

**CLASIFICACIÓN DE SEÑALES CEREBRALES RELACIONADAS CON LA IMAGINACIÓN DE
MOVIMIENTOS PARA APLICACIONES DE BCI**

JORGE ANDRÉS ZACCARO VALVERDE
JAVIER HERNÁN CÁRDENAS DÍAZ
LAURA MARCELA SALAZAR CRUZ

**TRABAJO DE GRADO PARA OPTAR POR EL TÍTULO DE
INGENIERO ELECTRÓNICO**

DIRECTORES:

ING. JULIÁN ARMANDO QUIROGA SEPÚLVEDA, M.Sc.
ING. DIEGO ALEJANDRO PATIÑO GUEVARA, Ph.D.

**PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE ELECTRÓNICA
BOGOTÁ DC, 2011**

**PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA ELECTRÓNICA**

RECTOR MAGNÍFICO: PADRE JOAQUIN EMILIO SÁNCHEZ GARCÍA S.J

DECANO ACADÉMICO: ING. FRANCISCO JAVIER REBOLLEDO MUÑOZ

DECANO DEL MEDIO UNIVERSITARIO: PADRE SERGIO BERNAL RESTREPO S.J

DIRECTOR DE CARRERA: ING. JUAN MANUEL CRUZ BOHÓRQUEZ

DIRECTORES DE PROYECTO: ING. JULIÁN ARMANDO QUIROGA SEPÚLVEDA. M.Sc.,
ING. DIEGO ALEJANDRO PATIÑO GUEVARA, Ph.D.

NOTA DE ADVERTENCIA

Artículo 23 de la Resolución N° 13 de Julio de 1946

“La Universidad no se hace responsable por los conceptos emitidos por sus alumnos en sus trabajos de tesis. Solo velará por que no se publique nada contrario al dogma y a la moral católica y por que las tesis no contengan ataques personales contra persona alguna, antes bien se vea en ellas el anhelo de buscar la verdad y la justicia”.

Índice

NOTA DE ADVERTENCIA	3
1. INTRODUCCIÓN	7
2. MARCO TEÓRICO	9
2.1. SEÑALES CEREBRALES	9
2.1.1. Adquisición	9
2.2. ANÁLISIS DE SEÑALES	10
2.2.1. Análisis en Frecuencia	10
2.2.2. Análisis no Lineal en Tiempo	11
2.2.3. Métodos de Análisis no Lineal-Exponentes de Lyapunov	11
2.2.4. Análisis Estadístico y Transformaciones Lineales	13
2.3. RECONOCIMIENTO ESTADÍSTICO DE PATRONES	15
2.3.1. Espacio de características	15
2.3.2. Métodos paramétricos	15
2.3.3. Métodos no paramétricos	15
2.3.4. Aprendizaje no supervisado	16
2.3.5. Aprendizaje supervisado	16
3. ESPECIFICACIONES	19
3.1. Diagrama de Bloques	19
3.2. MEL: Máximo Exponente de Lyapunov	21
4. DESARROLLOS	22
4.1. Descripción de las señales	22
4.2. Métodos de extracción de características	22
4.2.1. Discrete Wavelet Transform (DWT), Transformada Discreta Wavelet	22
4.2.2. Continuous Wavelet Transform (CWT), Transformada Continua Wavelet	24
4.2.3. Common Spatial Patterns (CSP), Patrones Espaciales Comunes	26
4.2.4. Máximo exponente de Lyapunov	26
4.3. Definición del vector de características	27
4.4. PCA e ICA	27
4.5. Métodos de Clasificación	28
4.5.1. Linear Discriminant Analysis (LDA), Análisis Lineal Discriminante	28
4.5.2. Support Vector Machines (SVM), Máquinas de Vectores de Soporte	28
4.5.3. Artificial Neural Networks (ANN), Redes Neuronales Artificiales	29
5. ANÁLISIS DE RESULTADOS	30
5.1. Cuadros de resultados	31
5.2. Discusión	35
5.3. Resultados anteriores	36
6. CONCLUSIONES	37

Índice de figuras

1.	Esquema descripción del problema. Tomada [37],[9], Matlab Help, [38], [27].	8
2.	Sistema Internacional 10-20 ubicación de electrodos. Tomado de [37].	9
3.	Algoritmo de Wolf	13
4.	Máquina de Vectores de Soporte	17
5.	Diagrama de bloques del sistema.	19
6.	Diagrama de bloques del sistema.	21
7.	Configuración en tiempo de cada ensayo. Tomada de [9].	22
8.	Descomposición Wavelet de 3 niveles. Tomada de Matlab Help	22
9.	Transformada Continua Wavelet de un ensayo	24
10.	CWT del canal C3, Ensayo 81	24
11.	CWT del canal C4, Ensayo 81	25
12.	Mejores Resultados con PCA	31
13.	Mejores Resultados 4 Subconjuntos	35

Índice de cuadros

1.	Bandas generadas por la descomposición utilizada	23
2.	Vector de características	27
3.	Parámetros SVM	30
4.	Parámetros ANN	30
5.	Mejores Resultados	31
6.	Resultados SVM	32
7.	Resultados LDA	33
8.	Resultados ANN	34
9.	Mejores Resultados 4 Subconjuntos	35

1. INTRODUCCIÓN

Los avances tecnológicos de las últimas décadas han llevado al ser humano a rodearse de una gran variedad de máquinas y sistemas computacionales, cuyas aplicaciones van desde los medios de transporte y el manejo de información hasta el entretenimiento y las comunicaciones. A medida que diferentes productos han sido refinados en funcionalidad, la interactividad ha pasado a tomar cada vez mayor importancia para el usuario, por lo que han surgido fuertes tendencias enfocadas en que la comunicación entre el hombre y las máquinas sea cada vez más espontánea.

Una de ellas es conocida como “Brain-Computer Interfaces” (BCI), en la que el objetivo es crear sistemas que procesen las señales cerebrales de un individuo y las conviertan en comandos específicos para un sistema de control dado (e.g. computador). Esto es posible gracias a que la actividad cerebral se manifiesta como potenciales eléctricos que pueden ser medidos tanto de forma invasiva (implantes de electrodos) como no invasiva (EEG, MEG, fMRI), y a que dicha actividad está estrechamente relacionada con procesos motrices, cognitivos y emocionales del cerebro.

Entre las principales aplicaciones de BCI se encuentran la predicción de epilepsia, la rehabilitación de la movilidad, las comunicaciones militares y la interacción con entornos virtuales. Esta última es posible gracias a que la actividad cerebral relacionada con la imaginación de movimientos puede ser registrada, procesada y convertida en comandos de acción coherentes para el contexto, como por ejemplo, en movimientos del puntero de un computador [1], de un personaje en un videojuego o de objetos virtuales en general.

Existen varios laboratorios alrededor del mundo que se encuentran desarrollando proyectos relacionados con esta aplicación, entre los cuales se encuentran el “Laboratory of Brain-Computer Interfaces” en Graz (Austria) [14], el “Bernstein Center for Computational Neuroscience” en Berlín (Alemania) [15] y el “Institute of Neural Engineering” de la Universidad Tsinghua en Beijing (China) [16]. Investigadores de estos y otros laboratorios han participado en la “Competencia BCI” organizada por el grupo Berlin BCI, en la que bases de datos de señales cerebrales son puestas a disposición del público para el entrenamiento y prueba de clasificadores propuestos por los participantes. La realización de este proyecto servirá de preparación para considerar la participación en alguna de las próximas competencias.

En cuanto a las proyecciones comerciales, existen compañías como “Emotiv Systems” y “Neurosky” que se encuentran desarrollando productos de software y hardware para BCI, y que están interesadas en resolver problemas como el que se tratará en este proyecto. Además, compañías fuertemente comprometidas con la interactividad como Apple y Google, y líderes del mercado de los videojuegos como Microsoft, Nintendo y Sony, podrían estar dispuestas a incorporar aplicaciones de BCI en su portafolio de servicios.

En el contexto colombiano, este proyecto marcará un punto de referencia de investigación en esta tendencia, y servirá de invitación para el desarrollo de investigaciones similares, de modo que la Pontificia Universidad Javeriana e investigadores de nuestro país continúen haciendo aportes a esta prometedora aplicación.

En la Figura 1 se muestra el diagrama de bloques que representa uno de los procedimientos a seguir en problemas de clasificación de señales cerebrales: adquisición, extracción de características, mejoramiento del espacio de características (opcional) y clasificación.

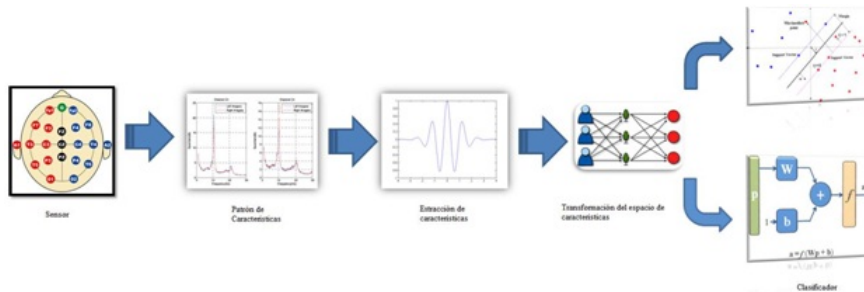


Figura 1: Esquema descripción del problema. Tomada [37],[9], Matlab Help, [38], [27].

Dado que la actividad cerebral relacionada con la imaginación de movimientos (MI) se presenta principalmente en las bandas de frecuencia μ (8-12 Hz) y β (18-25 Hz), y que las aplicaciones finales de BCI precisan de la implementación de los clasificadores en tiempo real, se requiere tanto buena resolución en tiempo como en frecuencia.

Entre las soluciones más efectivas para este problema se encuentran los métodos basados en wavelets, pues a diferencia de los enfoques por Transformada de Fourier, gozan de la propiedad de “multi-resolución”. Publicaciones como [9], [10], [11], [12] han comprobado el buen desempeño en la clasificación MI al realizar la extracción de características utilizando la Transformada Wavelet.

Por otro lado, también resulta conveniente usar métodos para extraer información temporal de las señales. Uno de los más utilizados para la predicción y detección de eventos es el análisis de caoticidad por Lyapunov, en el que las series de tiempo son representadas como trayectorias en el espacio de fase, y su caoticidad puede ser cuantificada por la distancia entre las mismas.

En cuanto al vector de características, como no existen criterios fijos para su construcción, algunos de sus componentes pueden resultar correlacionados. La aplicación de transformaciones como PCA (Principal Component Analysis) o ICA (Independent Component Analysis) podrían llegar a mejorar el espacio de características al reducir su dimensión o la dependencia entre sus componentes.

Algunas publicaciones se dedican a comparar el desempeño de ciertos clasificadores para exactamente el mismo problema. Entre las comparaciones que sirvieron de orientación se encuentran: Lineal vs NN-HMM [3], LDA vs SVM [4], [5], [6], LDA-QDA-Fisher y SVM-MLP-kNN. Teniendo en cuenta los resultados presentados en publicaciones como estas y en [11] y [13], se escogieron los clasificadores que presentaron los mejores resultados: Máquinas de Vectores de Soporte (SVM), Análisis Lineal Discriminante (LDA) y Redes Neuronales Artificiales (ANN).

El propósito de este trabajo es comparar el desempeño de tres estrategias de clasificación para el reconocimiento de patrones de señales cerebrales asociadas a la imaginación de movimientos de la mano izquierda (clase 1) y la mano derecha (clase 2), y evaluar el desempeño al aplicar PCA o ICA para mejorar el espacio de características o reducir la redundancia de las señales originales.

2. MARCO TEÓRICO

2.1. SEÑALES CEREBRALES

La actividad eléctrica cerebral puede ser registrada e interpretada como una serie de tiempo estocástica no estacionaria, en la que la información asociada a los procesos mentales que la generan (e.g. imaginación de movimientos) no se puede observar de forma directa, sino que debe ser caracterizada por medio de herramientas como la Transformada Wavelet, y clasificada usando métodos como las Máquinas de Vectores de Soporte (SVM) o las Redes Neuronales Artificiales (ANN).

2.1.1. Adquisición

La adquisición de la actividad cerebral es un procedimiento delicado tanto en el caso invasivo como en el no invasivo. En el primero, se debe ser extremadamente cuidadoso para que los instrumentos utilizados no causen lesiones en el tejido nervioso manipulado. En el segundo, la calidad del procedimiento está determinada por factores como la relación señal a ruido y la sensibilidad de los sensores empleados.

La gran mayoría de dispositivos BCI son diseñados para operar de forma no invasiva, por lo que el método de adquisición más utilizado es el Electroencefalograma (EEG). Este consiste en ubicar un conjunto de electrodos (sensores) en posiciones predefinidas de la corteza cerebral para capturar las señales eléctricas generadas sobre regiones determinadas del cerebro. En la Figura 2 se muestra el Sistema Internacional 10-20 de Ubicación de Electroodos.

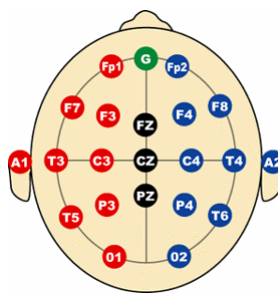


Figura 2: Sistema Internacional 10-20 ubicación de electrodos. Tomado de [37].

En aplicaciones BCI, se utilizan términos específicos para hacer referencia a distintos elementos del proceso de adquisición y registro de la actividad cerebral:

Definición 2.1. Un **canal** es el resultado de ubicar un electrodo en la posición indicada por un sistema como el sistema 10-20. De este modo, al hacer referencia al canal C3 en realidad se está hablando del electrodo ubicado en la posición C3 de un sistema dado.

Definición 2.2. Una **muestra** es un número que representa la actividad eléctrica cerebral en un instante de tiempo.

Definición 2.3. Un **ensayo** es un conjunto de muestras registradas consecutivamente durante un intervalo de tiempo dado.

A continuación se presentarán herramientas que se utilizarán a lo largo de todo el documento.

2.2. ANÁLISIS DE SEÑALES

2.2.1. Análisis en Frecuencia

La transformada de Fourier brinda información sobre las componentes de frecuencia de una señal. Sin embargo, esta información no se encuentra relacionada de forma explícita con la información temporal. Por ejemplo, para una señal no estacionaria no se tiene información sobre el tiempo en el cual se presenta o no una determinada componente de frecuencia [17].

Definición 2.4. Una **ventana temporal** $g(t)$ es una señal de energía finita que se encuentra concentrada en un intervalo acotado de tiempo [17].

Definición 2.5. Un **átomo tiempo-frecuencia** $g_{\tau,\xi}(t)$ se define como $g_{\tau,\xi}(t) = g_{\tau}(t)e^{j\xi t}$, donde $g_{\tau}(t)$ es una ventana temporal con centro en τ , y $\xi \in \mathfrak{R}$ [17].

Para analizar señales no estacionarias es necesario utilizar átomos tiempo-frecuencia con diferentes soportes en tiempo de acuerdo con la frecuencia que se vaya a analizar. Con este método se puede encontrar la transformada de Fourier de tiempo corto de una señal $x(t) \in L_2$. Sin embargo, dado que con esta no es posible tener una buena resolución en tiempo y en frecuencia simultáneamente resulta conveniente utilizar otras técnicas.

Definición 2.6. Una **wavelet** es una función $\psi \in L_2(\mathfrak{R})$ (energía finita) con media cero que satisface [18]:

$$\int_{-\infty}^{\infty} \psi(t) dt = 0 \quad (1)$$

Es normalizada $\|\psi\| = 1$, y centrada en la vecindad de $t = 0$. Una familia de wavelets se obtiene escalando ψ por s y trasladándola en u :

$$\psi_{u,s}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-u}{s}\right) \quad (2)$$

Estos átomos permanecen normalizados: $\|\psi(u, s)\| = 1$.

La principal característica de las wavelets es su localización tiempo-frecuencia, por lo que la mayoría de su energía se encuentra concentrada en un intervalo finito de tiempo. Esta característica hace posible que el análisis wavelet tenga buena resolución en frecuencias bajas (ventanas grandes en tiempo) y en frecuencias altas (ventanas pequeñas en tiempo) simultáneamente [19].

La transformada wavelet descompone señales en wavelets comprimidas y expandidas para analizar sus características tiempo-frecuencia, y goza de la propiedad de multiresolución.

Definición 2.7. La **transformada wavelet** de una señal $x(t) \in L_2(\mathfrak{R})$ en el tiempo u y escala s está dada por [18]:

$$W_x(u, s) = \langle x(t), \psi_{u,s}(t) \rangle = \frac{1}{\sqrt{s}} \int_{-\infty}^{\infty} x(t) \psi^*\left(\frac{t-u}{s}\right) dt \quad (3)$$

Para cada par de valores $(u, s) \in \mathfrak{R} \times \mathfrak{R}^+$, $W_x(u, s)$ almacena el resultado de la comparación entre $x(t)$ y $\psi_{u,s}(t)$ [17]. La descomposición de la señal da lugar a un conjunto de coeficientes llamados coeficientes wavelet. De este modo, la señal se puede reconstruir como una combinación lineal de funciones wavelet, teniendo en cuenta que para obtener una reconstrucción exacta de la señal se debe calcular el número adecuado de coeficientes.

Definición 2.8. Sea $f(t)$ una señal de tiempo continuo que es muestreada uniformemente y $f[n] = f(n)$ la señal discreta de tamaño N . Sea $\psi(t)$ una wavelet cuyo soporte se encuentra en $[-K/2, K/2]$. Para $2 \leq a^j \leq NK^{-1}$ una **wavelet discreta** escalada por a^j se define [18]:

$$\psi_j[n] = \frac{1}{\sqrt{a^j}} \psi\left(\frac{n}{a^j}\right) \quad (4)$$

La wavelet discreta tiene Ka^j valores diferentes de cero en $[-N/2, N/2]$. Para evitar problemas de frontera, se consideran $f[n]$ y $\psi_j[n]$ como señales periódicas de periodo N .

Definición 2.9. La **transformada wavelet discreta** puede ser escrita como una convolución circular $\bar{\psi}_j[n] = \psi_j^*[-n]$ [18]:

$$W_f[n, a^j] = \sum_{m=0}^{N-1} f[m] \bar{\psi}_j^*[m-n] \quad (5)$$

2.2.2. Análisis no Lineal en Tiempo

Serie de Tiempo

Una serie de tiempo es una sucesión temporal ordenada de observaciones tomadas a lo largo del tiempo.

Estacionariedad

Si se tiene una serie de tiempo de valores discretos $\{x_1, x_2, \dots, x_N\}$, estacionariedad significa que la función de distribución de probabilidad conjunta $f_{12}(x_1, x_2)$ depende únicamente de la diferencia temporal $|t_1 - t_2|$ y no de los valores absolutos t_1 y t_2 . De forma más clara, estacionariedad significa que las características de la serie de tiempo, tal como media, varianza o espectro de potencia, no cambian con el tiempo [20], [21].

2.2.3. Métodos de Análisis no Lineal-Exponentes de Lyapunov

Espacio de fase

El espacio de fase es un espacio en el que se puede describir el comportamiento dinámico de un sistema, donde las coordenadas espaciales representan las variables de estado, y la evolución del sistema se puede analizar a partir de trayectorias que dependen de las condiciones iniciales [21].

Uno de los enfoques para construir el espacio de fase es el método de los retardos, el cual consiste en calcular el retardo τ y la dimensión de embebimiento m que permiten construir la matriz [22]

$$\mathbf{X}(t) = \{\mathbf{x}(t), \mathbf{x}(t + \tau), \dots, \mathbf{x}(t + (m - 1)\tau)\}^T \quad (6)$$

Exponentes de Lyapunov

Definición 2.10. Dado un sistema dinámico continuo en un espacio de fase de n dimensiones, se observa la evolución a largo plazo de una n -esfera infinitesimal de condiciones iniciales; la esfera se convertirá en una n -elipsoide debido a la naturaleza de deformación del flujo. El **i -ésimo exponente de Lyapunov de una dimensión** λ_i se define en términos de la longitud del eje principal de la elipsoide $p_i(t)$ [23]

$$\lambda_i = \lim_{t \rightarrow \infty} \frac{1}{t} \log_2 \frac{p_i(t)}{p_i(0)} \quad (7)$$

Los λ_i se encuentran ordenados de mayor a menor. Como la orientación de una elipsoide cambia continuamente a medida que esta va evolucionando, las direcciones asociadas con un exponente dado varían alrededor del atractor, por lo que no se puede hablar de una dirección bien definida asociada a un exponente dado [24]. Esto implica que para un sistema dinámico, la sensibilidad a las condiciones iniciales está cuantificada por los exponentes de Lyapunov. Por ejemplo, si se consideran dos trayectorias con condiciones iniciales cercanas a un atractor múltiple, cuando el atractor es caótico las trayectorias divergen en promedio a una tasa exponencial caracterizada por el exponente de Lyapunov más grande.

Este concepto se puede generalizar para el espectro de los exponentes de Lyapunov, λ_i ($i = 1, \dots, n$), al considerar una pequeña esfera de condiciones iniciales de n dimensiones, donde n es el número de ecuaciones (ó el número de variables de estado) utilizadas para describir el sistema. Como se mencionó anteriormente, a medida que el sistema evoluciona, la esfera se convierte en un elipsoide cuyos ejes principales se expanden o se contraen en tasas dadas por los exponentes de Lyapunov.

La presencia de un exponente positivo es suficiente para diagnosticar caos y representa la inestabilidad local en una dirección particular. Nótese que para la existencia de un atractor, la dinámica global debe ser disipativa, es decir, globalmente estable, y la tasa total de contracción debe sobrepesar la tasa total de expansión. Luego, incluso cuando hay muchos exponentes de Lyapunov positivos, la suma sobre todo el espectro es negativa.

Las magnitudes de los exponentes de Lyapunov cuantifican la dinámica de un atractor en términos de teoría de la información. Los exponentes miden la tasa a la cual el sistema crea o destruye información; por lo que los exponentes se expresan en bits de información por segundo, ó bits por órbita para un sistema continuo y bits por iteración para un sistema discreto. La tasa promedio a la cual la información contenida en transientes se pierde puede ser determinada por los exponentes negativos. El decremento asintótico de una perturbación del atractor es gobernada por el exponente menos negativo, razón por la cual es el exponente más fácil de estimar [24].

Métodos para Estimar los Exponentes de Lyapunov

Algoritmo de Wolf et al. [24]: Dada una serie de tiempo $x(t)$, se reconstruye el espacio de estados por medio de coordenadas con retardos. Se localiza el vecino más cercano (por medio de la distancia euclídeana) del punto

$$\mathbf{X}(t) = \{\mathbf{x}(t), \mathbf{x}(t + \tau), \dots, \mathbf{x}[t + (m - 1)\tau]\}^T \quad (8)$$

y se denota la distancia entre estos dos puntos como $L(t_0)$. En un tiempo posterior t_1 , la longitud inicial evoluciona a la longitud $L'(t_1)$ y el elemento de longitud se propaga a través del atractor para un tiempo lo suficientemente corto para que solo una pequeña parte de la estructura del atractor sea examinada. Si la evolución de tiempo es muy grande, se puede observar que L' se comprime a medida que las dos trayectorias que lo definen pasan a través de una región de plegamiento del atractor, lo cual llevaría a una mala estimación de λ_1 . Ahora se busca por un nuevo punto que satisfaga dos criterios: que su separación $L(t_1)$ desde el punto de evolución de referencia sea pequeña, y que la separación angular entre los elementos evolucionados y de reemplazo sea pequeña. (Ver Figura 3). Si no se puede encontrar un punto de reemplazo adecuado, se retienen los puntos que se han utilizado. Este procedimiento se repite hasta que las trayectorias hayan atravesado todo el conjunto de datos, y en este momento se estima

$$\lambda_1 = \frac{1}{t_M - t_0} \sum_{k=1}^M \log_2 \frac{L'(t_k)}{L(t_{k-1})} \quad (9)$$

donde M corresponde al número total de reemplazos [24].

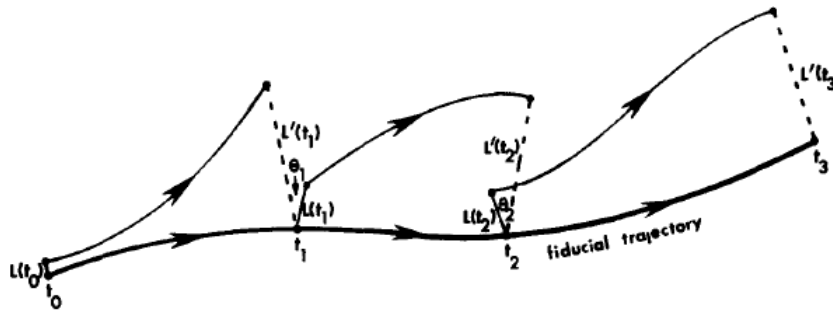


Figura 3: Representación esquemática de la evolución y procedimiento de reemplazo utilizado para estimar el mayor exponente de Lyapunov a partir de una serie de tiempo. Tomada de [24].

2.2.4. Análisis Estadístico y Transformaciones Lineales

Common Spatial Patterns (CSP), Patrones Espaciales Comunes

Partiendo del hecho que la generación de la actividad cerebral relacionada con el movimiento es contralateral, resulta conveniente utilizar una estrategia que aproveche tanto la información en frecuencia como la localización espacial de las señales EEG.

Una de las mayores ventajas del electroencefalograma reside en la información espacial que se agrega a los datos adquiridos con configuraciones multicanal. La existencia de información *a priori* que relaciona ciertas áreas de la corteza cerebral con acciones específicas como la imaginación de movimientos justifica la utilización de dicha información espacial para el procesamiento de las señales EEG. Esta se puede incluir en formas que van desde criterios de prioridad para la selección de canales hasta representaciones matriciales que consideran todas las series de tiempo como un conjunto.

La representación matricial ofrece numerosas ventajas para el procesamiento de señales EEG multicanal, entre las cuales se encuentra la posibilidad de encontrar transformaciones lineales que resalten características de las señales observadas, como la varianza temporal y la independencia estadística.

Basándose en los cambios de frecuencia y de energía de un canal EEG durante la imaginación de un movimiento, es posible suponer que dichos cambios a su vez implican cambios en la varianza de la serie de tiempo correspondiente, por lo que dicha medida estadística podría tenerse en cuenta como una característica para clasificación.

Una de las técnicas que aprovecha la representación matricial de las señales EEG multicanal es la técnica de “Common Spatial Patterns” (CSP), pues al aplicar ciertas transformaciones lineales busca generar una matriz cuyas varianzas sean más apropiadas que las de la matriz original para la discriminación de las clases del experimento [25].

Para encontrar la transformación lineal deseada, se deben calcular C filtros a partir de las matrices de covarianza \mathbf{R}_C , donde C es el número de clases y n_c es el número de canales.

$$\mathbf{R}_c = \sum_{i \in c} \frac{\mathbf{R}_{ci}}{n_c} \quad (10)$$

La matriz de covarianza \mathbf{R}_{ci} de cada ensayo \mathbf{X}_i , se calcula como [26]:

$$\mathbf{R}_{ci} = \frac{\mathbf{X}_i \mathbf{X}_i^T}{\text{traza}(\mathbf{X}_i \mathbf{X}_i^T)}, \quad i = 1, \dots, n_c \quad (11)$$

Luego de encontrar la matriz de whitening \mathbf{W} a partir de la matriz de covarianza compuesta \mathbf{R} , y las descomposiciones propias de las matrices de covarianza de cada clase (valores propios Λ y vectores propios \mathbf{U}_0), se calculan los filtros espaciales $\mathbf{S}\mathbf{F}_c$ como:

$$\mathbf{S}\mathbf{F}_c = \mathbf{U}_c^T \mathbf{W} \quad (12)$$

Donde, $\mathbf{W} = \Lambda^{-1/2} \mathbf{U}_0^T$, $\mathbf{R} = \sum_{i=1}^C \mathbf{R}_c$ con descomposición propia $\mathbf{R} = \mathbf{U}_0 \Lambda \mathbf{U}_0^T$, y \mathbf{U}_c es la matriz formada por los vectores propios de $\mathbf{W}\mathbf{R}_c\mathbf{W}^T$.

Principal Component Analysis (PCA), Análisis de Componentes Principales

El análisis de componentes es un enfoque no supervisado para encontrar características que sean estadísticamente más apropiadas. El análisis de componentes principales (PCA) es un procedimiento matemático que usa una transformación ortogonal para convertir un conjunto de variables correlacionadas a un conjunto de variables no correlacionadas llamadas componentes principales. Esta transformación de los datos a un nuevo sistema de coordenadas contiene, al proyectar cada uno de los datos a los componentes principales, las varianzas más grandes. El número de los componentes principales es menor o igual al número de variables originales, haciendo posible reducir la dimensión del espacio [27], [28].

En reconocimiento de patrones, los datos se representan con una matriz \mathbf{X} en donde cada columna es un ensayo. La tarea consiste en encontrar una matriz ortonormal \mathbf{P} tal que $\mathbf{Y} = \mathbf{P}\mathbf{X}$ y la matriz de covarianza de \mathbf{Y} sea diagonal. Las filas de \mathbf{P} corresponden a los componentes principales, es decir, una base alternativa para los datos y \mathbf{Y} son los datos expresados en términos de la base alternativa. Si la matriz de covarianza de \mathbf{Y} es diagonal, significa que no existe redundancia entre las diferentes dimensiones de \mathbf{Y} . Convenientemente, la cantidad de varianza mostrada por el i -ésimo componente principal es igual a la i -ésima entrada diagonal de la matriz de covarianza de \mathbf{Y} . Ordenando los componentes principales por la varianza, se obtiene una forma de clasificarlos con respecto a su importancia [29].

Independent Component Analysis (ICA), Análisis de Componentes Independientes

El método de ICA es útil cuando se desea estimar un vector de variables aleatorias latentes \mathbf{S} (que no pueden ser observadas directamente) a partir de un vector de variables aleatorias disponibles \mathbf{X} , que se pueden expresar por medio de una transformación lineal dada por:

$$\mathbf{X} = \mathbf{A}\mathbf{S} \quad (13)$$

donde \mathbf{A} es una matriz de mezcla cuadrada y los componentes de \mathbf{S} se asumen no gaussianos. El objetivo es estimar una matriz $\mathbf{W} = \mathbf{A}^{-1}$ cuyos componentes sean maximamente independientes, tal que [30]:

$$\mathbf{S} = \mathbf{W}\mathbf{X} \quad (14)$$

Cuando se aplica ICA en análisis de señales cerebrales, se asume que los componentes relacionados con la actividad son independientes de los componentes no relacionados y están mezclados linealmente para formar las señales observadas. El número de componentes es menor o igual al número de canales. Si existe alguna correlación entre los datos, debe realizarse preprocesamiento llamado “whitening” para remover esta correlación (transformando los componentes linealmente para alcanzar varianza unitaria) antes de aplicar ICA [31].

Definición 2.11. *Dos variables aleatorias y_1 y y_2 son independientes si la información en los valores de y_1 no brinda información alguna sobre los valores de y_2 y viceversa. Técnicamente la independencia puede ser definida por las densidades de probabilidad. Sea $p(y_1, y_2)$ la función de densidad de probabilidad conjunta de y_1 y y_2 , luego la función de probabilidad de densidad marginal de y_1 puede ser considerada como [30]:*

$$p_1(y_1) = \int p(y_1, y_2) dy_2 \quad (15)$$

Igualmente para y_2 . Entonces se define que y_1 y y_2 son independientes si y solo si la función de densidad de probabilidad conjunta es factorizable de la forma:

$$p(y_1, y_2) = p(y_1)p(y_2) \quad (16)$$

Esta definición naturalmente se extiende para cualquier número de n variables aleatorias, en cuyo caso la función de densidad de probabilidad conjunta será la multiplicación de n términos.

Por otro lado, se puede derivar una propiedad importante de las variables aleatorias independientes. Dadas 2 funciones, h_1 y h_2 siempre se tendrá:

$$E\{h_1(y_1)h_2(y_2)\} = E\{h_1(y_1)\}E\{h_2(y_2)\} \quad (17)$$

La restricción de la no gaussianidad en los componentes de la matriz \mathbf{S} se debe a que una función de densidad conjunta de vectores con distribuciones gaussianas no ofrece información sobre las direcciones de la matriz de mezcla \mathbf{A} .

2.3. RECONOCIMIENTO ESTADÍSTICO DE PATRONES

2.3.1. Espacio de características

El espacio de características es un espacio en donde se encuentran los vectores que representan a cada uno de los ensayos de un problema dado. La calidad de un espacio de características radica en la separabilidad entre sus clases, y no en su dimensión.

2.3.2. Métodos paramétricos

En los métodos paramétricos se asume que la forma de las funciones de densidad son conocidas [27].

2.3.3. Métodos no paramétricos

En la mayoría de aplicaciones de reconocimiento de patrones no se puede asumir que se conoce la forma de las funciones de densidad, ya que las formas paramétricas comunes no siempre se acomodan a las densidades encontradas en la práctica. Por lo tanto, resulta necesario utilizar técnicas de extracción de características que permitan describir los datos de forma compacta [27].

2.3.4. Aprendizaje no supervisado

El aprendizaje no supervisado busca extraer información de muestras no etiquetadas. Si la distribución proviene de una mezcla de densidades de probabilidad descritas por un conjunto de parámetros desconocidos Θ , estos parámetros pueden ser estimados por medio de métodos Bayesianos o de máxima verosimilitud. Un enfoque más general se tiene al definir alguna medida de similitud entre dos clusters, así como un criterio global tal como suma de error cuadrático o traza de una matriz de dispersión [27].

2.3.5. Aprendizaje supervisado

En el aprendizaje supervisado las muestras de entrenamiento utilizadas para diseñar el clasificador están etiquetadas con respecto a la clase a la cual pertenecen.

Linear Discriminant Analysis (LDA), Análisis Lineal Discriminante

Una función discriminante que es una combinación lineal de los componentes de $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$ puede ser escrita como [27]:

$$g(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0 \quad (18)$$

donde $\mathbf{w} = \{w_1, w_2, \dots, w_n\}$ es el vector de pesos y w_0 es el *offset*. La función discriminante $g(\mathbf{x})$ da una medida algebraica de la distancia entre \mathbf{x} y el hiperplano. Un clasificador lineal de dos categorías implementa la siguiente regla de decisión: Decide w_1 si $g(\mathbf{x}) > 0$ y w_2 si $g(\mathbf{x}) < 0$. Luego, \mathbf{x} es asignado a w_1 si el producto interno $\mathbf{w}^T \mathbf{x}$ excede el umbral $-w_0$ y w_2 en caso contrario. Si $g(\mathbf{x}) = 0$, \mathbf{x} puede ser asignado a cualquier clase.

La ecuación $g(\mathbf{x}) = 0$ define la superficie de decisión que separa los puntos asignados a w_1 de los puntos asignados a w_2 . Cuando $g(\mathbf{x})$ es lineal, la superficie de decisión es un hiperplano. En general, el hiperplano H divide el espacio de características en dos semiespacios: la región de decisión R_1 para w_1 y la región R_2 para w_2 .

Resumiendo, una función lineal discriminante divide el espacio de características por medio de una superficie de decisión que en este caso es un hiperplano. La orientación de la superficie es determinada por el vector normal \mathbf{w} y su localización es determinada por el bias w_0 .

Supóngase que se tiene un conjunto de n muestras $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n$, algunas etiquetadas con w_1 y otras con w_2 . Se debe determinar los pesos \mathbf{a} en una función discriminante lineal $g(\mathbf{x}) = \mathbf{a}^T \mathbf{y}$ a partir de estas muestras, es decir, se debe buscar un vector de pesos que clasifique todas las muestras correctamente. Si tal vector de pesos existe, se dice que las muestras son linealmente separables.

Una muestra \mathbf{y}_i es clasificada correctamente si $\mathbf{a}^T \mathbf{y}_i > 0$ y \mathbf{y}_i está etiquetada con w_1 , o si $\mathbf{a}^T \mathbf{y}_i < 0$ y \mathbf{y}_i está etiquetada con w_2 . Esto sugiere una normalización que simplifica el tratamiento para el caso de dos clases al reemplazar todas las muestras etiquetadas con w_2 por sus negativos. De esta manera, se pueden olvidar las etiquetas y buscar un vector de pesos \mathbf{a} tal que $\mathbf{a}^T \mathbf{y}_i > 0$ para todas las muestras. Tal vector de pesos es llamado vector de separación o vector solución.

Support Vector Machines (SVM), Máquinas de Vectores de Soporte

Las máquinas de vectores de soporte se encargan de preprocesar los datos para representar los patrones en un espacio de mayor dimensión con respecto al espacio original de características. Con un mapeo no lineal

apropiado $\varphi(\cdot)$ hacia una dimensión lo suficientemente grande, los datos provenientes de dos clases ω_1 y ω_2 pueden ser separados por un hiperplano. Se asume que cada patrón \mathbf{x}_k ha sido transformado a $\mathbf{y}_k = \varphi(\mathbf{x}_k)$. Para cada uno de los n patrones, $k = 1, \dots, n$ se tiene $z_k = \pm 1$, dependiendo si $k \in \omega_1$ ó $k \in \omega_2$. Un discriminante lineal $g(\mathbf{y})$ en el espacio de mayor dimensión \mathbf{y} está dado por [27]:

$$g(\mathbf{y}) = \mathbf{a}^T \mathbf{y} \quad (19)$$

En donde el vector de pesos y el vector de patrones se encuentran en el espacio \mathbf{y} , con $a_0 \omega_0$ y $y_0 = 1$, respectivamente. Luego, el hiperplano separador asegura:

$$z_k g(\mathbf{y}_k) \geq 1 \quad k = 1, \dots, n \quad (20)$$

El objetivo de entrenar una máquina de vectores de soporte es encontrar el hiperplano separador con el margen más grande; esperando que entre más grande el margen, mayor la generalización del clasificador. La distancia del hiperplano a cualquier patrón \mathbf{y} es $|g(\mathbf{y})|/||\mathbf{a}||$. Asumiendo que existe un margen positivo b , se tiene:

$$\frac{z_k g(\mathbf{y}_k)}{||\mathbf{a}||} \geq b \quad k = 1, \dots, n \quad (21)$$

Se debe encontrar el vector de pesos \mathbf{a} que maximiza b . Como el vector solución puede ser escalado arbitrariamente preservando el hiperplano, se imponen las condiciones que $b||\mathbf{a}|| = 1$ y que $||\mathbf{a}||^2$ sea máximo para asegurar unicidad.

Los vectores de soporte representan los patrones para los cuales $z_k g(\mathbf{y}_k) = 1$. Estos corresponden a las muestras de entrenamiento que definen el hiperplano óptimo de separación y son los patrones más difíciles de clasificar.

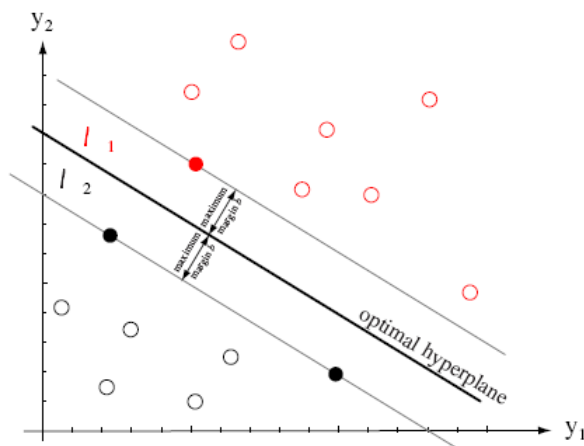


Figura 4: Entrenar una máquina de vectores de soporte consiste en encontrar el hiperplano óptimo, es decir, aquel con la máxima distancia desde los patrones de entrenamiento más cercanos. Los vectores de soporte corresponden a estos patrones más cercanos, a una distancia b del hiperplano. Se muestran tres vectores de soporte en la figura. Tomada de [27].

Artificial Neural Networks (ANN), Redes Neuronales Artificiales

Una red neuronal artificial está compuesta por una capa de entrada, capas ocultas y una capa de salida, interconectadas por medio de pesos modificables y compuestas por neuronas. En reconocimiento de patrones, las neuronas de la capa de entrada reciben los componentes de un vector de características (que va a ser utilizado en entrenamiento o en clasificación) y las señales emitidas por las neuronas de la capa de salida

son funciones discriminantes usadas para clasificación [27].

Cada neurona j de la capa oculta computa su salida en función de una suma ponderada net_j de sus entradas, que puede ser expresada como el producto punto de su vector de pesos ω_j con la entrada \mathbf{x} :

$$net_j = \sum_{i=1}^d x_i \omega_{ji} + \omega_{j0} = \sum_{i=0}^d x_i \omega_{ji} \equiv \mathbf{w}_j^T \mathbf{x} \quad (22)$$

donde d es el tamaño de la capa de entrada. Luego, cada salida y_j es encontrada al aplicar net_j a la función no lineal de activación $f(\cdot)$:

$$y_j = f(net_j) \quad (23)$$

De forma similar, cada neurona k de la capa de salida computa su resultado a partir de su vector de pesos ω_k y del vector \mathbf{y} proveniente de la capa oculta como:

$$net_k = \sum_{j=1}^{n_H} y_j \omega_{kj} + \omega_{k0} = \sum_{j=0}^{n_H} y_j \omega_{kj} = \mathbf{w}_k^T \mathbf{y} \quad (24)$$

donde n_H denota el número de neuronas de la capa oculta. Luego, cada salida z_k es encontrada al aplicar net_k a la función no lineal de activación $g(\cdot)$:

$$z_k = g(net_k) \quad (25)$$

Finalmente, la decisión de clasificación consiste en asignar el patrón de entrada a la clase k que obtuvo el mayor valor de z_k .

3. ESPECIFICACIONES

3.1. Diagrama de Bloques

El diagrama de bloques del sistema de reconocimiento de patrones implementado en este trabajo y su correspondiente explicación se muestra a continuación.

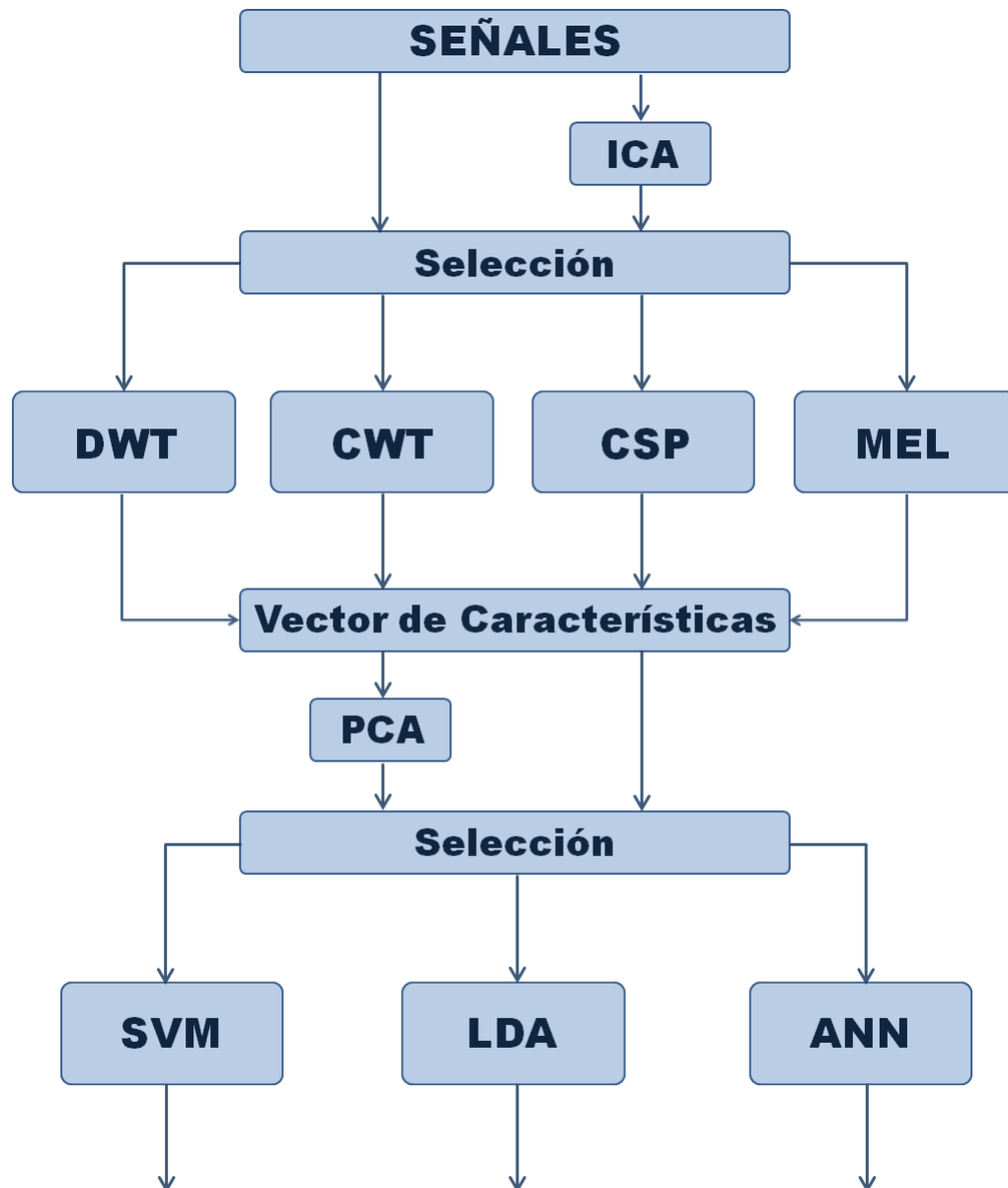


Figura 5: Diagrama de bloques del sistema.

- Señales: Corresponde al conjunto de señales EEG que serán pre-procesadas o que pasarán directamente al bloque de selección para el proceso de extracción de características.
- ICA: Pre-procesa el conjunto de señales con la técnica ICA.
- Selección 1: Escoge el conjunto de señales que será utilizado (pre-procesado o no) y los métodos de extracción de características.
- DWT: Método de extracción que calcula la potencia promedio y la energía de las señales de las descomposiciones DWT.
- CWT: Método de extracción que calcula el promedio de los coeficientes CWT sobre 8 ventanas de tiempo.
- CSP: Método de extracción que transforma los datos por medio de filtros espaciales.
- MEL: Método de extracción que calcula el máximo exponente de Lyapunov.
- Vector de características: Forma el vector de características agrupando las medidas extraídas por cada uno de los métodos. Este puede ser pre-procesado o pasar directamente al bloque de selección para el proceso de clasificación.
- PCA: Pre-procesa el vector de características con la técnica de PCA.
- Selección 2: Escoge el vector de características que será utilizado (pre-procesado o no) y los métodos de clasificación.
- SVM: Método de clasificación que crea, entrena y prueba la SVM a utilizar.
- ANN: Método de clasificación que crea, entrena y prueba la ANN a utilizar.
- LDA: Método de clasificación que implementa LDA.

3.2. MEL: Máximo Exponente de Lyapunov

El diagrama de bloques de la implementación propuesta para encontrar el máximo exponente de Lyapunov y su correspondiente explicación se muestra a continuación.

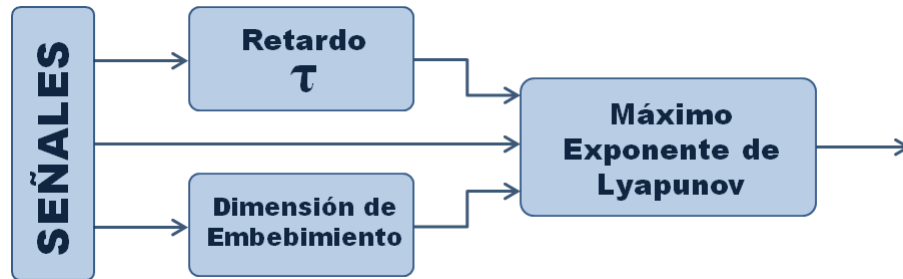


Figura 6: Diagrama de bloques del sistema.

- Señales: Corresponde a las señales a las que se les estimará el máximo exponente de Lyapunov.
- Retardo τ : Calcula el retardo de cada pareja ensayo-canal a partir de la función de autocorrelación de la señal.
- Dimensión de embebimiento: Calcula la dimensión de embebimiento utilizando la relación planteada por D. Kugiumtzis [33].
- Máximo exponente de Lyapunov: Usando el retardo y la dimensión de embebimiento reconstruye el espacio de fase y estima el máximo exponente de Lyapunov para cada canal.

4. DESARROLLOS

4.1. Descripción de las señales

Este proyecto se realizó con un conjunto de señales EEG descargadas del sitio web del grupo “Berlin Brain-Computer Interface” en Alemania, destinadas a la comunidad participante en la Competencia BCI 2003 (Data set III) [2]. En este se dispone de 140 ensayos de tres variables (canales C3, CZ, C4) con sus respectivas etiquetas, de los que se utilizaron 112 para entrenamiento y 28 para prueba, en 5 combinaciones de validación cruzada.

La actividad cerebral de los voluntarios fue registrada durante 9 segundos a una frecuencia de muestreo de 128 Hz (1152 muestras). La Figura 7 muestra la configuración en tiempo de cada ensayo: dos segundos después de iniciar el registro se presentó un estímulo acústico para indicar el inicio del ensayo, y se mostró una cruz ('+') durante 1 segundo; finalmente, entre el segundo 3 y el 9 se mostró una flecha hacia la izquierda (clase 1) o hacia la derecha (clase 2), mientras el usuario debía imaginar el movimiento de la mano correspondiente a la dirección indicada.

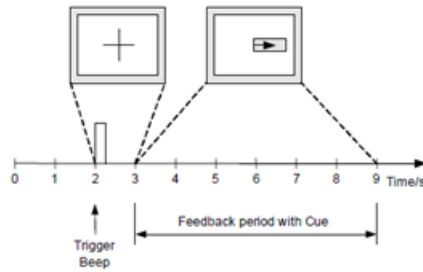


Figura 7: Configuración en tiempo de cada ensayo. Tomada de [9].

4.2. Métodos de extracción de características

4.2.1. Discrete Wavelet Transform (DWT), Transformada Discreta Wavelet

La DWT es una herramienta de gran utilidad para la caracterización espectral de una señal, pues al utilizar bancos de filtros y hacer descomposiciones en bandas de frecuencia ortogonales, se puede analizar el contenido de energía de una señal en cada una de ellas por separado.

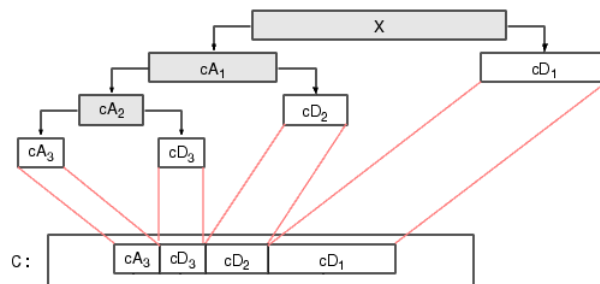


Figura 8: Descomposición Wavelet de 3 niveles. Tomada de Matlab Help

La Figura 8 muestra el esquema de descomposición wavelet de 3 niveles utilizado en este trabajo, del cual resultan un vector cA_3 con la señal de aproximación, y tres vectores cD_1 , cD_2 y cD_3 con las señales de detalle.

Para implementar este esquema de descomposición se utilizó la función 'wavedec' de Matlab con 3 niveles de profundidad y la wavelet 'Daubechies10'. Las bandas generadas por esta descomposición para la frecuencia de muestreo trabajada (128 Hz) se muestran en el Cuadro 1.

Descomposición	Rango de Frecuencia (Hz)	Nivel
D1	32-64	1
D2	16-32	2
D3	8-16	3
A3	0-8	3

Cuadro 1: Bandas generadas por la descomposición utilizada

La función 'wavedec' retorna un vector 'Coef' con las señales descompuestas en cada una de las bandas, y un vector 'Lint' con sus respectivas longitudes.

$$[Coef, Lint] = wavedec(EEGdata_DWT(:, c), 3, 'db10') \quad (26)$$

Algunas medidas de la señal en cada sub-banda de frecuencia pueden encontrarse a partir de las señales resultantes de la descomposición wavelet [9]. Las medidas seleccionadas para incluir en el vector de características de este trabajo fueron la potencia promedio y la energía.

■ *Potencia promedio de las descomposiciones DWT*

La potencia promedio de cada descomposición en cada canal se calculó como la norma al cuadrado del vector que las contenía, dividida por la longitud de dicho vector.

$$P_x = \frac{1}{L} \sum_{i=1}^L (x_i)^2 = \frac{1}{L} (norm(\mathbf{x}))^2 \quad (27)$$

Teniendo en cuenta que las bandas de interés para el problema son μ (8-13 Hz) y β (18-25 Hz), y que estas se encuentran contenidas en las descomposiciones D3 y D2, respectivamente, las cantidades incluidas en el vector de características fueron:

- C3- μ D3: Potencia promedio de la descomposición D3 del canal C3.
- C3- β D2: Potencia promedio de la descomposición D2 del canal C3.
- C4- μ D3: Potencia promedio de la descomposición D3 del canal C4.
- C4- β D2: Potencia promedio de la descomposición D2 del canal C4.

■ *Energía de las descomposiciones DWT*

La energía de las descomposiciones de aproximación se encontró utilizando la función 'wenergy' de Matlab. Los parámetros de la función 'wenergy' son el vector de descomposiciones 'Coef' y el vector de longitudes 'Lint' que retorna la función 'wavedec'.

$$[Ea3, Eb3] = wenergy(Coef_C3, L_C3) \quad (28)$$

$$[Ea4, Eb4] = wenergy(Coef_C4, L_C4) \quad (29)$$

Siguiendo un razonamiento similar al caso de la potencia promedio, las cantidades incluidas en el vector de características fueron:

- C3.en: Energía de las descomposiciones de aproximacion del canal C3.
- C4.en: Energía de las descomposiciones de aproximacion del canal C4.

4.2.2. Continuous Wavelet Transform (CWT), Transformada Continua Wavelet

La Transformada Continua Wavelet, al calcular la participación de distintas frecuencias en cada instante de tiempo de una señal dada, permite observar simultáneamente las características tiempo-frecuencia de la misma. La mencionada propiedad de multi-resolución se puede observar al graficar las matrices de coeficientes resultantes (e.g. Figura 9), en las que las escalas bajas (frecuencias altas) tienen mayor resolución temporal y las escalas altas mayor resolución frecuencial.

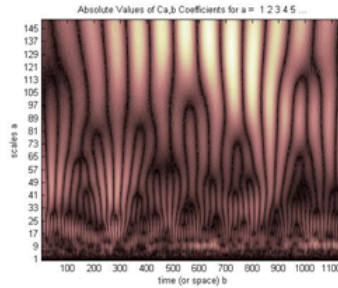


Figura 9: Transformada Continua Wavelet de un ensayo

Para implementar la CWT se utilizó la función 'cwt' de Matlab, que genera una matriz con los coeficientes CWT, la cual se puede graficar en tres dimensiones para visualizar una medida proporcional a la energía de una señal en cada pareja tiempo-escala del rango especificado.

A diferencia de la función 'wavedec', la función 'cwt' permite observar simultáneamente la información de tiempo y escala (frecuencia), por lo que es posible determinar visualmente las regiones del plano (u, t) donde se encuentra la mayor cantidad de energía de una señal dada.

Las Figuras 10 y 11 muestran la CWT para uno de los ensayos con etiqueta 1 (imaginación mano izquierda).

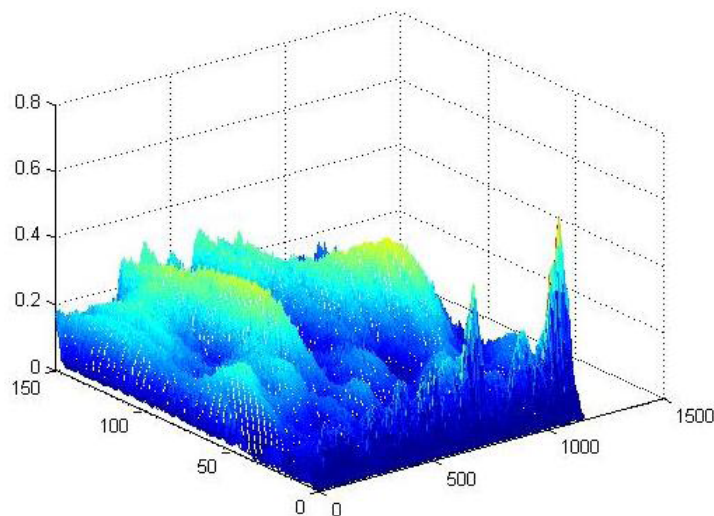


Figura 10: CWT del canal C3, Ensayo 81

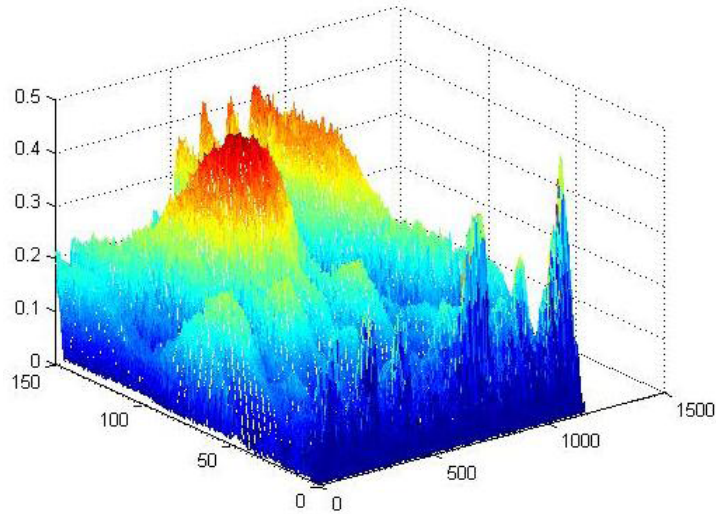


Figura 11: CWT del canal C4, Ensayo 81

Luego de probar esta función con distintos intervalos de escalas s , se determinó que el intervalo $[1, 150]$ era suficiente para analizar el contenido espectral de la señal, y que efectivamente existía información discriminante en la CWT.

- *Suma de los coeficientes CWT sobre regiones muestra/escala*

Como una medida de la cantidad de energía presente en una región tiempo-escala, se sumaron los coeficientes CWT de cada canal y cada ensayo desde 3 hasta 9 segundos y desde la escala 50 a la 150. Teniendo en cuenta la etiqueta de cada ensayo, se comparó la cantidad de cada canal con el canal contrario, esperando que fuera mayor la del canal contrario al lado de la imaginación del movimiento.

- *Promedio de los coeficientes sobre 8 ventanas de tiempo*

Publicaciones como [7] y [32] proponen el cálculo de cantidades sobre varias ventanas de tiempo en vez de una sola ventana para todo el ensayo. En particular, [32] propone utilizar el promedio de los coeficientes de la CWT, y sostiene que el número óptimo de ventanas para una SVM es 8. En este caso, las medidas escogidas para incluir en el vector de características fueron los máximos en escalas s sobre las medias del valor absoluto de las muestras en tiempo t_j de las ventanas W_j para cada canal C_i .

- C3_W3: Máximo de la ventana 3 del canal C3.
- C3_W4: Máximo de la ventana 4 del canal C3.
- C3_W5: Máximo de la ventana 5 del canal C3.
- C3_W6: Máximo de la ventana 6 del canal C3.
- C4_W3: Máximo de la ventana 3 del canal C4.
- C4_W4: Máximo de la ventana 4 del canal C4.
- C4_W5: Máximo de la ventana 5 del canal C4.
- C4_W6: Máximo de la ventana 6 del canal C4.

donde

$$C_{i-Wj} = \max_s(\text{mean}_t(|\text{Coefficients_CWT_trial}(s, t_j, C_i)|)), \quad t_j \in W_j \quad (30)$$

4.2.3. Common Spatial Patterns (CSP), Patrones Espaciales Comunes

Para implementar la técnica de CSP, se creó una función que construye los filtros espaciales a partir de las muestras de entrenamiento $EEGdata_{CSP}$ y las etiquetas $EEGlabels$:

$$[SpatialFilter_1, SpatialFilter_2] = Z_Function_Filters_CSP(EEGdata_CSP, EEGlabels) \quad (31)$$

Luego se creó una función que aplica las transformaciones a los datos originales X por medio de los filtros espaciales SF_c ($X'_c = SF_c X$) y se construye el vector de características a partir de la varianza de los canales.

$$Features_CSP_trial = Z_Function_Features_CSP(SpatialFilter_1, SpatialFilter_2, EEGdata_CSP(:, :, t)) \quad (32)$$

donde las características para cada filtro c y canal i del ensayo \mathbf{X}' son encontradas como:

$$FeaturesCSPtrial(\mathbf{x}_{ci}) = \log \left(\frac{var(\mathbf{x}'_{ci})}{\sum_{k=1}^K var(\mathbf{x}'_{ck})} \right) \quad (33)$$

Finalmente, las medidas incluidas en el vector de características fueron:

- C3_SF1: Varianza del canal C3 transformado por el filtro espacial 1.
- C4_SF2: Varianza del canal C4 transformado por el filtro espacial 2.

4.2.4. Máximo exponente de Lyapunov

Dado que las señales trabajadas provienen de un sistema inherentemente caótico (cerebro), estas deben ser analizadas con herramientas que permitan cuantificar dicho comportamiento. Una de estas es el máximo exponente de Lyapunov.

El primer paso para hallar el máximo exponente de Lyapunov es encontrar el retardo y la dimensión de embebimiento para reconstruir el espacio de fase [22]:

$$\mathbf{X}(t) = \{\mathbf{x}(t), \mathbf{x}(t + \tau), \dots, \mathbf{x}[t + (m - 1)\tau]\}^T \quad (34)$$

Para encontrar la matriz de retardos τ , se creó una función que calcula el retardo de cada ensayo-canal a partir de la función de autocorrelación de la señal, asignando a τ el índice del primer cruce por cero.

Para calcular la dimensión de embebimiento se utilizó la relación planteada por D. Kugiumtzis [33]

$$\tau_w = (m - 1)\tau \quad (35)$$

donde τ_w es igual a la media de tiempo entre picos ($mtbp$) [34], que se encontró como el inverso de la frecuencia fundamental de la serie de tiempo.

Finalmente, haciendo uso de la función *lyapvec* tomada de [21], modificada para suministrar la matriz de retardos y la dimensión de embebimiento, se encontró el máximo exponente de Lyapunov para cada canal, y se incluyeron en el vector de características:

- C3_MaxExp: Máximo exponente de Lyapunov del canal C3.
- C4_MaxExp: Máximo exponente de Lyapunov del canal C4.

4.3. Definición del vector de características

Tomando las cantidades presentadas en la sección 4.2, se procedió a construir el vector de características como se muestra en el Cuadro 2.

Posición	Canal_Medida	Método
1	C3_W3	Medias de los coeficientes CWT sobre 8 ventanas de tiempo w_i
2	C3_W4	
3	C4_W3	
4	C4_W4	
5	C3_W5	
6	C3_W6	
7	C4_W5	
8	C4_W6	
9	C3.en	Energía de los coeficientes DWT
10	C4.en	
11	C3- μ D3	Potencia promedio de los coeficientes DWT
12	C3- β D2	
13	C4- μ D3	
14	C4- β D2	
15	C3_SF1	Varianzas por CSP
16	C4_SF2	
17	C3_MaxExp	Lyapunov
18	C4_MaxExp	

Cuadro 2: Vector de características

4.4. PCA e ICA

Independent Component Analysis (ICA), Análisis de Componentes Independientes: Algoritmo FastICA

La regla de aprendizaje en FastICA encuentra una dirección, en este caso un vector unidad \mathbf{w} cuya proyección $\mathbf{w}^T \mathbf{x}$ maximiza la no gaussianidad de las variables latentes s_i . La no gaussianidad puede ser medida por la aproximación de la negentropía $J(\mathbf{w}^T \mathbf{x})$ o por derivaciones iterativas de aproximaciones de Newton para una función g dada.

La forma básica del algoritmo es:

1. Escoger un vector inicial de pesos \mathbf{w} .
2. Hacer $\mathbf{w}^+ = E\{\mathbf{x}g(\mathbf{w}^T \mathbf{x})\} - E\{g'(\mathbf{w}^T \mathbf{x})\}\mathbf{w}$.
3. Hacer $\mathbf{w} = \frac{\mathbf{w}^+}{\|\mathbf{w}^+\|}$.
4. Repetir desde 2 hasta que la diferencia entre los antiguos y nuevos valores en un punto de \mathbf{w} converja [30].

Para aplicar este algoritmo se usó la función 'fastica' de Matlab (versión 2.5). Para el preprocesamiento de las señales cerebrales se creó la función

$$[ICA_data] = Z_Function_Preprocessing_ICA(EEGdata_ICApr) \quad (36)$$

FastICA recibe la matriz de datos cuyas filas corresponden a los ensayos y retorna una matriz en cuyas filas se encuentran los componentes independientes. Los dos parámetros más importantes son el número máximo de iteraciones y la función de aproximación g (cúbica, tangente hiperbólica, gaussiana y cuadrática).

Principal Component Analysis (PCA): Análisis de Componentes Principales

Para implementar la técnica de preprocesamiento del espacio de características PCA se utilizó la función 'princomp' de Matlab. 'Princomp' recibe la matriz de características con los ensayos en las filas y retorna una matriz con los componentes principales.

$$[Coeff\ Features_All] = princomp(EEGdata_PC\ Apre_Features) \quad (37)$$

4.5. Métodos de Clasificación

4.5.1. Linear Discriminant Analysis (LDA), Análisis Lineal Discriminante

Para implementar el clasificador LDA, se separaron tanto las muestras como las etiquetas de entrenamiento de las de prueba, y se utilizaron las primeras para encontrar los parámetros del clasificador conforme muestran las siguientes ecuaciones:

Primero se encontró la media general de los datos a partir de las medias de las clases:

$$\mu = \frac{N_{izq}\mu_{izq} + N_{der}\mu_{der}}{N_{izq} + N_{der}} \quad (38)$$

donde μ_{izq} es la media de los vectores del espacio de características pertenecientes a la clase 1, μ_{der} la media de los pertenecientes a la clase 2, y N_{izq} y N_{der} el número de vectores de cada clase.

Luego se calculó la matriz de covarianza \mathbf{C} de los datos utilizando la función 'cov' de Matlab, y se procedió a calcular el vector de pesos \mathbf{w} y el bias \mathbf{w}_0 como:

$$\mathbf{w} = \mathbf{C}^{-1}(\mu_{der} - \mu_{izq})^T \quad (39)$$

$$\mathbf{w}_0 = -\mu\mathbf{w} \quad (40)$$

Después, se creó la función discriminante a partir de las cantidades anteriores para generar el hiperplano de separación en el espacio de características:

$$D(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + \mathbf{w}_0 \quad (41)$$

Finalmente, se utilizó dicha función discriminante con las muestras de prueba para formar el vector de salidas del clasificador LDA, y luego compararlo con las etiquetas de prueba y evaluar el desempeño del mismo.

4.5.2. Support Vector Machines (SVM), Máquinas de Vectores de Soporte

Para crear y entrenar la SVM se utilizó la función 'svmtrain' de Matlab, a la que se le proporcionaron la matriz de características (ensayos por características) y las etiquetas de entrenamiento.

$$SVMgraz = svmtrain(Data_Train_SVM, Labels_Train_SVM, 'KernelFunction', ...); \quad (42)$$

Para probar el desempeño del clasificador se utilizó la función 'svmclassify' de Matlab, ingresando como parámetros la SVM creada anteriormente y los datos de prueba.

$$SVMoutputlabels = svmclassify(SVMgraz, DataTest_SVM); \quad (43)$$

4.5.3. Artificial Neural Networks (ANN), Redes Neuronales Artificiales

Para crear la red neuronal *feed-forward back-propagation* se utilizó la función 'newff' de Matlab, ingresando como parámetros el conjunto completo de características junto a sus etiquetas.

$$ANNfunction = newff(minmax(Features_All_train'), [3 1], {'purelin', 'purelin'}); \quad (44)$$

Esta red se entrenó utilizando la función 'train' de Matlab limitando el número de épocas a 100, y se simuló utilizando la función 'sim' de Matlab, ingresando como parámetros la red neuronal creada y los datos de prueba.

$$ANNfunction = train(ANNfunction, Features_All_train', LabelsTrain'); \quad (45)$$

$$ANNoutputs = sim(ANNfunction, Features_All_test'); \quad (46)$$

Debido a que el resultado de una red neuronal puede variar para diferentes ejecuciones de un mismo conjunto de características, se optó por encontrar el resultado final como el promedio de varias ejecuciones.

5. ANÁLISIS DE RESULTADOS

Para analizar la calidad del espacio de características construido y el desempeño de los clasificadores propuestos se abordaron los siguientes enfoques:

1. Combinación de técnicas de extracción: Se probaron todas las combinaciones posibles del vector de características agrupado por técnicas de extracción, para un total de 15 combinaciones.
2. K-folds: Para estudiar la generalización de los clasificadores propuestos, se implementó una validación cruzada de orden 5, en donde todos los subconjuntos tienen el mismo tamaño.

Durante el desarrollo del proyecto se probaron distintas combinaciones de los parámetros de los clasificadores con el fin de encontrar los que mejores desempeños presentaran para este problema. Para SVM se probaron el kernel lineal, el polinomial y el RBF; para ANN se probaron arquitecturas de 3, 6 y 12 neuronas en la capa oculta y funciones de activación 'purelin', 'logsig' y 'tansig'.

Los siguientes cuadros muestran los porcentajes de clasificación para el subconjunto de prueba 3 y las características DWT con diferentes combinaciones de parámetros.

Kernel	Parámetro	Valor
lineal		92,86
Polinomial	Orden=2	78,57
	Orden=3	64,29
	Orden=4	60,71
RBF	$\sigma=0.1$	46,43
	$\sigma=1$	82,14
	$\sigma=10$	85,71

Cuadro 3: Parámetros SVM

Neurona CO	tansig-purelin	logsig - purelin	purelin - purelin
12	84.82	84.64	92.86
6	87.5	86.79	92.86
3	88.04	85.71	92.68

Cuadro 4: Parámetros ANN

Basados en los resultados de los cuadros 3 y 4, se escogió el kernel lineal para SVM y una arquitectura de 1 capa oculta con 3 neuronas para ANN, con funciones de activación 'purelin' tanto para la capa oculta como para la de salida. Los mejores resultados obtenidos se resumen en el cuadro 5:

Grupos de Resultados	Clasificación (%)		
	SVM	LDA	ANN
1	92.86	92.86	92.86
2	89.29	89.29	85.71
3	85.71	85.71	83.21
4	82.14	82.14	82.32
5	78.57	78.57	77.5
Promedio	85.714	85.714	84.32

Cuadro 5: Mejores Resultados

En la siguiente gráfica de barras se muestran los mejores resultados obtenidos para cada clasificador.

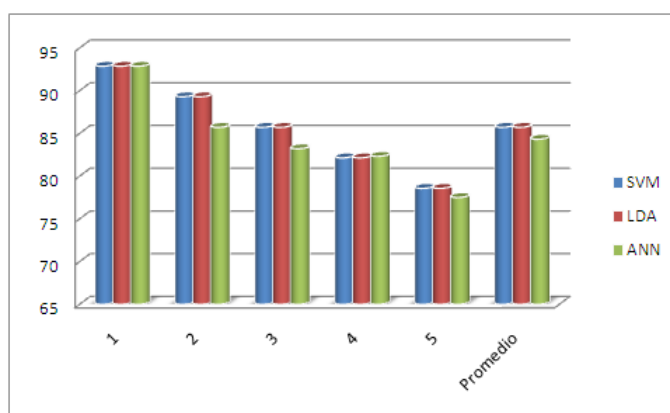


Figura 12: Mejores Resultados con PCA

5.1. Cuadros de resultados

Cada una de los cuadros de resultados se estructura de la siguiente manera:

- En la columna de características se indican con una 'x' las técnicas de extracción que participan en cada clasificación.
- En la columna de PCA se indica con una 'x' si se realizó preprocesamiento con esta técnica.
- En la columna de partición de prueba se muestran los resultados de clasificación para cada subconjunto de prueba.
- En la última columna se muestra el promedio de los resultados obtenidos para cada combinación de características.

A continuación, se muestra un cuadro de resultados para cada uno de los clasificadores propuestos.

Resultados SVM:

Características				Subconjuntos de Prueba					Promedio	
DWT	CWT	CSP	Lyapunov	PCA	1	2	3	4		5
		x			50	50	53.57	71.43	42.86	53.572
		x		x	50	50	57.14	53.57	50	52.142
x					71.43	60.71	92.86	75	71.43	74.286
x				x	71.43	64.29	92.86	75	71.43	75.002
			x		67.86	53.57	75	53.57	57.14	61.428
			x	x	67.86	53.57	75	53.57	53.57	60.714
	x				57.14	39.29	57.14	50	50	50.714
	x			x	67.86	53.57	75	53.57	53.57	60.714
x	x				78.57	64.29	78.57	71.43	75	73.527
x	x			x	82.14	64.29	78.57	71.43	71.43	73.572
x			x		67.86	60.71	92.86	71.43	71.43	72.858
x			x	x	64.29	60.71	92.86	71.43	71.43	72.144
x		x			78.57	57.14	85.71	71.43	67.86	72.142
x		x		x	78.57	57.14	89.29	71.43	67.86	72.858
		x	x		67.86	42.86	67.86	53.57	64.29	59.288
		x	x	x	67.86	42.86	67.86	57.14	64.29	60.002
	x	x			53.57	42.86	60.71	53.57	46.43	51.428
	x	x		x	53.57	42.86	53.57	53.57	42.86	49.286
x			x		64.29	53.57	60.71	60.71	57.14	59.284
x			x	x	64.29	53.57	60.71	64.29	57.14	60
x	x		x		75	60.71	78.57	75	71.43	72.142
x	x		x	x	71.43	60.71	78.57	75	71.43	71.428
x	x	x			78.57	57.14	71.43	71.43	71.43	70
x	x	x		x	75	57.14	71.43	71.43	71.43	69.286
x		x	x		71.43	53.57	85.71	71.43	67.86	70
x		x	x	x	71.43	50	85.71	71.43	67.86	69.286
	x	x	x		60.71	35.71	57.14	78.57	57.14	57.854
	x	x	x	x	60.71	35.71	57.14	78.57	57.14	57.854
x	x	x	x		67.82	50	71.43	75	71.43	67.136
x	x	x	x	x	64.29	53.57	71.43	75	67.86	66.43
Promedio					63.23	49	68.26	62.97	59.24	64.38
				x	63.23	50.12	69.38	62.52	59.01	64.71

Cuadro 6: Resultados SVM

El máximo porcentaje de clasificación obtenido por SVM fue 92,86 % para el subconjunto de prueba 3, utilizando las características de DWT y la combinación DWT-Lyapunov. El mejor porcentaje promedio de clasificación fue 75,002 %, utilizando las características de DWT y aplicando PCA para su preprocesamiento.

Resultados LDA:

Características				Subconjuntos de Prueba						
DWT	CWT	CSP	Lyapunov	PCA	1	2	3	4	5	Promedio
		x			53.57	53.57	57.14	71.43	50	57.142
		x		x	42.86	50	57.14	60.71	50	52.142
x					71.43	57.14	92.86	75	67.86	72.858
x				x	71.43	57.14	92.86	75	67.86	72.858
			x		71.43	60.71	71.43	57.14	46.43	61.428
			x	x	71.43	60.71	71.43	57.14	46.43	61.428
	x				50	39.29	53.57	46.43	42.86	46.43
	x			x	71.43	60.71	71.43	57.14	46.43	61.428
x	x				82.14	64.29	78.57	71.43	67.86	72.858
x	x			x	82.14	64.29	78.57	71.43	67.86	72.858
x			x		64.29	60.71	92.86	75	67.86	72.144
x			x	x	64.29	60.71	92.86	75	67.86	72.144
x		x			78.57	50	89.29	71.43	64.29	70.716
x		x		x	78.57	50	89.29	71.43	64.29	70.716
		x	x		67.86	50	64.29	60.71	57.14	60
		x	x	x	67.86	50	64.29	60.71	57.14	60
	x	x			53.57	42.86	53.57	50	53.57	50.714
	x	x		x	53.57	42.86	53.57	50	53.57	50.714
x			x		71.43	53.57	60.71	57.14	57.14	59.998
x			x	x	71.43	53.57	60.71	57.14	57.14	59.998
x	x		x		71.43	64.29	78.57	78.57	64.29	71.43
x	x		x	x	71.43	64.29	78.57	78.57	64.29	71.43
x	x	x			75	50	71.43	67.86	71.43	67.144
x	x	x		x	75	50	71.43	67.86	71.43	67.144
x		x	x		75	50	85.71	75	71.43	71.428
x		x	x	x	75	50	85.71	75	71.43	71.428
	x	x	x		57.14	32.14	57.14	71.43	57.14	54.998
	x	x	x	x	57.14	32.14	57.14	71.43	57.14	54.998
x	x	x	x		67.86	57.14	71.43	75	67.86	67.858
x	x	x	x	x	67.86	57.14	71.43	75	67.86	67.858
Promedio					63.23	49.23	67.59	62.97	57.01	63.80
				x	63.9	50.34	68.71	62.97	57.23	64.47

Cuadro 7: Resultados LDA

El máximo porcentaje de clasificación obtenido por LDA fue 92,86 % para el subconjunto de prueba 3, utilizando las características de DWT y la combinación DWT-Lyapunov. El mejor porcentaje promedio de clasificación fue 72,858 %, utilizando las características de DWT y la combinación DWT-CWT.

Resultados ANN:

Características					Subconjuntos de Prueba					
DWT	CWT	CSP	Lyapunov	PCA	1	2	3	4	5	Promedio
		x			50	46.43	57.14	53.57	50	51.428
		x		x	50	46.43	57.14	53.57	50	51.428
x					70.36	63.57	92.86	73.39	67.86	73.608
x				x	71.43	63.39	92.68	73.04	67.86	73.68
			x		67.86	53.57	75	50	57.14	60.714
			x	x	67.86	53.57	75	50	57.14	60.714
	x				57.14	39.29	57.14	50	50	50.714
	x			x	57.14	39.29	57.14	50	50	50.714
x	x				75.71	64.29	76.96	70.18	67.5	70.928
x	x			x	75.54	63.04	77.5	70.71	71.43	71.644
x			x		76.07	63.21	77.5	69.64	63.93	70.07
x			x	x	65	58.93	92.68	74.64	68.21	71.892
x		x			75.54	57.68	85.71	70	64.11	70.608
x		x		x	77.32	57.14	85.71	70.54	63.75	70.892
		x	x		67.86	46.43	64.29	53.57	64.29	59.288
		x	x	x	67.86	46.43	64.29	53.57	64.29	59.288
	x	x			53.37	42.86	53.57	53.57	42.86	49.286
	x	x		x	53.37	42.86	53.57	53.57	42.86	49.286
x			x		64.29	50	60.71	64.29	53.57	58.572
x			x	x	64.29	50	60.71	64.29	53.57	58.572
x	x		x		72.86	59.64	76.07	75.54	68.93	70.608
x	x		x	x	73.57	62.86	77.5	77.32	70.89	72.428
x	x	x			75.54	55.18	72.32	69.11	66.25	67.68
x	x	x		x	77.14	55.36	71.43	70.36	70.18	68.894
x		x	x		68.93	52.14	82.32	70.18	64.11	67.536
x		x	x	x	72.5	50.89	83.21	71.43	64.64	68.534
	x	x	x		60.71	35.71	57.14	75	57.14	57.14
	x	x	x	x	60.71	35.71	57.14	75	57.14	57.14
x	x	x	x		70.71	54.11	71.61	74.11	68.57	67.822
x	x	x	x	x	68.21	53.57	71.43	74.82	68.04	67.214
Promedio					63	49.13	66.45	61	56.95	63.06
				x	62.69	58.84	67.5	61.67	57.81	63.48

Cuadro 8: Resultados ANN

El máximo porcentaje de clasificación fue obtenido por ANN 92,86 % para el subconjunto de prueba 3, utilizando las características de DWT. El mejor porcentaje promedio de clasificación fue 73,68 %, utilizando las características de DWT y aplicando PCA para su preprocesamiento.

5.2. Discusión

Para estudiar la generalización de los clasificadores propuestos con respecto al número de subconjuntos, se realizó el mismo procedimiento de clasificación con una validación cruzada de orden 4 con subconjuntos de igual tamaño. Los mejores grupos de resultados para este caso se muestran en el Cuadro 9.

Grupos de Resultados	Clasificación (%)		
	SVM	LDA	ANN
1	85.71	88.57	83.93
2	82.86	85.71	81.71
3	80	82.86	80
4	79.11	80	79
5	77.14	74.29	77.59
Promedio	80.96	82.28	80.44

Cuadro 9: Mejores Resultados 4 Subconjuntos

La siguiente gráfica de barras muestra los resultados anteriores obtenidos para cada clasificador.

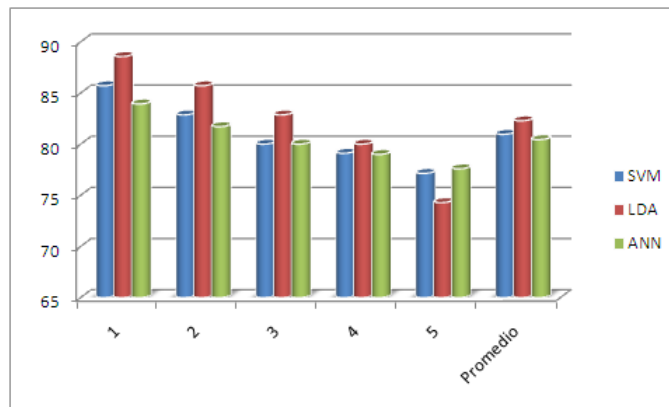


Figura 13: Mejores Resultados 4 Subconjuntos

■ Generalización

Los clasificadores propuestos no presentaron una buena generalización para los distintos subconjuntos de prueba. La única combinación que presentó un buen promedio y poca dispersión fue la de DWT-CWT con SVM y ANN.

Esto se debe principalmente a que en la mayoría de los casos el peor porcentaje de clasificación se presenta en el segundo subconjunto de prueba, sesgando el promedio para los tres clasificadores. El mismo fenómeno se pudo evidenciar con el primer subconjunto al hacer validación cruzada de orden 4, lo cual podría significar que las condiciones experimentales de algunos de estos ensayos fueron bastante distintas a las de los demás.

■ Preprocesamiento

A pesar de que la diferencia no fue tan significativa al aplicar preprocesamiento con PCA a las características, se pudo observar que en la mayoría de los subconjuntos de prueba se igualó ó se obtuvo un aumento en el porcentaje de clasificación.

En cuanto al preprocesamiento con ICA, desafortunadamente los resultados de clasificación empeoraron notablemente, y por este motivo no son incluidos en los cuadros.

5.3. Resultados anteriores

Para comparar los resultados de clasificación de las diferentes técnicas y clasificadores, se buscaron artículos que trabajaran con el mismo conjunto de datos (Competencia BCI 2003, dataset III).

Los máximos porcentajes de clasificación obtenidos con el máximo exponente de Lyapunov como característica, fueron 71.43 % para LDA y 75 % para SVM y ANN. Sin embargo, no se encontraron publicaciones que utilizaran el mismo conjunto de datos.

En [25], utilizan CSP como técnica de extracción de características y SVM como clasificador, obteniendo un porcentaje máximo de clasificación de 92.86 %. A pesar de que nuestro máximo porcentaje de clasificación en este caso fue de 71.43 %, al combinar CSP con DWT y PCA, se obtuvo un porcentaje máximo de 89.29 %. En el método propuesto en [25] se toman 40 muestras entre el segundo 4 y 5, en las que se presenta la mayor diferencia de potencias instantáneas. Sin embargo, como no se utilizaron todas las muestras del ensayo, y las utilizadas se seleccionaron de forma visual, este enfoque no satisface los criterios de generalización y detección automática deseados para este trabajo.

En [35] utilizan DWT como técnica de extracción de características y una red neuronal *back-propagation* como clasificador, obteniendo un porcentaje máximo de clasificación de 92.4 %. Nuestro máximo porcentaje de clasificación en este caso fue de 92.86 %; sin embargo, dado que se desconoce si dicho porcentaje corresponde al promedio de varias redes ó al promedio de una validación cruzada, y que el tamaño de su conjunto de muestras es el doble, no es posible hacer una comparación directa de los resultados.

En [32] utilizan CWT como técnica de extracción de características y SVM como clasificador, obteniendo un porcentaje máximo de clasificación de 81.43 %. Aunque nuestro mejor caso no superó este resultado (75 %), al combinarlo con DWT y PCA se obtuvo un porcentaje máximo de 82.14 %. La diferencia con nuestros resultados puede deberse a que en [32] se usa una validación cruzada de orden 10 y se construye un vector de 48 características separando las señales por bandas de frecuencia y usando los tres canales.

Finalmente, los ganadores de la competencia BCI 2003-Data Set III [36] reportaron un error tan bajo como 10.7 %. En nuestro caso, el mínimo error de clasificación fue 7.14 %.

6. CONCLUSIONES

En el procesamiento de señales cerebrales para aplicaciones BCI, existe un conjunto de variables y parámetros que pueden llegar a limitar el desempeño de los sistemas de reconocimiento de patrones. Algunas de las que más afectan estos sistemas son: el ruido durante la adquisición, el tamaño de los datos, los artefactos, las condiciones del experimento y la disposición del sujeto de prueba. Dado que en este trabajo no se llevo a cabo la adquisición de datos, no se tuvo control sobre las variables mencionadas, por lo que los resultados obtenidos estuvieron sujetos a estas. A pesar de esto, los resultados fueron superiores a los reportados por los ganadores de esta competencia (Competencia BCI 2003, dataset III).

En cuanto a la extracción de características, la Transformada Discreta Wavelet fue la técnica con la que se obtuvo el mejor porcentaje de clasificación. Por otro lado, la Transformada Continua Wavelet, a pesar de no haber reportado buenos resultados, permitió corroborar de forma visual el cambio de la energía de los canales debido a la activación contralateral del cerebro durante la imaginación de movimientos.

Para el caso del análisis de caoticidad por Lyapunov, a pesar de no haber sido sugerido por publicaciones para problemas de BCI, se encontró que sus resultados pueden llegar a igualar e incluso superar a los de técnicas usadas tradicionalmente como la Transformada Continua Wavelet. Queda abierta la posibilidad de seguir estudiando su utilidad para la caracterización de los patrones asociados a la imaginación de movimientos.

Con respecto a la selección del conjunto de prueba, se pudo confirmar que el uso de validaciones cruzadas mejora los resultados de clasificación. En particular, la validación cruzada de orden 5 fue la que mejor desempeño presentó con el conjunto de datos utilizado.

Una de las principales limitaciones de este trabajo radica en el tamaño del conjunto de datos, pues al utilizar 140 ensayos y 28 de estos para prueba, el mínimo error que se puede obtener es de 3.57 % y las diferentes tasas de error varían en esta misma proporción.

Por otro lado, para las técnicas de preprocesamiento se pudo observar que el Análisis de Componentes Principales produjo un aumento en los porcentajes de clasificación, mientras que el Análisis de Componentes Independientes no ofreció resultados favorables.

Para cada uno de los sistemas de clasificación utilizados, se comprobó la efectividad de escoger los parámetros que mejor se adapten al problema en cuestión. En este caso, la selección de un kernel 'lineal' para la Máquina de Vectores de Soporte y de funciones de activación 'purelin' para la Red Neuronal fue la más apropiada. Cabe mencionar que los resultados de clasificación por Análisis Lineal Discriminante fueron comparables con los de la Máquina de Vectores de Soporte, sin la necesidad de seleccionar parámetros.

Finalmente, el vector de características formado por las cantidades derivadas de la Transformada Discreta Wavelet generó un espacio de características linealmente separable (como lo confirmó LDA), y con este, la Máquina de Vectores de Soporte presentó los mejores resultados de clasificación, alcanzando un error tan bajo como 7.14 %.

BIBLIOGRAFÍA

- [1] D. Huang, P. Lin, D.Y. Fei, X. Chen, and O. Bai, "EEG-Based Two- Dimensional Cursor Control", *31st Annual International Conference of the IEEE EMBS*, pp. 4547-50, Sep 2009.
- [2] BCI Competition 2003 website: <http://www.bbc.de/competition/ii/>
- [3] A. Akrami, S. Solhjoo, A. Motie-Nasrabadi, and M. Hashemi-Golpayegani, "EEG-Based Mental Task Classification: Linear and Nonlinear Classification of Movement Imagery", *Proceedings of the IEEE Engineering in Medicine and Biology 27th Annual Conference*, pp. 4626-29, Sep 2005.
- [4] A. Tahirl and M. Arif, "Classification of Single-Trial Self-paced Finger Tapping Motion for BCI Applications", *Pakistan Institute of Engineering & Applied Sciences*, Islamabad, Pakistan.
- [5] D. Garrett, D. A. Peterson, C.W. Anderson, and M. Thaut, "Comparison of Linear, Nonlinear, and Feature Selection Methods for EEG Signal Classification", *IEEE Transactions On Neural Systems And Rehabilitation Engineering*, Vol. 11, No. 2, Jun 2003.
- [6] B. Wang, C. Wong, F. Wan, P. Mak, P. Mak, and M. Vai "Comparison of Different Classification Methods for EEG-Based Brain Computer Interfaces: A Case Study", *Proceedings of the 2009 IEEE International Conference on Information and Automation*, pp- 1416-21, Jun 2009.
- [7] P. Herman, G. Prasad, T.M. McGinnity and D. Coyle, "Comparative Analysis of Spectral Approaches to Feature Extraction for EEG-Based Motor Imagery Classification", *IEEE Trans. on Neural Systems and Rehabilitation Engineering*, vol. 16, no. 4, pp. 317-326, Aug 2008.
- [8] S. Hosni, M. Gadallah, S. Bahgat and M. AbdelWahab, "Classification of EEG Signals Using Different Feature Extraction Techniques for Mental Task", *Report of Ain shams University*, El Cairo, Egipto.
- [9] B. Xu, A. Song and J. Wu, "Algorithm of Imagined Left-right Hand Movement Classification Based on Wavelet Transform and AR Parameter Model", *School of Instrument Science and Engineering Southeast University*, Nanjing, China.
- [10] W. Tu and Q. Wei, "Classification of Three-Class Motor Imagery EEG Data by Combining Wavelet Packet Decomposition and Common Spatial Pattern", *International Conference on Intelligent Human-Machine Systems and Cybernetics*, Vol No.1, pp. 188-191, China, 2009.
- [11] B. Wang, L. Jun, J. Bai, L. Peng, Y. Li, and G. Li, "EEG Recognition Based on Multiple Types of Information by Using Wavelet Packet Transform and Neural Networks", *Proceedings of the IEEE 27th Annual Conference Engineering in Medicine and Biology*, pp. 5377-5380, Shanghai, China, Sep 2005.
- [12] J. Xue, H. Zhang, C. Zheng and X. Yan, "Wavelet Packet Transform for Feature Extraction of EEG During Mental Tasks", *Proceedings of the Second International Conference on Machine Learning and Cybernetics*, pp. 360-363, Xi'an, China, nov 2003.
- [13] L. Shoker, S. Sanei and A. Sumich, "Distinguishing Between Left and Right Finger Movement from EEG Using SVM", *Proceedings of the IEEE 27th Annual Conference Engineering in Medicine and Biology*, pp. 5420-5423, Shanghai, China, Sep 2005.
- [14] <http://bci.tugraz.at/>
- [15] <http://www.bccn-berlin.de/Home>

- [16] <http://neuro.med.tsinghua.edu.cn/eng/>
- [17] J. Quiroga, “Señales y Sistemas”. Bogotá, Colombia.
- [18] S. Mallat, “A Wavelet Tour of Signal Processing”, ed., Springer, second edition.
- [19] P. Jahankhani, V. Kodogiannis and K. Revett, “EEG Signal Classification Using Wavelet Feature Extraction and Neural Networks”, *IEEE International Symposium on Modern Computing* John Vincent Atanasoff 2006
- [20] D. POLLOCK, “A Handbook of Time-Series Analysis, Signal Processing and Dynamics”, ed., Academic Press.
- [21] J. Bonilla, A. Manotas “Análisis de la Actividad Eléctrica Cerebral en Pacientes Epilépticos”, Dept. Electrónica, Pontificia Universidad Javeriana, Bogotá D.C., 2003.
- [22] Z. Xie and K. Wang, “Selection of Embedding Parameters in Phase Space reconstruction”, *Second International Conference on Intelligent Computation Technology and Automation*, 2009.
- [23] M. Rosenstein, J. Collins J.J. and C. De Luca, “A Practical Method for Calculating Largest Lyapunov Exponents from Small Data Sets”, *Boston University*, 1992.
- [24] A. Wolf, J. Swift, L. Swinney and J. Vastano, “Determining Lyapunov Exponents from a Time Series”, *Department of Physics, University of Texas Texas*.
- [25] M. Li, J. Liu, D. HAO and J. YANG, “An Improved CSP Algorithm and Application in Motor Imagery Recognition”, *IEEE Fifth International Conference on Natural Computation, Institution of Artificial Intelligence and Robot, Beijing University of Technology, China* 2009.
- [26] L. Wang, X. Wu, “Classification of Four-Class Motor Imagery EEG Data Using Spatial Filtering”, *The Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, Anhui Univesity, Hefei, China*.
- [27] R. Duda, P. Hart, D. Stork, “Pattern Classification”, second edition.
- [28] I. Jolliffe, “Principal Component Analysis”, ed., Springer, NY, 2002, XXIX, 487 p. 28.
- [29] A. Flugge, “Non-negative PCA for EEG-Data Analysis”, Abril 20, 2009.
- [30] Hyvärinen y E. Oja, “Independent Component Analysis: Algorithms and Applications”, *Neural Networks Research Center, Helsinki University of Technology, Finlandia*.
- [31] J. Lau and J. Rajapakse, “Extraction of Event-Related Potentials from EEG Signals Using ICA With Reference”, *Proceedings of Intemational Joint Conference on Neural Networks, School of Computer Engineering Nanyang Technological University, Singapore, Montreal, Canada, July 31 - August 4, 2005*.
- [32] S. Darvishi and A. Al-Ani, “Brain-Computer Interface Analysis Using Continuous Wavelet Transform and Adaptive Neuro-Fuzzy Classifier”, *Proceedings of the 29th Annual International Conference of the IEEE EMBS*, pp. 3220-3223, aug 2007.
- [33] D. Kugiumtzis, “Physica D: Nonlinear Phenomena, State Space Reconstruction Parameters in the Analysis of Chaotic Time Series-the Role of the Time Window Length”, ed., Elsevier, 1996.
- [34] S. Yang, C. Jia, “Two practical methods of phase space reconstruction”, *Chinese Journal of Physics*, 2002.

- [35] M. Li, R. Wang, D. Hao, J. Yang, "Feature Extraction and Classification of Mental EEG for Motor Imagery", *Fifth International Conference on Natural Computation*, Tianjian, China, 2009.
- [36] S. Lemm, C. Schafer, G. Curio "BCI Competition 2003-Data Set III: Probabilistic Modeling of Sensorimotor μ Rhythms for Classification of Imaginary Hand Movements, *IEEE transactions on Biomedical Engineering*, 2004.
- [37] <http://www.edmontonneurotherapy.com/i//10-20placement.gif>
- [38] http://files.nocions.webnode.com/200000066-414b54244c/resint_eeg16.jpg