datos por partes, divide cada trama de datos en paquetes de
%256 datos, además se empieza a aplicar el ventaneo hamming, los filtros,
% el cálculo la dimensión fractal y cuando se tenga todo las dimensiones
% se clasificaran para saber que se desea mover (derecha, izquierda o sin
movimiento)

%% Write and Read to a MCC 1608GX-2AO DAQ device
clear
clc
[target_test2,red]=redneuron12();%Net Training
fs=256;
tic
[data,tiempo,tipo] = carga();%Cargar datos a analizar
sign = data;

senal2 = sign(14,:);%Derecho (C4)
senal3 = sign(6,:);%Izquierdo (C3)

```

```

salida(1,:)=senal2(1,:);%CH0
salida(2,:)=senal3(1,:);%CH1

%Creacion del filtro notch
Wo = 60/(fs/2); BW = Wo/35;
[b,a] = iirnotch(Wo,BW);
salida=filter(b,a,salida);

senal=double(salida);
senal_out=senal./5000;%Se multiplica por dos para aprovechar el rango de
la daq

muestras=length(senal);
tam=round(muestras/256);
adquirido=zeros(tam,257);
mover=zeros(4,tam);
clasi=zeros(3,tam);
% Initialization of DAQ-----
inicio=1;
for i=1:tam
% Analog Input:
ai = analoginput('mcc');
% Analog output
ao = analogoutput('mcc');
% Adding Channels-----
% Analog Input - Channel 0 to 1
ai0 = addchannel(ai,0:1);
% Analog Output - Channel 0
ao0 = addchannel(ao, 0:1);
% Write Data-----

    ao_value(1,2:257) = senal_out(1,inicio:fs*i);
    ao_value(2,2:257) = senal_out(2,inicio:fs*i);
    ao_value(:,1)=0;
    set(ao, 'SampleRate',256);
    putdata(ao,ao_value');
    start(ao);
    % Configure a one second acquisition with SampleRate set to 512
samples
    % per second and SamplesPerTrigger set to 30000 samples per trigger.
    set(ai, 'SampleRate',256);%sampleper second
    set(ai, 'SamplesPerTrigger',257);%Sample per Trigger
    start(ai);
    adquirido=getdata(ai,257, 'double');
    senal_in(inicio:fs*i,:)= adquirido(2:257,:);
    senal_in2(inicio:fs*i,:)=senal_in(inicio:fs*i,:).*5000;
    mover(:,i)=clasifica6(senal_in2(inicio:fs*i,:));%Features Extraction
by
    % Fractal Dimension with Higuchi,PSD and DFA algorithm

    inicio=inicio+256;
    % Cleaning Up-----
    delete(ai);
    delete(ao);
end
for jjj=1:tam
    clasi(:,jjj)=sim(red,mover(:,jjj));
end
clasi=clasi';

```

```

% target_real=xlsread('M5_Nov.xlsx',3,'f1444:h1981');%Paciente 4
% target_real=xlsread('jessica_4nov2.xlsx',3,'f1258:h1802');%Paciente 1
% target_real=xlsread('pedro_4nov.xlsx',3,'f1221:h1802');%Paciente 3
% target_real=xlsread('jesus_4nov.xlsx',3,'f1244:h1788'); %Paciente 2
% target_real=xlsread('F8_Nov.xlsx',1,'f1441:h1991');%Paciente 5

% plotconfusion(clasi(67:604,:),'target_real');%Paciente 4
% plotconfusion(clasi(83:626,:),'target_real');%Paciente 1
% plotconfusion(clasi(83:662,:),'target_real');%Paciente 3
% plotconfusion(clasi(83:627,:),'target_real');%Paciente 2
plotconfusion(clasi(80:630,:),'target_real');%Paciente 5

toc

```

Para cargar los archivos HDF5 se usa la siguiente función; la cual está dentro del programa carga para tener el llamada de cada archivo para cada persona junto.

```

function [data,tiempo,tipo] = carga();
%Este programa carga los datos a analizar al programa principal

data = h5read('F8_S2_5.hdf5','/RawData/Samples');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tiempo=h5read('MONTALVO_AGUILAR_JESUS_150114_S2_12014.01.15_13.44.01.hdf5',
'/AsynchronData/Time');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tipo=h5read('MONTALVO_AGUILAR_JESUS_150114_S2_12014.01.15_13.44.01.hdf5',
'/AsynchronData/TypeID');
end

```

Programa para entrenar la Red Neuronal, este programa es diferente para cada paciente, además de que el archivo de Excel que se lee es donde se guardaron previamente la dimensión fractal de las señales con sus salidas para poder entrenar a la red.

```

function [target_test2,red]=redneuronal2()

%% F8 target and test
entrenamiento=xlsread('F8_Nov.xlsx',1,'b498:e733');%psd
target=xlsread('F8_Nov.xlsx',1,'f498:h733');%M5
test=xlsread('F8_Nov.xlsx',1,'b1112:e1346');% entradas(2 canales Fd)
M5_test1
test2=xlsread('F8_Nov.xlsx',1,'b1441:e1991');% entrad

tam_entrenamiento=length(test)+length(entrenamiento);
entrenamiento((length(entrenamiento)+1):tam_entrenamiento,:)=test;

```

```

tami=length(entrenamiento);
test=test';
tami2=length(test2);
completo=entrenamiento';
completo(:,tami+1:tami+tami2)=test2';

%% F8
target_real=xlsread('F8_Nov.xlsx',1,'f1112:h1346');%con 3 entradas(3
canales Fd) Jessica_target1
target_real2=xlsread('F8_Nov.xlsx',1,'f1441:h1991');%con 3 entradas(3
canales Fd) Jessica_target2
tam_entrenamiento2=length(target_real)+length(target);
target(length(target)+1:tam_entrenamiento2,:)=target_real;

%% Pedro
% target_real=xlsread('pedro_4nov.xlsx',1,'f968:h1201');%con 3 entradas(3
canales Fd) Jessica_target1
% target_real2=xlsread('pedro_4nov.xlsx',1,'f1221:h1800');%con 3
entradas(3 canales Fd) Jessica_target2
% tam_entrenamiento2=length(target_real)+length(target);
% target(length(target)+1:tam_entrenamiento2,:)=target_real;
%% Entrenamiento de la Red y creacion de la matriz de confusion

[target_test2,red]=clas3(completo,entrenamiento',target',test2');
% plotconfusion(target_real',target_test1);
% plotconfusion(target_real2',target_test2);

%[target_test2,red]=clas3(completo,entrenamiento',target',test2');

end

```

Entrenamiento de la red

```

function [simu2,red]=clas3(completo,entrenamiento,target,test2)
net = newff(minmax(completo), [20 20 3], {'logsig','tansig','tansig'},
'trainlm');

net.trainParam.show=10;
net.trainParam.epochs=1000;
net.trainParam.goal=0;
net.trainParam.lr=0.001;
red = train(net,entrenamiento,target);
simu2=sim(net,test2);

end

```

En el siguiente código se encarga de ventanear la señal por medio de la ventana Hamming, obtener los ritmos Mu y Beta por medio de filtros butter y finalmente obtener la dimensión fractal por diferentes métodos.

```

function [together_dfa]=clasifica6(sign);
%% Analisis fractal
[hur21,hur22]=fractal6(sign(:,1)); %c4

```

```

[hur31,hur32]=fractal6(sign(:,2)); %c3

%% 4 entradas dos canales Fd
tam=length(hur21);
together_dfa=zeros(4,tam);
together_dfa(1,:)=hur21; % Primera señal C4 ritmo mu
together_dfa(3,:)=hur22; % Ritmo beta
together_dfa(2,:)=hur31; % Segunda señal c3 ritmo beta
together_dfa(4,:)=hur32; % Ritmo beta

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
% En este programa leemos la señal del EEG, dividimos en
% ventanas,realizamos un ventaneo tipo hamming; luego fltramos para
% obtener
% algun ritmo y despues obtener su dimension fractal por el metodo de
% Hurst,higuchi und Katz
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%

% Revisar las regiones del ritmo mu(C3(4) Cz(9) C4(12)) a frecuencias 8
1 13Hz y 15 a 25hZ;
% Y ver en que regiones es mas fuerte el ritmo beta
function [fddf1,fddf2]=fractal6(sign);%DFA
function [fdhig1,fdhig2]=fractal3(sign);%Higuchi
function [fdpsd1,fdpsd2]=fractal3(sign);%PSD

% function [fdhig1,hig_int1,fdhig2,hig_int2]=fractal(sign);%Higuchi hoja
[N L] = size(sign);
mu = double(sign);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% %Ventaneo de 1 segundo
hamm=hamming(L);
senal=mu.*hamm';

fm=256;
fe=fm/2;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Después de ventanear la señal procederemos a calcular los coeficientes
%para el filtro del ritmo mu
Wpmu=[8 12]/fe; %8-12hz [7 13]
Wsmu=[7 13]/fe;
Rpmu=1;
Rsmu=2;
[nmu Wnmu]=buttord(Wpmu,Wsmu,Rpmu,Rsmu);
[bmu amu]=butter(6,Wnmu);
mul = filtfilt(bmu, amu, senal);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
mu2 o beta(15 a 25Hz) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Wpmu2=[18 26]/fe;
Wsmu2=[17 27]/fe;
Rpmu2=1;
Rsmu2=2;
[nmu2 Wnmu2]=buttord(Wpmu2,Wsmu2,Rpmu2,Rsmu2);

```



```
[bmu2 amu2]=butter(6,Wnmu2);  
mu2 = filtfilt(bmu2, amu2, senal);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Analisis de DFA%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
[df]=dfa(mu1,8);  
fddfa1=2-df;
```

```
[df]=dfa(mu2,8);  
fddfa2=2-df;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Método de Higuchi %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
fdhig1=higuchi(mu1');  
fdhig2=higuchi(mu2');
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Método PSD%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
fdpsd1=dspfft(mu1');  
fdpsd2=dspfft(mu2');
```

En base a estos programas se pueden obtener los resultados mostrados en esta trabajo de tesis.