



INTERFAZ CEREBRO COMPUTADOR ADAPTATIVA, BASADA EN AGENTES SOFTWARE PARA LA DISCRIMINACIÓN DE CUATRO TAREAS MENTALES

Tesis doctoral presentada por Javier Ferney Castillo Garcia
Programa de Postgrado en Ingeniería Eléctrica y Electrónica

Dirigida por Eduardo Caicedo Bravo



INTERFAZ CEREBRO COMPUTADOR ADAPTATIVA, BASADA EN AGENTES SOFTWARE PARA LA DISCRIMINACIÓN DE CUATRO TAREAS MENTALES

Tesis doctoral presentada por Javier Ferney Castillo Garcia

Programa de Postgrado en Ingeniería Eléctrica y Electrónica

Dirigida por Eduardo Caicedo Bravo

Cali, agosto 2015

*A mi madre, mi padre, mi esposa, mis hermosos hijos y mis hermanos:
Gladys, Oscar, Diana, Josue, Kaleb, Alfredo, Gladyscita, Elsy, Yuli y
Harrinson.*

Resumen

Las interfaces cerebro computador buscan mejorar la calidad de vida de las personas con discapacidad, y se han convertido en una línea de investigación de gran crecimiento por facilitar un nuevo canal de comunicación y por potencializar las capacidades humanas. Una interfaz cerebro computador se compone de una red de sensores que se encargan de capturar las señales Electro-Encefalográficas (EEG), digitalizarlas para posteriormente enviarlas a un computador, donde se realiza un proceso de extracción de características, discriminación y clasificación, para finalmente generar los comandos necesarios en la ejecución de una aplicación.

La propuesta formalizada en este documento tiene como objetivo realizar el proceso de adaptación mutua entre una interfaz cerebro computador y el usuario; incrementando su usabilidad y permitiendo que se discriminen hasta cuatro tareas mentales espontáneas. Como resultado se implementó una interfaz adaptativa basada en agentes de software, los cuales se encargan de identificar los estados mentales del usuario y de los ajustes de parámetros para el proceso de discriminación de al menos cuatro tareas mentales, con el fin de activar comandos en un robot o silla de ruedas eléctrica. Este propósito se logró con la caracterización de los estados mentales de mayor incidencia en el proceso de discriminación de las tareas mentales definidas, seguido de una identificación de las tareas mentales de mayor separación temporal-frecuencial y la implementación de los algoritmos adecuados para su discriminación.

Abstract

Brain computer interfaces seek to improve the quality of life of people with disabilities and have become a research of high growth by providing a new communication channel and potentiate human capabilities. A brain computer interface consists of a sensors network that are responsible for capturing the EEG signals, digitalization and later sent to a computer where a features extraction, discrimination and classification are performed to finally generate the commands needed in the application execution.

The proposal formalized in this document has as main objective to make the process of mutual adaptation between a brain computer interface and the user; increasing its usability and allowing it to discriminate up to four spontaneous mental tasks. As a result an adaptive interface based on software agents is implemented, which is responsible for identifying mental states of the user and the parameter settings for the discrimination process at least four mental tasks, in order to activate commands in a robot or electric wheelchair. This purpose is achieved with the characterization of mental states with the highest incidence in the process of discrimination of defined mental tasks, followed by an identification of mental tasks with temporal-frequency separation and implementation of appropriate algorithms for discrimination.

Agradecimientos

A Dios por proveerme de todo lo necesario para cumplir mis metas, a mis padres por el apoyo incondicional, a mi esposa e hijos por darme la motivación e inspiración para mirar de frente al sol y dar todo de mí para resolver todas las vicisitudes inherentes al proceso de formación.

A mi director Eduardo Caicedo, por su ejemplo, por inculcar en mí ese deseo enorme de ser mejor y pensar en positivo.

A todas aquellas personas maravillosas que he tenido la oportunidad de conocer y compartir ideas, sueños y tantas otras cosas: un abrazo inmenso por todo.

Para tod@ a galera do laboratório de automação Inteligente da UFES, ~ deixo para vocês meus melhores desejos e me lembrarei sempre daquelas experiências, as quais serão inesquecíveis na minha vida: “Cadê os artigos” (Obrigado Prof. Teodiano pelo aprendizado).

Javier Castillo,

Cali, agosto 2015

Índice general

Índice de figuras	xiii
Índice de tablas	xvii
1 Introducción	1
1.1 Planteamiento del problema	3
1.2 Objetivos de la tesis	4
1.2.1 Objetivo General	4
1.2.2 Objetivos Específicos	4
1.3 Valoración de resultados	4
1.4 Importancia y aplicabilidad	5
1.5 Estructura del documento	6
1.6 Conclusión	7
2 Estado de la técnica de las Interfaces Cerebro Computador Adaptativas	9
2.1 Antecedentes	10
2.2 Discusión	13
2.3 Tendencias	15
2.3.1 Limitaciones presentes en la aBCIs	15
2.4 Conclusión	16

ÍNDICE GENERAL

3	Métodos para la implementación de una BCI	17
3.1	Pre-procesamiento	18
3.1.1	Filtros Espaciales	18
3.1.1.1	Promedio de la referencia común	18
3.1.1.2	Laplaciano de superficie	19
3.1.2	Análisis de componentes independientes	19
3.2	Extracción de Características	19
3.2.1	Bandas de Potencia	20
3.2.2	Autoregresores	21
3.2.3	Coherencia	21
3.2.4	Factor de Fase bloqueada	22
3.2.5	Espectro de potencias basado en Wavelets	22
3.2.6	Transformada de Hilbert-Huang	23
3.2.7	Características Estadística de series de tiempo	24
3.3	Métodos de selección de características y reducción de dimensio- nalidad	25
3.3.1	Selección secuencial hacia adelante	26
3.3.2	Selección secuencial hacia atrás	26
3.3.3	Selección secuencial flotante	26
3.3.4	Selección basado en la distancia de Silhoutte	27
3.3.5	Algoritmo Genético	28
3.3.6	Análisis de componentes principales	28
3.3.7	Divergencia del Kullback Leibler	29
3.4	Clasificadores	29
3.4.1	Analisis de discriminante Lineal	30
3.4.2	K vecinos cercanos	31
3.4.3	Redes Neuronales	32
3.4.3.1	Perceptrón multicapa	32
3.4.3.2	Máquinas de aprendizaje extremo	33
3.4.3.3	Aprendizaje de vectores de cuantización	33
3.4.3.4	Teoria resonancia adaptativa	33
3.4.3.5	Otras arquitecturas de redes neuronales	35
3.4.4	Máquinas de vectores de soporte	36

ÍNDICE GENERAL

3.4.5	Validación cruzada	36
3.5	Agentes de software	37
3.5.1	Taxonomía según Nwana	38
3.5.2	Agente de Software para una Interfaz Cerebro Computador	39
3.5.2.1	Agente de Software para selección de características	40
3.5.2.2	Agente de Software con umbral estático	47
3.5.2.3	Agente de Software con umbral dinámico	47
3.5.2.4	Agente de Software con aprendizaje activo	48
3.5.3	Medidas de Desempeño	50
4	Pruebas, análisis de datos y resultados	53
4.1	Plataforma de experimentación para una BCI para tareas mentales	53
4.1.1	Análisis ERD/ERS	54
4.1.2	Análisis de la nube de datos complejos para señales EEG	55
4.1.3	Análisis Bandas de Potencia	55
4.1.4	Análisis de la estadística compleja para señales EEG	56
4.1.5	Análisis de la coherencia	58
4.1.6	Análisis del factor de fase bloqueada	58
4.2	Plataforma de Simulación	59
4.2.1	Análisis de base de datos para cinco tareas mentales	59
4.2.2	Análisis de base de datos para cuatro tareas mentales	67
4.3	Protocolo de experimentación	77
4.3.1	Descripción de los equipos	77
4.3.2	Descripción de los usuarios	79
4.3.3	Descripción de la tareas mentales	79
4.3.4	Criterios de Selección de la tareas mentales	80
4.3.5	Entrenamiento activo	80
4.4	Discusión y Análisis de Resultados	84
5	Conclusiones	89
5.1	Introducción	89
5.2	Conclusiones sobre el problema de investigación	89
5.3	Implicaciones teóricas	90

ÍNDICE GENERAL

5.4	Implicaciones prácticas	91
5.5	Limitaciones	91
5.6	Investigaciones Posteriores	91
5.7	Publicaciones	92
	Bibliografía	101

Índice de figuras

2.1	Representación del proceso de aprendizaje.	10
3.1	Esquema general de una BCI	18
3.2	Representación de la SW	27
3.3	Atributos de agentes según [63]	39
3.4	aBCI propuesta basada en agentes de software	41
3.5	Representación en modelo neuronal para una aBCI.	42
3.6	Diagrama esquemático de una aBCI con ajuste de parámetros usando SA.	43
3.7	Diagrama esquemático de un SA	46
3.8	aBCI propuesta basada en agentes de software con umbral estático	47
3.9	aBCI propuesta basada en agentes de software con umbral dinámico	48
3.10	aBCI propuesta basada en agentes de software con aprendizaje activo	50
3.11	Matriz de Confusión y medidas de desempeño para una BCI . . .	52
4.1	Plataforma de experimentación	54
4.2	Modo de análisis para ERD/ERS, representación topográfica . . .	55
4.3	Modo de análisis de la nube de datos complejos para un canal . . .	56
4.4	Modo de análisis de la densidad espectral de potencia y ERD/ERS en todos los electrodos	56
4.5	Análisis de la cuantitativo de la señal EEG para un canal)	57
4.6	Análisis de la coherencia en las banda alfa baja 8–10[Hz], alfa alta 10–12 [Hz], alfa 8–13 [Hz], beta 13–30 [Hz] y las interconexiones de todos los canales	58

ÍNDICE DE FIGURAS

4.7	Análisis del factor de fase de bloqueo antes y después de la tarea mental en todas las combinaciones de los canales	59
4.8	Desempeño de los métodos de extracción de característica, la tasa de acierto de los clasificadores para cada usuario.	61
4.9	Representación en modelo neuronal para una aBCI para el usuario 2.	63
4.10	Tasa de aciertos e ITR para todos los usuarios. Desempeño para una aBCI para todos los usuarios. a) Exactitud (Izquierda) e ITR (Derecha) para el usuario 1; b) Exactitud e ITR para el usuario 2; c) Exactitud e ITR para el usuario 3; d) Exactitud e ITR para el usuario 4; e) Exactitud e ITR para el usuario 5; f) Exactitud e ITR para el usuario 6; y g) Exactitud e ITR para el usuario 7.	64
4.11	Desempeño de todas las tareas mentales para cada usuario	65
4.12	Distribución de las características usando SW para el usuario 1.	68
4.13	Distribución de las características usando SW para el usuario 2.	69
4.14	Distribución de las características usando SW para el usuario 3.	70
4.15	Distribución de las características usando SW para el usuario 4.	71
4.16	Resultados de un SA con umbral estático, P_u es el tiempo sin clasificar.	72
4.17	Resultados de un SA con umbral dinámico.	73
4.18	Resultados de un SA con aprendizaje activo.	74
4.19	Representación en componentes principales de los datos del usuario 1 antes y después del SA con aprendizaje activo.	74
4.20	Representación en componentes principales de los datos del usuario 2 antes y después del SA con aprendizaje activo.	75
4.21	Representación en componentes principales de los datos del usuario 3 antes y después del SA con aprendizaje activo.	75
4.22	Representación en componentes principales de los datos del usuario 4 antes y después del SA con aprendizaje activo.	76
4.23	Gorro en Neoprene con el <i>hardware</i> modificado del Emotiv EPOC™.	78
4.24	Posicionamiento de los electrodos	78
4.25	Protocolo experimental para entrenamiento activo para aBCI.	81
4.26	Índice NASA TLX para sobre carga.	82

ÍNDICE DE FIGURAS

4.27 Evaluacion de un par de características para cuantificar el índice NASA TLX	83
---	----

Índice de tablas

2.1	Relación de las principales características de las interfaces adaptativas	14
4.1	Resultados de la tasa de aciertos, el coeficiente Kappa, y el mejor clasificador para cada usuario.	60
4.2	Resultados para todos los métodos de extracción de características.	62
4.3	Desempeño de una aBCI. Medidas de desempeño de la aBCI y la BCI tradicional para todos los usuarios, los valores entre parentesis “()” corresponden a la BCI tadicional.	63
4.4	Rechazo de Trials. Calidad de los datos usando el rechazo de muestras para varios valores.	66
4.5	Tasa de aciertos y SW para cada método de extracción de características	72
4.6	Tabla comparativa de las implementaciones aBCI basadas en SA para cuatro tareas mentales. BCI es la interfaz tradicional, Th FX es SA con umbral estático, Th Dn es SA con umbral dinámico y Ac Ln es SA con aprendizaje activo.	76
4.7	Resultados para entrenamiento activo para una aBCI.	84
4.8	Resumen de los resultados obtenidos.	87

Si la mente humana fuese tan simple de entender, entonces seríamos demasiado simples para entenderla.

Emerson Pugh

CAPÍTULO

1

Introducción

Las interfaces cerebro computador (BCI del inglés, *Brain Computer Interface*) han motivado las tendencias actuales de investigación por su capacidad de facilitar un nuevo canal de comunicación, permitiendo la interacción de personas con discapacidad con su entorno: control de silla de ruedas eléctricas o robots [19, 40, 41, 54, 79, 82], control de sistemas domóticos [4], operación de prótesis [50], selección de letras de un teclado virtual [37], navegación en internet [45, 60] y por potencializar las capacidades humanas: cognición aumentada [80].

Si bien, en muchos sentidos, el cerebro humano es la más flexible y poderosa máquina de aprendizaje existente en la actualidad, los sistemas BCIs pueden utilizar esta capacidad para mejorar su rendimiento. En esta línea de pensamiento necesitamos: en primer lugar, nuevos paradigmas de entrenamiento y sistemas de retroalimentación para mostrar al usuario sus estados cerebrales en línea y aprender a cómo controlarlos, y en segundo lugar, los sistemas cooperativos de aprendizaje deben realizar un seguimiento y adaptarse a las cambio de estado del usuario (sobre-carga mental, [47]; niveles de atención, [35]; y fatiga, [80]). En la actualidad las BCIs, incluso para aquellos usuarios que reciben muchas horas de entrenamiento en una situación de neuro-regeneración, sólo admiten un máximo de dos o tres dimensiones de control independiente [76]. La búsqueda para que el control dimensional en el espacio de las señales del cerebro sea uno a uno, podría surgir

1. INTRODUCCIÓN

como resultado de una mutua adaptación, esta se podría lograr, iniciando por una decodificación independiente de las dimensiones, es decir obteniendo un control muy complejo sin la necesidad de tener datos complejos de entrenamiento en todo el espacio de trabajo.

En la actualidad existe amplia información sobre la interfaces cerebro computador desde la óptica de sistemas transductores de ondas cerebrales a comandos de computador [52], estas presentan niveles de confiabilidad y usabilidad bastante aceptables, pero aun existe la dificultad en el proceso de entrenamiento, ya que esta actividad tiene altos requerimiento de tiempo, pues el usuario es quien se adapta al sistema, y no se presenta una mutua adaptación que facilite esta interacción.

La presente tesis doctoral presenta una interfaz cerebro computador adaptativa (aBCI del inglés, *Adaptive Brain Computer Interface*), cuya adaptabilidad está relacionada al proceso de discriminación de clases, facilitando la interacción usuario-interfaz, reduciendo el tiempo de entrenamiento y ampliando la capacidad de discriminación de hasta cuatro tareas mentales espontáneas. La estrategia utilizada para abordar este problema consiste en la implementación de algoritmos que detectan los estados mentales en el proceso de clasificación.

Esta investigación se encuentra enmarcada en el proyecto CYTED-IBERADA, liderado por países de la comunidad Europea para el desarrollo de interfaces adaptadas a personas con discapacidad, cuyo impacto social es muy alto debido al énfasis de mejorar la calidad de vida de las personas con discapacidad.

1.1 Planteamiento del problema

Las BCIs corresponden a un campo de investigación que crea un nuevo canal de comunicación, y permite potencializar las capacidades de aquellos que presentan alguna deficiencia, propendiendo en una mejora de su calidad de vida. Las BCIs espontáneas en la actualidad presentan buenos desempeños del orden del 70 al 90 % de exactitud cuando se trata de clasificar hasta tres tareas mentales [52], con un mayor número de tareas este porcentaje se reduce del 50 al 79 % [25]. Las técnicas de clasificación para la discriminación de más de tres tareas, implican la utilización de nuevos algoritmos o estrategias de clasificación (fusión de datos, transformación frecuencia-tiempo, máquinas de aprendizaje, etc) y modificaciones en los paradigmas de entrenamiento [1], esto debido a la no estacionaridad de las ondas cerebrales. Esta dificultad se hace evidente cuando los estados mentales del sujeto bajo prueba cambian (fatiga, [80] ; niveles de atención, [35]; sobrecarga mental, [47], etc), trayendo consigo la necesidad de un mecanismo de adaptación mutua entre la interfaz y el usuario [86].

Sin embargo, hasta la fecha no se han presentado métodos de diseño de interfaces de usuario, donde explícitamente se incluyan las capacidades de adaptación en una BCI, y que su diseño se encuentre incluido dentro del ciclo de vida de una aplicación. Es este uno de los problemas donde pretende profundizar en la presente tesis, aportando en la integración total del diseño de la adaptación en el proceso de desarrollo de las BCIs.

El diseño de aBCIs se plantean dos frentes principales: por una parte es necesario modelar los constructores que describen el proceso de adaptación, y por otra parte se hace necesaria la inclusión de algún mecanismo que sea capaz de inferir conocimiento a partir de la información recogida de la interacción con el usuario, y que sea capaz de tomar decisiones basándose en dicha información. Estas decisiones pueden ser determinantes en la aceptación o no por parte del usuario de un sistema adaptativo. La mala elección de la adaptación o la elección inadecuada para aplicarla a una BCI, puede conducir a una clara degradación de la usabilidad de la interfaz. La toma de decisiones, así como la inferencia de conclusiones a partir de una serie de datos son facultades humanas, las cuales se intenta emular a través

1. INTRODUCCIÓN

de distintas técnicas dentro del mundo de la Inteligencia Artificial. Sistemas basados en reglas, reglas Bayesianas o redes neuronales son algunas de las técnicas tradicionalmente usadas para dicho propósito. Sin embargo, en los últimos años se ha suscitado un gran interés por el concepto de agente software, como abstracción para el desarrollo de aplicaciones altamente dinámicas donde puede ser incluido un modelo mental que permita aportar al sistema ciertas capacidades de razonamiento.

1.2 Objetivos de la tesis

1.2.1 Objetivo General

Desarrollar una BCI adaptativa, capaz de discriminar al menos cuatro tareas mentales.

1.2.2 Objetivos Específicos

- Diseñar una interfaz adaptativa para la comunicación cerebro computador
- Caracterizar los estados mentales y seleccionar el(os) que ofrezca(n) la mayor separación temporal-frecuencial para la definición de tareas mentales en una BCI
- Implementar una estrategia de integración de agentes software en la identificación de estados mentales para una interfaz adaptativa
- Proponer una estrategia de entrenamiento para una interfaz adaptativa
- Desarrollar un clasificador para el reconocimiento de al menos cuatro tareas mentales empleando señales EEG y máquinas de aprendizaje
- Validar a nivel de simulación la interfaz desarrollada.

1.3 Valoración de resultados

Un mayor discernimiento de los algoritmos para clasificación de los estados mentales y los tipos de tareas mentales con altos niveles de separación tiempo-frecuencia, permitirán en un futuro, el control de sistemas remotos usando solo el pensamiento,

el control de robots para terapias de rehabilitación, etc.

La metodología o estrategia utilizadas para abordar este problema, precisó de pasos y pequeños logros que permitieron sortear las dificultades que se presentaron en el camino. Las herramientas computacionales con las que se cuentan en la actualidad permitieron optimizar los algoritmos, plantear nuevos esquemas de entrenamiento, probar y seleccionar los mejores clasificadores para tal fin. Los logros de esta tesis permiten abordar las interfaces desde una perspectiva distinta, ya que al tener un nuevo esquema para la adaptación mutua, las investigaciones futuras pueden enfocarse en resolver otras de las tantas limitaciones presente en las BCIs, como el trabajar sobre el concepto de interfaces a bajo nivel (seleccionar características sencillas para luego encontrar la forma de utilizar esta información para crear combinaciones y realizar tareas más complejas), profundizar en la investigación para incrementar la dimensionalidad de las tareas a utilizar.

Las interfaces adaptativas, permitirán a las personas con discapacidad apropiarse de esta tecnología ya que se necesitaría menos tiempo en el proceso de aprendizaje, y al tenerse en cuenta la carga cognitiva del usuario, se puede ajustar los parámetros del algoritmo de discriminación, incrementando la usabilidad e interacción entre el usuario y la interfaz. La metodología propuesta es sólo un ejemplo de otras tantas posibles. Su utilidad y eficacia podría ponerse en duda, pero la necesidad de disponer de nuevos métodos que permitan mejorar el diseño y desarrollo de este tipo de herramientas es una realidad indiscutible, dada la heterogeneidad de la población de usuarios.

1.4 Importancia y aplicabilidad

Recientemente se habla de sistemas de cognición aumentada (*augmented cognition*) [80], la cual podría definirse como un ambiente de cómputo consciente de un entorno altamente personalizado, en el que el contexto se especializa en la actividad del usuario con el computador. Ante el creciente uso de la tecnologías de la comunicación, una persona invierte cada vez más tiempo interactuando con dispositivos computacionales y se ve inundada por volúmenes de información cada vez mayores. Sin embargo, el ser humano tiene un límite en sus capacidades cognitivas (memoria, aprendizaje, comprensión, atención, capacidad de juicio), determinado

1. INTRODUCCIÓN

por diversos factores como la fatiga, el ambiente, la complejidad de las tareas y el nivel de estrés. Los sistemas de cognición aumentada exploran estrategias para filtrar la información y presentarla al usuario en el momento, y a través del canal, más apropiado tratando de aumentar la capacidad de asimilación y de toma de decisiones. En este sentido, estos sistemas están formados por cuatro componentes: sensores para determinar el estado del usuario, un modelo para interpretar la información recabada, una interfaz de usuario adaptable y la infraestructura necesaria para integrarlos. La investigación presentada en este documento podría ser utilizada para evaluar el estado del usuario y así seleccionar la estrategia adecuada para apoyarlo, específicamente, el sistema sería capaz de detectar saturación en alguna de las áreas cognitivas, y elegiría una estrategia para limitar la información, utilizando algún canal alternativo (visual, auditivo, táctil), o renegociación de actividades para aumentar el desempeño. El aporte o contribución original de esta tesis está en la realización de una estrategia para la integración de la caracterización de los estados mentales del sujeto (basado en agentes software) y las tareas mentales definidas, para la ejecución de un comando específico. Esta integración es la que permitiría obtener una aBCI. Otro aporte significativo está, en la estrategia e implementación de algoritmos para la discriminación de las cuatro tareas mentales, el esquema de entrenamiento y la selección de las tareas mentales espontáneas.

1.5 Estructura del documento

Este documento presenta en el capítulo 1, una breve introducción sobre las BCIs, y las motivaciones fundamentales que dieron origen a esta investigación. En el capítulo 2 se presenta el estado de la técnica de la interfaces adaptativas haciendo énfasis en las tendencias y limitaciones actuales. Los métodos y técnicas de procesamiento propuestos para abordar el uso de agentes software, el pre-procesamiento, la extracción de características y los clasificadores son presentados en el capítulo 3. En el capítulo 4 se hace la descripción de la plataforma de análisis y presentación de resultados, y finalmente en el capítulo 5 se presentan las conclusiones y trabajos futuros.

1.6 Conclusión

El foco principal de esta tesis es la puesta en escena del proceso de interacción cerebro-computador, donde existe un proceso de mutua adaptación usando el concepto de agentes de software, este aspecto es presentado en la problemática o asunto de estudio de esta tesis. La estructura del documento facilita el proceso de presentación de la información, partiendo de un análisis del estado actual sobre las interfaces adaptativas, la descripción de los métodos ampliamente usados en la extracción de características y clasificación, también es presentada la metodología para evaluar, analizar y validar los resultados.

*Si buscas resultados distintos, no
hagas siempre lo mismo.*

Albert Einstein

CAPÍTULO

2

Estado de la técnica de las Interfaces Cerebro Computador Adaptativas

Una de las restricciones más significativas de las BCIs es la falta de adaptación para incluir la flexibilidad del cerebro en los algoritmos de clasificación o extracción de características. Este es el punto de partida para buscar procesos de adaptación que mejoren la discriminación de señales obtenidas durante la adquisición de ondas cerebrales. Un sistema adaptativo es “aquel que con base en el conocimiento altera automáticamente los aspectos de funcionalidad e interacción para conseguir ajustar las diferentes preferencias y necesidades de sus diferentes usuarios”. La necesidad de sistemas adaptativos deriva, en primera instancia, de la heterogeneidad de la población de usuarios y , en segunda instancia, en la necesidad de incrementar el número de tareas capaz de ser discriminadas.

Específicamente, una interfaz adaptativa es presentada como dos “sistemas” altamente interdependientes, donde el proceso de aprendizaje es mutuo, en la Figura 2.1 se presentan las diferentes modalidades para la interacción y aprendizaje hombre-máquina [26]. El proceso de auto-condicionamiento esta relacionado al entrenamiento del usuario en el uso de la BCI. El proceso de aprendizaje de máquina

2. ESTADO DE LA TÉCNICA DE LAS INTERFACES CEREBRO COMPUTADOR ADAPTATIVAS

esta relacionado con el entrenamiento del sistema de reconocimiento de patrones de la BCI. El proceso de aprendizaje mutuo, relaciona el aprendizaje del usuario y de la interfaz de forma simultanea.

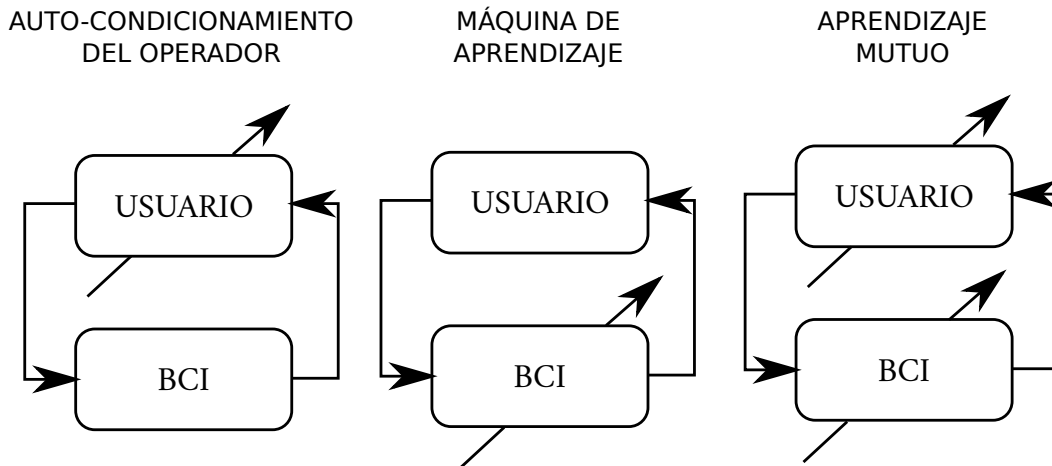


Figura 2.1: Representación del proceso de aprendizaje.

2.1 Antecedentes

Varios estudios han demostrado que la adaptación en la etapas de extracción de características y/o clasificación mejora el rendimiento del sistema de BCI [3, 15, 25, 28, 51, 55, 57, 74, 77, 84, 85].

El proyecto ABI (*Adaptive Brain Interface*), integra a las personas con discapacidad a sus labores sociales, económicas y de formación con el uso de las nuevas tecnologías de la comunicación. El núcleo de ABI es un clasificador basado en una red neuronal, la cual reconoce la tarea mental que esta realizando el sujeto, el sistema se concentra en el análisis de las variaciones de los ritmos EEG en varias áreas corticales del cerebro, las tareas mentales pueden ser: imaginación motora (por ejemplo, imaginar el movimiento de las extremidades) o cognitivas (como operaciones aritméticas) [57].

La Descomposición Empírica en Modos es una técnica capaz de procesar señales no-estacionarias y no lineales [11], esta técnica fue empleada para extraer los modos oscilatorios de la señal, a partir de los cuales se calcularon varios parámetros,

como son el valor cuadrático medio, la varianza, la entropía de Shannon, la complejidad de Lempel-Ziv y las frecuencias centrales y máximas. Además se analizaron las diferentes combinaciones de estos parámetros. Se utilizó Análisis Discriminante Lineal y Redes Neuronales para clasificar cinco tareas mentales a la vez. Para reducir la dimensionalidad de los vectores de entrada a los clasificadores, se implementó una selección y reducción de características utilizando como criterio la lambda de Wilks. Fueron alcanzados resultados promedios en la clasificación entre 50 % y 79 %. Mediante este método es posible extraer características de la señal que permiten obtener buenos resultados en la clasificación de las tareas mentales [25].

Shenoy y otros en [77], usaron técnicas de simulación para clasificación adaptativa, basadas en patrones espaciales comunes (CSP del inglés, *Common Spatial Pattern*), con esto se mejoró la tasa de acierto. Se usó un esquema de reclasificación para lo cual se trabajó sobre una ventana inicial, la cual se desplazaba sobre los datos posteriores, con lo cual se observó una tendencia en los resultados presentando una mejora en la exactitud de clasificación en comparación con la exactitud de la clasificación original. El clasificador LDA recicla datos etiquetados de acuerdo a las características del conjunto de datos sin conexión, y mejora su separación.

El enfoque de Buttfield y otros en [15], trabajan a partir del reconocimiento de los estados de los errores cognitivos, es decir, identifica los errores cometido por los usuarios usando señales cerebrales. La producción de estos potenciales de error (ERRP del inglés, *Error Related Potentials*) son respuestas a un error cometido por el usuario y son usados para reconfigurar el clasificador.

Vidaurre y otros en [84], construyeron un clasificador para BCI basado en el análisis discriminante cuadrático adaptativo. El clasificador se actualiza mediante una estimación *on-line* de la matriz de información (ADIM del inglés, *Analysis Discriminative Information Matrix*). El sistema también es capaz de proporcionar retroalimentación continua para el usuario. La validación del sistema se hizo mediante el análisis de la tasa de error y la información mutua de cada sesión. Los resultados muestran que la BCI es estable y robusta, y basado en la información de este sistema, los sujetos eran capaces de mejorar su rendimiento. Debido a la forma en que se llevaron a cabo las sesiones de entrenamiento, la retroalimentación no es necesaria y se puede encontrar una estrategia basada únicamente en la respuesta del sistema.

2. ESTADO DE LA TÉCNICA DE LAS INTERFACES CEREBRO COMPUTADOR ADAPTATIVAS

Millán y otros en [55], investigaron los efectos de la adaptación en el rendimiento del clasificador en tres sistemas de BCIs diferentes y los resultados del estudio demuestran la necesidad de conocer la intención del usuario para determinar el rendimiento de la interfaz de adaptación y la necesidad de desarrollar métodos de adaptación no supervisada y los aspectos relacionados con el aprendizaje por refuerzo.

Vidaurre y otros en [85] simulan tres tipos de procedimientos de adaptación no supervisados basados en el análisis discriminante lineal LDA con una matriz de covarianza combinada. El primer método reduce el sesgo de los resultados de la clasificación basado en el promedio común actualizado. Las actualizaciones en el segundo método utiliza ambas matrices de covarianza y de promedios común, y tercer método aplica una rotación en el espacio de características. Los tres métodos tratan de reducir las tasas de error, en comparación con los sistemas sin ninguna adaptación.

Llera y otros en [51] presentan un método para el uso de un clasificador adaptativo de error global como un amplificador de señal para modificar los parámetros del clasificador. El clasificador utilizado para este fin fue un conjunto de redes neuronales.

Artusi y otros en [3], proponen la inclusión de errores en un sistema basado en los movimientos relacionados a los potenciales corticales (MRCP del inglés, *Movement Related Corticals Potentials*) asociados con la velocidad de una tarea. La exactitud se cuantificó, mediante la corrección del error global a la salida del clasificador. Se realizó una simulación de actuación del sistema en realimentación para evaluar su capacidad de adaptarse a los cambios en el estado mental del usuario. Para el ajuste de parámetros con los valores determinados experimentalmente en los simuladores, se ha demostrado que el aprendizaje para actualizar el conjunto de ejemplos se calcula basándose en la decodificación correcta del error potencial y la convergencia conduce a la solución óptima.

Faller y otros en [28], proponen la auto-calibración de la desincronización relacionada a evento (ERD del inglés, *Event Related Desynchronization*) de una BCI en línea. Esta interfaz proporciona el aprendizaje co-adaptativo basado en un número de nuevas características. El sistema proporciona inteligencia en la retroalimentación de refuerzo positivo, es decir, el paradigma sólo necesitan unos pocos minutos

de ejecución, cuando los primeros parámetros de características específicas pueden ser identificadas, a intervalos regulares, este sistema lleva a cabo la selección de seis bandas de frecuencia estándar, que tienen el más alto nivel de discriminabilidad. El sistema entonces rehace el entrenamiento del clasificador por el LDA y actualiza el modelo en el sistema en línea. Este sistema requiere pocas presuposiciones y sin calibración manual, ya que la mayoría de los parámetros se seleccionan en línea. La eficacia del concepto y la aplicación de este sistema se muestra en un estudio con dos clases en línea ERD- BCI, en diez de los doce usuarios novatos alcanzó niveles de 70 % de acierto entre una y tres sesiones de entrenamiento.

Samek y otros en [74], discuten la cuestión de que muchas extensiones de CSP se han propuesto como objetivo de reducción de la sensibilidad al ruido o incorporar información de otras características, pero ninguno de ellos aborda directamente el problema de la no estacionariedad. En este artículo, se presenta un enfoque de solución basado en CSP para subespacios estacionarios, se extraen características que no varían ante variaciones en las propiedades de la señal. Esto, por lo tanto, es un método que regulariza para subespacios estacionarios CSP (SCSP) y muestra que esto mejora la precisión de la clasificación, sobre todo para las personas que apenas son capaces de controlar una BCI.

2.2 Discusión

En la Tabla 2.1 son presentados los estudios que fueron tomados como base para el análisis de aBCI, cada estudio es discriminado identificando el módulo donde se realiza el proceso de adaptación, se discrimina el número de clases y la tasa de acierto obtenida. De la tabla anterior se puede inferir que en el desarrollo de interfaces adaptativas los procesos de adaptación tienen lugar principalmente, en el bloque de clasificación. Los autores muestran que el entrenamiento en línea favorece la interacción con el usuario. En las aBCI que requieren el uso de pontenciales de error tienen una gran limitante, ya que la BCI está obligada a tener una alta tasa de acierto en la detección del error, para obtener el resultado final deseado.

2. ESTADO DE LA TÉCNICA DE LAS INTERFACES CEREBRO COMPUTADOR ADAPTATIVAS

Tabla 2.1: Relación de las principales características de las interfaces adaptativas

Autor	Ext. Características	Módulo de adaptabilidad			Monit. Usuario	# Clases	Tasa Acierto
		Clasificador	Entrenamiento	Clasificador			
Shenoy et al, 2006			X		X	2	75 %
Buttfield et al, 2006		X			X	2	72 %
Vidaurre et al, 2006	X	X					75 %
Millán et al, 2007		X	X			2	84.3 %
Vidaurre et al, 2008	X	X				2	85 %
Diez et al , 2011	X					5	59 %
Lleras et al , 2011		X			X	2	70 %
Artusi et al, 2011		X			X	2	76 %
Faller et al 2012			X			2	80.2 %
Samek et al 2012	X					2	80 %
este trabajo	X	X	X		X	4	74 %

2.3 Tendencias

Las investigaciones sobre BCI están trabajando en nuevas modalidades sensoriales que mejoran la resolución temporal y espacial en la adquisición de las señales. También, la investigaciones están siendo dirigidas entorno de las estrategias para procesar la seña de la BCI, de tal forma que los trabajos están enfocados en la mejora de los procesos de filtrado (espacial y temporal); máquinas de aprendizaje y adaptación de señales para cada usuario a lo largo del tiempo. La tendencia es integrar una BCI en una aplicación final para el usuario, donde los aspectos como la relación hardware-software y la inclusión en ambientes reales y virtuales sean intuitivas y amigables. El uso de clasificadores adaptativos basados en aprendizaje supervisado también apunta en la dirección de las tendencias futuras, incluyendo la combinación de los potenciales de error y aprendizaje activo necesarios para crear un BCI con un gran potencial.

2.3.1 Limitaciones presentes en la aBCIs

Dada la diversidad y complejidad de las necesidades humanas, la BCI se perfila como una herramienta que puede mejorar la calidad de vida y satisfacer las necesidades básicas, tales como entretenimiento, movilidad, control de dispositivos, etc. De ahí la gran importancia en la mejora de diversos aspectos de las BCIs para lograr una mayor penetración en el mercado y la facilidad de uso cotidiano para las personas con o sin necesidades especiales. La incorporación de un canal directo de comunicación con el cerebro puede proporcionar un grado de libertad y extender las capacidades humanas, pero para lograr esta tarea, se necesita un equipo flexible y personalizado para las necesidades de cada usuario. Las interfaces adaptativa aun se encuentran muy lejos de esos ideales, debido a que el número de tareas que se pueden discriminar es bajo (<4). La inclusión de un bloque funcional encargado de monitoriar la actividad cerebral del usuario para inferir algún tipo de comportamiento o reacción producto de la interacción con la interfaz, es claramente una de las grandes limitaciones.

2. ESTADO DE LA TÉCNICA DE LAS INTERFACES CEREBRO COMPUTADOR ADAPTATIVAS

2.4 Conclusión

Las tendencias indican que la búsqueda de adaptatividad de las interfaces incluyen mejoras en los paradigmas de entrenamiento y clasificación. El aprendizaje activo y la integración de un modelo de usuario (inferir conocimiento del estado del usuario) son prometedoras opciones para lograr reducir la brecha entre lo que se puede hacer con las interfaces y lo que existe en la actualidad. Muchos grupos de investigación han presentados sus propuestas con relación a mejorar la interacción humano-computador, logrando mejorar las tasas de acierto e incluyendo los potenciales de error como mecanismo de corrección o actualización de parámetros del clasificador.

*Lo que sabemos es una gota de
agua, lo que ignoramos es el
océano.*

Isaac Newton

CAPÍTULO 3

Métodos para la implementación de una BCI

Varias etapas de procesamiento tienen que ser completadas en el uso de la señal EEG para que una BCI pueda verificar el estado actual del usuario y generar un comando de salida. Estas etapas son descritas en detalle en este capítulo, en la figura 3.1, se muestra en diagrama de bloques el esquema general de una BCI. El bloque relacionado con la adquisición de señales hace la conversión de los impulsos eléctricos a señales digitales. Los impulsos generados en la corteza cerebral y capturados en el cuero cabelludo, son adquiridos utilizando un gorro con unos electrodos y amplificados electrónicamente, esta técnica es denominada Electroencefalografía (**EEG** del inglés, *Electro-EncephaloGraphy*). En las secciones siguientes se explicará en detalle las diferentes etapas que constituyen el bloque de procesamiento. Finalmente, el bloque de aplicación son todas aquellas herramientas, equipos o sistemas que utilizan la salida generada por la BCI, entre ellas destacan: sillas de ruedas robóticas, sistemas de entretenimiento, realidad virtual, aplicaciones para rehabilitación, etc.

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

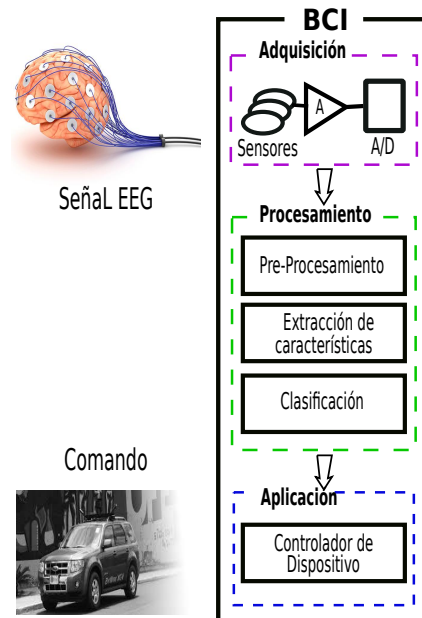


Figura 3.1: Esquema general de una BCI

3.1 Pre-procesamiento

Esta etapa tiene la finalidad de mejorar la relación senal/ruido, para ese fin se implementan diferentes tipos de filtros y aplicación de técnicas para la remoción de artefactos.

3.1.1 Filtros Espaciales

La selección adecuada de un filtro espacial para cualquier BCI se determina por la localización y extensión de la señal de control cerebral seleccionada y de las diversas fuentes de EEG o ruido no EEG.

3.1.1.1 Promedio de la referecia común

El promedio de la referecia común (CAR del inglés, *Common Average Reference*) ha sido sugerido como una solución a el problema del electrodo de referencia [10]. El CAR supone el registro de manera bipolar a partir de un número de electrodos, todo referido a un solo sitio. Para obtener el CAR se calcula la media de toda la

señal EEG, promediando através de los electrodos, y se resta el resultado puntual de cada electrodo.

3.1.1.2 Laplaciano de superficie

El laplaciano de superficie (SL del inglés, *Surface Laplacian*) se define como la derivada espacial de segundo orden del potencial de superficie [62]. Debido a sus características intrínsecas de filtrado espacial paso alto, el SL puede reducir el efecto de conducción mediante la mejora de los componentes espaciales de alta frecuencia, por lo tanto, se puede conseguir una mayor resolución espacial de los potenciales de superficie.

3.1.2 Análisis de componentes independientes

Los artefactos más importantes de las BCIs son generadas por los músculos, pestañeo y movimientos oculares [89]. Los Métodos automáticos clásicos para la eliminación de este tipo de artefactos se pueden clasificar en métodos de rechazo y métodos de sustracción:

- Métodos de rechazo, consisten en descartar el EEG contaminado, ya sea sobre la base de detección automática o visual, su éxito depende fundamentalmente de la calidad de la detección.
- Métodos de sustracción, se basan en la suposición de que el EEG contaminado es una combinación lineal de otras señales de artefactos independientes generada por los músculos y el parpadeo, y un EEG original.

El EEG original se recupera por lo tanto quitando señales relacionadas a los artefactos grabados por separado de la medida de la señal EEG.

3.2 Extracción de Características

Con el fin de seleccionar el clasificador más adecuado para un determinado sistema de BCI, es esencial entender claramente qué características se utilizan, cuáles son sus propiedades y cómo son utilizadas. En este apartado se describen los rasgos comunes de las BCIs y más particularmente sus propiedades. Una gran variedad

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

de características se han utilizado para diseñar BCIs, tales como la amplitud de los valores de las señales EEG [44], densidad espectral de potencia (PSD) [20, 57], parámetros autorregresivo (AR) y autorregresivos adaptativos (AAR) [65, 67], características de tiempo-frecuencia [88] e inversa basado en el modelo las características [23].

Sobre el diseño de un sistema BCI, algunas de las propiedades fundamentales de estas características deben ser consideradas:

- El ruido y los valores extremos: las BCIs son ruidosas o contienen valores extremos, porque las señales de EEG tienen una pobre relación señal/ruido.
- La alta dimensionalidad: en los sistemas BCIs, los vectores de características son a menudo de alta dimensionalidad. Se extraen varias características de varios canales y de varios segmentos de tiempo antes que se concatenan en un vector de características individuales.
- La no estacionariedad: las BCIs no son fijas ya que las señales EEG pueden variar rápidamente con el tiempo y más especialmente entre sesiones.
- Paradigmas de entrenamiento: el proceso de entrenamiento es largo y exigente para los sujetos, se requiere mejorar este aspecto para tornar más funcional y extensivo el tema de las BCIs.

3.2.1 Bandas de Potencia

Las características de banda de potencia (BP del inglés, *Band Power*) pueden ser extraídas de las señales de EEG mediante la estimación de la distribución de potencia en bandas de frecuencia predefinidas. En [67] se utilizó las BP y ha demostrado que diferentes componentes de frecuencia de la banda α y β proporcionaban una mejor discriminación entre la imaginación de movimientos de la mano izquierda y derecha. Estas bandas de frecuencias variaron entre 9 y 14 Hz y entre 18 y 26 Hz

3.2.2 Autoregresores

En estadística, un modelo paramétrico o modelo de dimensión finita se refiere a una familia de las distribuciones que se pueden describir utilizando un número finito de parámetros, estos parámetros se representan por único vector de parámetros k -dimensional $\theta = (\theta_1, \theta_2, \dots, \theta_k)$. El modelo paramétrico asume que la serie temporal analizada tiene una salida que puede ser representada por un modelo matemático lineal.

3.2.3 Coherencia

Entre los métodos matemáticos no paramétricos aplicados al análisis del EEG, ha cobrado especial importancia el análisis de la coherencia [29] o coeficiente de correlación de Pearson aplicado a la covarianza del análisis de Fourier. La covarianza permite establecer el grado de similitud entre 2 señales eléctricas, pero es directamente dependiente de la amplitud de las mismas. Para independizarla de la amplitud se la divide por la raíz cuadrada del producto de la varianza de cada señal. La covarianza normalizada se transforma en el coeficiente de correlación, conocido también como coherencia. El análisis de coherencia es la correlación lineal normalizada que existe entre dos pares de canales EEG e informa sobre los componentes frecuenciales comunes a ellos, junto con la magnitud de su asociación. Sus valores van de 0 (ninguna asociación) a 1 (asociación total). En las Ecuaciones 3.1 y 3.2 son presentadas las expresiones que describen la coherencia:

$$C_{XY}^2(t) = \frac{|R_{XY}(t)|^2}{R_{XX}(t)R_{YY}(t)}, \quad (3.1)$$

donde

$$R_{XY}(t) = \frac{1}{2W+1} \sum_{k=t-W}^{t+W} X(k)Y^*(k) \quad (3.2)$$

Los elementos de la función de correlación son: el espectro de potencia cruzada (la expresión de las armónicas que tienen en común ambas señales); el espectro de coherencia (que indica la simetría entre las señales); y el espectro de fase (una expresión de la sincronía de las señales).

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

3.2.4 Factor de Fase bloqueada

La activación coordinada sobre diferentes regiones del cerebro se puede medir utilizando la sincronización de fase. La sincronización de fase han sido ampliamente usada en los sistemas caóticos y en el campo de la dinámica no lineal. La detección de sincronización de fase puede proporcionar una indicación de sincronías de corto alcance, interpretado comúnmente como la unión perceptual entre regiones cerebrales adyacentes o las mismas regiones y puede describir patrones de sincronización de largo alcance entre las regiones del cerebro muy distantes entre sí, se cree que relaciona mecanismos cognitivos, como la memoria, las emociones y la planificación motora. El factor de fase bloqueada (PLF del inglés, *Phase Locking Factor*)[49] es definido como:

$$PLF_t = \frac{1}{N} \left| \sum_{n=1}^N \exp(j\theta(t, n)) \right|, \quad (3.3)$$

donde $\theta(t, n)$ es la diferencia de fase $\phi_1(t, n) - \phi_2(t, n)$, y n es el número de muestras.

3.2.5 Espectro de potencias basado en Wavelets

El análisis de espectro de potencias de Fourier de una sinfonía revelará, por supuesto, los tonos y sus armónicos, así como otras frecuencias que se repiten en modulaciones y vibraciones. Si tocamos las partes en otro orden, el espectro de potencias no cambia en absoluto, pero para el escucha se tratará de una pieza totalmente diferente, y más aun si intercambiamos partes dentro de las partes, a una escala más fina. Por el contrario, el análisis de wavelets no sólo nos da las frecuencias principales, sino que nos indica cuándo ocurren y cuál es su duración. La transformada de Wavelets fue diseñada originalmente para estudiar señales no estacionarias, se trata de un análisis de tiempo-frecuencia. Es capaz de revelar aspectos de los datos como tendencias, puntos de quiebre, discontinuidades en las derivadas, y auto-similaridad. La transformada Wavelet presenta mayor efectividad en la caracterización de señales EEG que las técnicas tradicionales [24, 81]. El espectro de potencia wavelet (WPS del inglés, *Wavelet Power Spectrum*) se calcula a través de una llamada transformada wavelet $\psi(t)$, que como una función del tiempo, se

puede definir como:

$$\psi(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right), \quad (3.4)$$

donde a denota un parámetro de escala para la frecuencia representada por la wavelet, y b un factor de desplazamiento, es decir, el punto central de la wavelet. Cuando se usan escalas arbitrarias entre los intervalos de muestreo que contienen la serie de tiempo, se hace referencia a la transformada wavelet continua (CWT) de una función de tiempo, $f(t)$, es expresada como:

$$W(f) = \int_{-\infty}^{+\infty} f(t) \psi(f) dt. \quad (3.5)$$

Para obtener el WPS es solo elevar al cuadrado los valores de la transformada Wavelet.

3.2.6 Transformada de Hilbert-Huang

Una señal puede ser representada por sus características temporales o frecuenciales; para la representación temporal, se registra su evolución en la escala de tiempo escogida y se almacena la información de su amplitud. Para la representación en el dominio de la frecuencia se utiliza la transformada de Fourier. Existe la posibilidad de representar esta misma señal almacenando información tanto en el dominio del tiempo como de la frecuencia, una transformada ampliamente utilizada para este fin es la Transformada de Hilbert (HT del inglés, *Hilbert Transform*) [73]. La HT de una señal $x(t)$ se define como:

$$HT(x(t)) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\tau)}{(t-\tau)} d\tau, \quad (3.6)$$

donde $HT(t)$ representa la convolución entre la señal $x(t)$ y una función $h(t) = \frac{1}{\pi t}$. Esta operación produce un desfase de -90° para frecuencias positivas y 90° para frecuencias negativas. Una señal analítica puede ser representada como la composición de una señal de valor real y la suma de una señal compleja proveniente de la señal real:

$$x_{sa} = x(t) + iHT(x(t)) \quad (3.7)$$

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

La señal analítica contiene sólo el espectro positivo de la transformada de Fourier de la señal original. La señal analítica puede reescribirse como:

$$x_{sa} = a(t)e^{i\theta(t)}, \quad (3.8)$$

donde la amplitud es $a(t) = |x_{sa}(t)|$ y el ángulo es $\theta(t) = \arctan(HT(x(t))/x(t))$. De la ecuacion 3.8, $a(t)$ es la amplitud instantánea de la señal $x(t)$, y $\theta(t)$ representa la fase. La frecuencia instantánea $w(t)$ puede ser definida como :

$$w(t) = \frac{d\theta(t)}{dt} \quad (3.9)$$

3.2.7 Características Estadística de series de tiempo

Una serie temporal es una secuencia de datos, observaciones o valores, medidos en determinados momentos y ordenados cronológicamente. Para el análisis de las series temporales se usan métodos que ayudan a interpretarlas y que permiten extraer información representativa sobre las relaciones subyacentes entre los datos de la serie o de diversas series. Las características estadísticas de las series de tiempo (SFTS del inglés, *Statistical Features Times of Series*) describen los métodos que permiten realizar un análisis del estado actual de la señal [68]. Las características más relevantes son:

1. La media de las señales, esta es presentada en la ecuacion 3.10

$$\mu x = \frac{1}{N} \sum_{n=1}^N X_n, \quad (3.10)$$

donde N corresponde al número de elementos y X_n es la señal de entrada.

2. La desviacion estandar de la señal:

$$\sigma x = \left(\frac{1}{N-1} \sum_{n=1}^N (X_n - \mu x)^2 \right)^{(1/2)}. \quad (3.11)$$

3. La media de los valores absolutos de la primera diferencia de la señal:

$$\delta x = \frac{1}{N-1} \sum_{n=1}^{N-1} |X_{n+1} - X_n|, \quad (3.12)$$

3.3 Métodos de selección de características y reducción de dimensionalidad

4. La media de los valores absolutos de la segunda diferencia de la señal:

$$\gamma x = \frac{1}{N-2} \sum_{n=1}^{N-2} |X_{n+2} - X_n|. \quad (3.13)$$

Las características representadas en las ecuaciones 3.10, 3.11, 3.12 y 3.13 son ampliamente utilizadas en la identificación fisiológica de las emociones [87]. Una de las ventajas de estas características está relacionada a la facilidad y el bajo costo computacional para su implementación, lo cual lo vuelve muy apropiado para implementaciones en tiempo real en aplicaciones BCIs. Sin embargo, las características estadísticas no explotan el conocimiento que subyace de las fuentes físicas de las señales, pues están fuertemente afectadas por los procesos de normalización, filtrado, factores relacionadas a la conductividad de los electrodos, la gel y variaciones inter e intra-usuario.

3.3 Métodos de selección de características y reducción de dimensionalidad

En aplicaciones BCIs, varias características son extraídas, de varios canales que registran la actividad cerebral (varios electrodos en el caso de las mediciones de EEG) y de varios segmentos (o sesiones), antes de ser concatenados en un único vector de características. Por lo tanto, las BCIs son a menudo afectadas por un problema conocido como la maldición de la dimensionalidad [7]. La maldición de la dimensionalidad consiste en que la cantidad de datos necesarios para describir adecuadamente las diferentes clases, aumenta exponencialmente con la dimensionalidad de los vectores de características [31, 42]. En [30] y [66] han demostrado que cuando se utiliza la función de selección de características, la exactitud de la clasificación es mejor que cuando se utilizan todas las características. Si el número de datos de entrenamiento es relativamente pequeño para el número de características, los algoritmos de clasificación que utilicen estos datos muy probablemente entregarán malos resultados, es recomendable el uso de al menos 5 a 10 veces más de datos de entrenamiento por clase que el número de características [69], por desgracia esto no puede ser aplicado en todos los sistemas BCIs con alta dimensionalidad.

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

De ahí, la necesidad de métodos de selección de características y de reducción de dimensionalidad en los diseños de BCI. Aquí se presenta un breve resumen y las definiciones de los métodos que son los más aplicados en diseños BCIs:

3.3.1 Selección secuencial hacia adelante

Es una técnica de búsqueda abajo-arriba. Selecciona primero la mejor variable según algún criterio ζ , luego se combina la variable original con cada una de las variables restantes, entonces, se busca la pareja que aporta el mayor valor de evaluación y se escoge como nuevo conjunto de partida. A continuación se combina esta pareja con cada una de las variables restantes, formando ternas, se selecciona la terna que dé un mayor valor en el criterio de evaluación. El proceso se repite una y otra vez en la misma forma. La búsqueda se detiene cuando un conjunto de más variables no mejore los resultados de la función de costo para un conjunto de menos variables.

3.3.2 Selección secuencial hacia atrás

Al igual que la técnica anterior, la idea es construir conjuntos diferentes iteración tras iteración, con la diferencia que ahora se inicia con el conjunto completo de características de dimensión p , y en cada iteración se remueve una variable. La variable que se elimina es aquella que al no estar presente en el subconjunto a evaluar, hace que la función de costo reporte el mayor valor entre todos los subconjuntos evaluados en la misma iteración. El algoritmo se detiene cuando el valor de ζ no supera cierta cota preestablecida. Entonces, se selecciona el último subconjunto que al ser evaluado haya superado el umbral.

3.3.3 Selección secuencial flotante

A diferencia de los casos anteriores, este procedimiento permite tanto adicionar como eliminar características al subconjunto ya elegido. Básicamente, en una iteración $\hat{k}$, se adiciona la característica que maximice el criterio ζ , posteriormente se elimina del subconjunto actual aquella variable que más reduzca el valor de ζ , si es la última variable agregada, entonces el conjunto no se modifica y se adiciona

3.3 Métodos de selección de características y reducción de dimensionalidad

una nueva variable; en caso contrario se remueve la característica del subconjunto y se continúan removiendo características siempre y cuando ζ no decrezca. Luego se agrega nuevamente una característica y se continúa el proceso. La búsqueda se detiene cuando al ζ no es mejorado.

3.3.4 Selección basado en la distancia de Silhouette

La anchura de Silhouette (SW del inglés, *Silhouette's Width*) permite relacionar la cohesión y la separación para cluster. La SW se refiere a un método para la interpretación y validación de cluster [72]. En este índice se refleja la cohesión; medida distancia intracluster y separación; medida de distancia intercluster, y puede ser aplicado a diversas métricas. En la figura 3.2 puede observarse una representación de la medida de cohesión y separación para un cluster, visto desde la perspectiva de los datos. Para cada *trial* i , su SW $s(i)$ es definida como:

$$s(i) = \begin{cases} 1 - a(i)/b(i), & \text{if } a(i) < b(i) \\ 0, & \text{if } a(i) = b(i) \\ b(i)/a(i) - 1, & \text{if } a(i) > b(i), \end{cases} \quad (3.14)$$

donde $a(i)$ es el promedio de la distancia del *trial* i para los otros *trials* en el mismo cluster, $b(i)$ es la distancia promedio del *trial* i hacia los *trials* de las clusters vecinos. El promedio de $s(i)$ a través de todos los *trials* refleja la calidad del cluster resultante. La SW como índice de desempeño puede permitir evaluar que

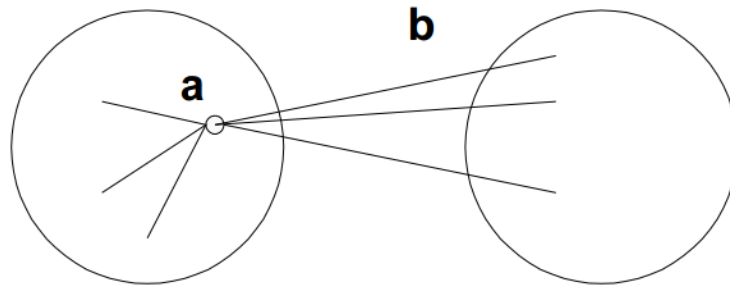


Figura 3.2: Representación de la SW

características proveen el mejor desempeño (exactitud) en una tarea mental para ser implementado en una BCI [18].

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

3.3.5 Algoritmo Genético

Las técnicas de la teoría de la información son habituales en la selección de las variables en la predicción de series de tiempo o patrones de reconocimiento. La maximización de la información mutua entre la entrada y salida de datos es un procedimiento que requiere un esfuerzo computacional alto, debido al cálculo de la entropía, donde es necesario realizar la estimación de las distribuciones de probabilidad de los diferentes conjuntos y subconjunto de datos. Para evitar este esfuerzo computacional, es posible aplicar la selección de variable basado en el principio de mínima-redundancia/máxima relevancia, lo que maximiza la información mutua, presentando el menor costo computacional posible. Sin embargo, el problema de optimización es de naturaleza combinatoria, es decir, comprobar todas las posibles combinaciones de variables todavía representa un gran esfuerzo computacional [53].

Los algoritmos genéticos [30] (GA del inglés, *Genetic Algorithm*), se basan en los principios evolutivos, donde los subconjuntos de características están codificados en la forma de secuencias simples que se consideran el genoma de los individuos de una población. La población cambia de acuerdo con la reproducción de sus individuos. Para la reproducción, los operadores como la mutación y los cruces son aplicados a la población. La aptitud de los individuos está representada por el desempeño de la clasificación del correspondiente subconjunto de características y determina la posibilidad de la reproducción. Durante varias generaciones, la idoneidad de la población y sus individuos mejora. Cuando se cumple el criterio se presenta una parada, y se selecciona el subconjunto de característica que representan los más aptos. Los GAs son estrategias de optimización que no asumen un espacio de búsqueda continuamente diferenciable. En una población, los subconjuntos de características presentes son cubiertos inicialmente por búsquedas al azar [38].

3.3.6 Análisis de componentes principales

El análisis de componentes principales (PCA del inglés, *Principal Component Analysis*) se puede utilizar en la etapa de pre-procesamiento de los diseños de BCI y también se puede utilizar como una técnica de reducción de dimensionalidad. En

términos de la captura de la varianza de los datos, el PCA transforma un conjunto de m variables en otro conjunto de $k \leq m$ variables de no correlacionadas, manteniendo la mayor cantidad de la varianza de los datos originales [59]. El modelo de transformación está dado por : $Z = XW$, donde z es la matriz $(n \times m)$ de observaciones en el espacio transformado, X es la matriz $(n \times p)$ de observaciones del espacio original y W es la matriz $(p \times m)$ de transformación que corresponde a los m vectores propios de la matriz de covarianza o correlación asociados a los m mayores valores propios. El algoritmo [1] resume el procedimiento para PCA.

Algoritmo 1. Procedimiento para la implementación del PCA

- 1: Centralizar la matriz de datos X .
 - 2: Obtener la matriz de covarianza $S = \frac{1}{n} X^T X$.
 - 3: Calcular los valores propios de la matriz S y sus valores propios.
 - 4: Ordenar descendientemente los valores propios.
 - 5: Proyectar los datos sobre las componentes principales (valores propios ordenados).
-

3.3.7 Divergencia del Kullback Leibler

En teoría de la probabilidad y teoría de la información, la divergencia de Kullback-Leibler (KL) (también conocida como divergencia de la información, ganancia de la información, entropía relativa) es una medida no simétrica de la similitud o diferencia entre dos funciones de distribución de probabilidad P y Q . Aunque a menudo se considera como una métrica o distancia, la divergencia KL no lo es en realidad — por ejemplo, no es simétrica: la divergencia KL de P a Q no necesariamente es la misma KL de Q a P . En [8] fué implementada una modificación para obtener un KL simétrico y ser usado como método para la selección de características para una BCI basada en tareas mentales.

3.4 Clasificadores

Con el fin de elegir el clasificador más apropiado para un determinado conjunto de características, las propiedades de estos deben ser conocidas. En esta sección se ofrece una descripción de los clasificadores. Entiendase, el proceso de clasificación como aquel que permite mapear un conjunto de datos de entrada a un espacio para que las etiquetas de las clases sean aprendidas. La clasificación tiene como función

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

decidir cual es la clase a la que pertenece un dato, sin etiqueta, dependiendo de sus características. Para llevar a cabo esta decisión, un clasificador que incorpore el conocimiento *a priori* necesita ejemplos etiquetados. La adquisición de este conocimiento se llama fase de aprendizaje de máquina o fase de entrenamiento del clasificador. Los siguientes son los clasificadores ampliamente utilizados en las BCIs:

3.4.1 Análisis de discriminante Lineal

El objetivo de la LDA (del inglés, *Linear Discriminant Analysis*), también conocido como LDA Fisher, es el uso de hiperplanos para separar los datos que representan las diferentes clases [27]. Para un problema de dos clases, la clase de un vector de características depende de donde se ubique el vector con respecto del hiperplano. El hiperplano separador se obtiene mediante la búsqueda de la proyección que maximiza la distancia entre las medias de las clases y minimiza la varianza interclase. Para resolver un problema de la clase N ($N > 2$) se utilizan varios hiperplanos. La estrategia utilizada para BCIs multiclase es el único versus el resto (OVR), esta estrategia consiste en separar cada clase de todas las demás. Esta técnica tiene un requerimiento computacional muy bajo, lo que la hace adecuada para un sistema BCI en línea. Además, este clasificador es fácil de usar y generalmente entrega buenos resultados, en consecuencia, el LDA ha sido utilizado con éxito en un gran número de sistemas BCI, como las BCI basado en imaginación motora [61], multiclase [32]. El principal inconveniente de LDA es su linealidad, la cual puede ofrecer pobres resultados con datos complejos y no lineales de un EEG. El LDA Fisher regularizado (RFLDA) también se ha utilizado en BCI [12], este introduce un parámetro de regularización que permite sancionar los errores de clasificación en el conjunto de entrenamiento. El clasificador resultante puede ajustar los valores atípicos y obtener una mejor capacidad de generalización. Como los valores extremos son comunes en los datos de EEG, esta versión regularizada de LDA puede dar mejores resultados que la versión de las BCIs no regularizadas [13].

3.4.2 K vecinos cercanos

El método K-NN (del inglés, *K-Nearest Neighbors*) [27] es un método de clasificación supervisada que sirve para estimar la función de densidad $F(x/C_j)$ de las predictoras x por cada clase C_j . Este es un método de clasificación no paramétrico, que estima el valor de la función de densidad de probabilidad o directamente la probabilidad *a posteriori* de que un elemento x pertenezca a la clase C_j a partir de la información proporcionada por el conjunto de prototipos. En el proceso de aprendizaje no se hace ninguna suposición acerca de la distribución de las variables predictoras. En el reconocimiento de patrones, el algoritmo K-NN es usado como método de clasificación de objetos (elementos) basado en un entrenamiento mediante ejemplos cercanos en el espacio de los elementos. El K-NN es un tipo de "Lazy Learning", donde la función se aproxima solo localmente y todo el cálculo es diferido a la clasificación. Los ejemplos de entrenamiento son vectores en un espacio característico multidimensional, cada ejemplo está descrito en términos de p atributos considerando q clases para la clasificación. Los valores de los atributos del i –esimo ejemplo (donde $1 \leq i \leq n$) se representan por el vector p –dimensional

$$x_i = (x_{1i}, x_{2i}, \dots, x_{pi}) \in X \quad (3.15)$$

El espacio es particionado en regiones por localizaciones y etiquetas de los ejemplos de entrenamiento. Un punto en el espacio es asignado a la clase C_j si esta es la clase más frecuente entre los k ejemplos de entrenamiento más cercanos. Generalmente se usa la distancia euclidiana:

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^p (x_{ri} - x_{rj})^2} \quad (3.16)$$

La fase de entrenamiento del algoritmo consiste en almacenar los vectores característicos y las etiquetas de las clases de los ejemplos de entrenamiento. En la fase de clasificación, la evaluación del ejemplo (del que no se conoce su clase) es representada por un vector en el espacio característico. Se calcula la distancia entre los vectores almacenados y el nuevo vector, y se seleccionan los k ejemplos más cercanos. El nuevo ejemplo es clasificado con la clase que más se repite en los vectores seleccionados.

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

Este método supone que los vecinos más cercanos dan la mejor clasificación y esto se hace utilizando todos los atributos; el problema de dicha suposición es que es posible que se tengan muchos atributos irrelevantes que dominen sobre la clasificación: dos atributos relevantes perderían peso entre otros veinte irrelevantes.

Para corregir el posible sesgo se puede asignar un peso a las distancias de cada atributo, dándole así mayor importancia a los atributos más relevantes. Otra posibilidad consiste en tratar de determinar o ajustar los pesos con ejemplos conocidos de entrenamiento. Finalmente, antes de asignar pesos es recomendable identificar y eliminar los atributos que se consideran irrelevantes.

3.4.3 Redes Neuronales

Las Redes Neuronales (NN del inglés, *Neural Networks*) son junto con los clasificadores lineales, la categoría de clasificador que más se utiliza en las BCI [52]. Recordemos que un NN es un conjunto de varias neuronas artificiales que permite producir los límites lineales de decisión. En esta sección son descritas las NN más utilizadas en el ámbito de las BCIs.

3.4.3.1 Perceptrón multicapa

Un perceptrón multicapa (MLP del inglés, *Multilayer Perceptron*) se compone de varias capas de neuronas: una capa de entrada, posiblemente una o varias capas ocultas y una capa de salida. La entrada de cada neurona se conecta con la salida de las neuronas de la capa anterior, mientras que las neuronas de la capa de salida se encargan de determinar la clase de la función de vector de entrada. Las Redes Neuronales y por lo tanto MLP, son aproximadores universales, es decir, cuando se compone de neuronas y capas suficientes, se puede aproximar cualquier función continua. Añadido al hecho de que se puede clasificar cualquier número de clases, esto hace que NN sea un clasificador muy flexible que puede adaptarse a una gran variedad de problemas. En consecuencia, las MLP, son los esquemas de clasificación más populares aplicados a casi todos los problemas de BCI, tales como binarios o multiclase [2], sincrónico [36] o asíncrono [20]. Sin embargo, el hecho de que MLP sean aproximadores universales hace que a estos clasificadores sean sensibles a la sobre-entrenamiento, especialmente con datos tan ruidoso y no estacionarios como

el EEG[5]. Por lo tanto, la selección cuidadosa de la arquitectura y la regularización es necesaria [42]. Un perceptrón multicapa, sin capas ocultas que se conoce como un perceptrón. Curiosamente, un perceptrón es equivalente a la LDA y, como tal, ha sido a veces utilizado para aplicaciones de BCI [23].

3.4.3.2 Máquinas de aprendizaje extremo

Las máquinas de aprendizaje extremo (ELM del inglés, *Extreme Learning Machines*) han atraído la atención de más y más investigadores como una técnica emergente que resuelve algunos problemas a los que se enfrentan otras técnicas. ELM trabaja en redes mono capa hacia delante o de tipo “feedforward”. La esencia del ELM es que la capa oculta de la red no necesita ser entrenada. Comparado con las técnicas de inteligencia computacional tradicionales, ELM mantiene la capacidad de generalización y un entrenamiento mucho más rápido con menor intervención humana [6, 21, 39].

3.4.3.3 Aprendizaje de vectores de cuantización

El aprendizaje de vectores de cuantización (LVQ del inglés, *Learning Vector Quantization*) [48], es una técnica en la cual el espacio de entradas es dividido en un número determinado de regiones y para cada una de ellas es definido un vector que la caracteriza. La entrada al sistema son vectores n –dimensionales y la salida es una representación discreta del espacio de entradas. El LVQ es un método para el entrenamiento de capas competitivas de una manera supervisada. Una capa competitiva aprenderá a clasificar automáticamente vectores de entrada. Sin embargo, las clases que la capa competitiva encuentra dependen únicamente de los vectores de entrada. Si dos vectores de entrada son muy similares, la capa competitiva probablemente los colocara en la misma clase. No existe mecanismo en el diseño de una capa competitiva para dictar si cualquiera de los vectores de entrada están la misma clase o no.

3.4.3.4 Teoría resonancia adaptativa

La teoría de la resonancia adaptativa (ARTMAP del inglés, *Adaptive Resonance Theory Map*) [16] fue desarrollada por Stephen Grossberg y Gail Carpenter, es un

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

modelo de red neuronal artificial que basa su funcionamiento en la manera en que el cerebro procesa información y que describe una serie de modelos de redes neuronales que utilizando métodos de aprendizaje supervisado y no supervisado, abordan problemas tales como el reconocimiento y la predicción de patrones. Grossberg y Carpenter desarrollaron la teoría de resonancia adaptativa en respuesta al “dilema de la estabilidad y plasticidad del aprendizaje”, en el que se plantean las siguientes cuestiones:

1. Plasticidad del aprendizaje. Permite a una red neuronal aprender nuevos patrones.
2. Estabilidad del aprendizaje. Permite a una red neuronal retener los patrones aprendidos.
3. Conseguir que un modelo de red neuronal sea capaz de resolver uno solo de estos problemas es sencillo, el reto está en conseguir un modelo que sea capaz de dar respuesta a ambos.

Las redes más conocidas, tales como el Perceptrón multicapa, son capaces de aprender cómo han de responder ante unos patrones de entrada pero, una vez entrenados, el intentar que aprendan nuevos patrones puede suponer el “olvido” de lo aprendido previamente.

La red neuronal basada en la teoría de la resonancia adaptativa, es un clasificador de vectores. Un vector de entrada se clasifica dentro de una de un número de categorías, dependiendo de la similitud con los patrones previos. Si el criterio de similitud no se satisface se crea una nueva categoría. Así, la red neuronal con base en la teoría de la resonancia adaptativa resuelve el dilema entre plasticidad y estabilidad al habilitar que el aprendizaje ocurra solamente en un estado de resonancia. El modelo asociado a la teoría de la resonancia adaptativa soluciona el dilema de la estabilidad y plasticidad del aprendizaje mediante un mecanismo de realimentación entre las neuronas competitivas de la capa de salida. Cuando a la red se le presenta un patrón de entrada este se hace resonar con los prototipos de las categorías conocidas por la red, si el patrón entra en resonancia con alguna clase entonces es asociado a esta y el centro de cluster es desplazado ligeramente para adaptarse mejor al nuevo patrón que le ha sido asignado. En caso contrario, si el

patrón no entra en resonancia con ninguna clase, pueden suceder dos cosas: si la red posee una capa de salida estática entrará en saturación pues no puede crear una nueva clase para el patrón presentado pero tampoco puede asignarlo a una clase existente, si la red posee una capa de salida dinámica se creará una nueva clase para dicho patrón, esto no afectará a las clases ya existentes.

Como resultado de este enfoque se presentaron dos redes neuronales especialmente adecuadas para las tareas de clasificación de patrones. Estas redes suelen denominarse ART1 (o ART) y ART2. Ambas difieren de la naturaleza de la información que se presenta a la entrada. La red ART1 trabaja con vectores de entrada binarios mientras que ART2 es capaz de procesar informaciones continuas o analógicas. ARTMAP[17], conocido como ART predictivo, combina las funcionalidades modificadas de ART-1 o ART-2 en una estructura de aprendizaje supervisado donde la primera unidad toma los datos de entrada y la segunda unidad toma los datos de salida, luego se hacen los ajustes de parámetros para que el sistema presente la mejor clasificación posible.

3.4.3.5 Otras arquitecturas de redes neuronales

Otros tipos de arquitectura de redes neuronales se utilizan en los sistemas BCIs. Entre ellos, una merece una atención especial en su uso para las BCI: es el clasificador de Gauss [56, 57]. Cada unidad de esta red es una función discriminante de Gauss, la cual representa un prototipo de la clase. Según sus autores, esta supera a MLP en los datos y la BCI puede realizar un rechazo eficiente de las muestras de incertidumbre [56]. Como consecuencia de ello, este clasificador se ha aplicado con éxito a la imaginación motora y clasificación de tareas mentales, particularmente durante los experimentos asíncronos [56]. Además de los clasificadores de Gauss, otras varias redes han sido aplicadas en BCIs, de una forma más marginal, que no se describen aquí, debido a limitaciones de espacio: redes neuronales dinámicas, tales como la Red de Respuesta Finita al Impulso (FIRNN) [36], Redes neuronales de tiempo de retardo (TDNN), Bayesiano de regresión logística (red neuronal BLRNN) [65], entre otras.

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

3.4.4 Máquinas de vectores de soporte

La máquina de vectores de soporte (SVM del inglés, *Support Vector Machine*) [83] es un sistema de aprendizaje para construir clasificadores y funciones de regresión lineales y no lineales. La SVM es una técnica no paramétrica que construye la solución de forma explícita mediante una combinación lineal de las muestras de entrenamiento. La característica más relevante de la SVM es su capacidad para resolver problemas en los que los datos son de gran dimensionalidad, sin degradar la solución por la falta de éstos. Un SVM utiliza un hiperplano discriminante para identificar las clases [9]. Sin embargo, en relación con SVM, el hiperplano seleccionado es el que maximiza los márgenes, es decir, la distancia entre los puntos más cercanos. Un SVM utiliza un parámetro de regularización que permite el alejamiento de los valores atípicos y permite errores en el conjunto de entrenamiento. La SVM permite clasificación con límites lineales de decisión, y se conoce como SVM lineal. Este clasificador se ha aplicado con éxito, a un número relativamente grande de problemas de BCI sincrónica [13]. Sin embargo, es posible crear fronteras lineales de decisión, con sólo un pequeño aumento de la complejidad del clasificador, utilizando la figura del kernel. Este consiste en trasladar de forma implícita la cartografía de los datos a otro espacio, en general, de mucha mayor dimensión, con una función del núcleo $k(x, y)$ el kernel utilizado generalmente en BCI es la función de Gauss o de base radial. La SVM correspondiente se conoce como SVM Gauss o SVM RBF [9]. Como la LDA, la SVM ha sido aplicada a problemas multiclase, mediante la estrategia de OVR. Una SVM tienen varias ventajas, gracias a la maximización del margen y el término de regularización, la SVM tienen buenas propiedades de generalización [9, 42], al ser menos afectado por el sobreentrenamiento [42] y el problema de la dimensionalidad [9].

3.4.5 Validación cruzada

Para obtener las medidas de desempeño de un clasificador se debe evaluar la capacidad que este tenga de generalizar los datos con los cuales cuenta el sistema. Dado que no se cuenta con un número elevado de patrones para validar el clasificador, se divide el conjunto de muestras en k particiones de tamaño $Kp = nMuestras/k$, donde $nMuestras$ es el número total de patrones. Una partición es usada como

conjunto de validación, mientras $k - 1$ particiones son usadas como conjunto de entrenamiento. Los pasos para evaluar el clasificador basado este método son:

1. Dividir aleatoriamente el conjunto de muestras en k particiones.
2. Seleccionar una de las k particiones para usar como conjunto de validación.
3. Usar las restante $k - 1$ particiones para entrenar el clasificador.
4. Presentar el conjunto de validación al clasificador y obtener las etiquetas asignadas a estas muestras.
5. Determinar el error de clasificación.
6. Repetir los pasos mencionados anteriormente hasta que las k particiones hayan sido usadas como conjunto de validación.
7. Calcular el valor promedio de los $k - errores$

Este método posee como ventaja que es poco sensible a la partición de los datos, ya que cada uno de ellos logra estar en el conjunto de validación al menos una vez, y $k - 1$ veces en el conjunto de entrenamiento. La varianza estimada se reduce a medida que k es seleccionado más grande. Como desventaja, este método debe entrenar el clasificador k veces, lo cual eleva el tiempo de cálculo.

3.5 Agentes de software

Existen varias definiciones entre las cuales podemos mencionar [33]:

- **Coen:** los agentes de software son programas que se comprometen en diálogo, negocian y coordinan transferencia de información.
- **IBM:** los agentes de software inteligentes son entidades de software que realizan algún conjunto de operaciones en beneficio de un usuario o de otro programa con cierto grado de independencia o autonomía, y realizando esto, emplean cierto conocimiento o representación de los deseos y objetivos del usuario.

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

- **Hayes-Roth:** los agentes de software inteligentes realizan continuamente tres funciones:
 - Percepción de las condiciones dinámicas en el ambiente.
 - Acción para afectar las condiciones en el ambiente.
 - Razonamiento para interpretar las percepciones, resolver problemas, realizar inferencias, y determinar acciones.
- **Rusell y Norving:** los agentes de software son cualquier cosa que puede ser vista como algo que percibe su ambiente a través de sensores y actúa sobre su ambiente a través de efectores.
- **Maes:** los agentes de software autónomos son sistemas computacionales que habitan en algún ambiente dinámico y complejo, monitorean y actúan de forma autónoma en este ambiente, y realizando esto logran un conjunto de objetivos o tareas para los cuales están diseñados.
- **FIPA:** los agentes de software son una entidad que reside en entornos donde interpreta datos que reflejan eventos y ejecuta comandos que producen efectos en ese entorno.

Un agente de software (SA del inglés, *Software Agent*), es una entidad capaz de percibir su entorno, procesar tales percepciones y responder o actuar en su entorno de manera racional [71], es decir, de manera correcta y tendiendo a maximizar un resultado esperado [14]. Un agente inteligente puede ser una entidad física o virtual. Los agentes inteligentes se describen esquemáticamente como un sistema funcional abstracto.

3.5.1 Taxonomía según Nwana

1. Según su movilidad: Tenemos Agentes estáticos y Agentes Móviles
2. Deliberativos/reactivos:
 - Deliberativos: son agentes que poseen un modelo de razonamiento interno simbólico, y planifican y negocian para conseguir coordinación con otros agentes.

- Reactivos: los agentes no tiene ningún modelo interno simbólico de su entorno, y actúan de acuerdo a un tipo de comportamiento estímulo/-respuesta
3. Según los atributos los agentes deberían manifestar: autonomía, aprendizaje y cooperación (Véase la figura 3.3)

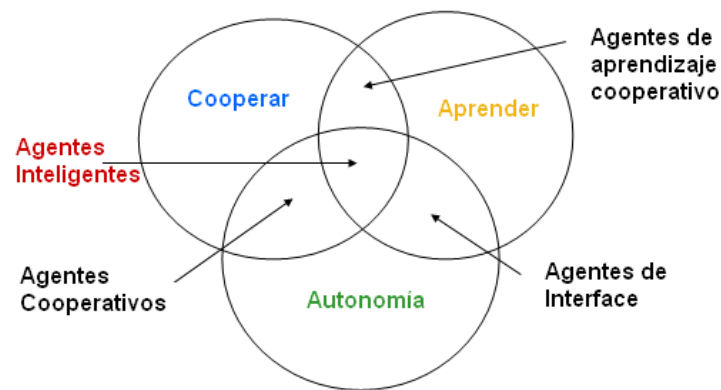


Figura 3.3: Atributos de agentes según [63]

De acuerdo a estos atributos podemos clasificarlos en las siguientes sub categorías:

- Agentes colaborativos
 - Agentes de aprendizaje colaborativos
 - Agentes de interface
 - Agentes inteligentes (“smart”)
4. Los agentes pueden clasificarse por sus roles: Por ejemplo : agentes de información o también llamados agentes de Internet.
5. Agentes híbridos: combinación de dos o más filosofías

3.5.2 Agente de Software para una Interfaz Cerebro Computador

Una aBCI debe permitir configurar y seleccionar las características específicas para la tarea a realizar por el usuario. El sistema busca inicialmente las características

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

que presentan la información más pertinente, haciendo uso de agentes de software basados en algoritmos genéticos y análisis estadístico de datos [53]. La decisión sobre el número de características y qué canales deben ser utilizados se obtiene de la selección del clasificador [43].

Durante el proceso de selección de características, diferentes estrategias de búsqueda son posibles. Sin embargo, el número de subconjuntos de características presentan un resultado combinatorio, y una búsqueda completa a través de todos los subconjuntos posibles suele no ser factible [70]. Muchos de los problemas relacionados con la selección de características son complejos, los métodos de búsqueda *forward* (a partir de una función y la construcción iterativa de conjuntos de características de mayor tamaño) y la eliminación del *feedback* (comenzando con todas las características, y forma iterativa, va eliminando rasgos) son la selección de características más comunes [78]. Un inconveniente importante de estos métodos más simples es que pueden estar presentes interacciones no lineales entre las características. En ese caso, el problema de cómo evaluar la relevancia de una característica no es trivial, ya que el rendimiento global podría presentar un crecimiento exponencial en el costo computacional, debido al número de características utilizadas. En la figura 3.4 se muestra el esquema propuesto utilizado en este trabajo. La aBCI está compuesta por una BCI tradicional, a la cual se han añadido agentes de software para la selección de características y del clasificador.

3.5.2.1 Agente de Software para selección de características

Esta propuesta implementa agentes de software utilizando algoritmos genéticos y una función de costos para la selección de características y selección del clasificador. Los agentes de software entregan una topología para la aBCI, similar a un modelo neuronal. En este modelo se relacionan los canales, las características y el clasificador seleccionado para el usuario.

- **Función de Costo:** la tasa de acierto es un índice muy intuitivo y fácil interpretar; este toma valores entre 0 (totalmente en desacuerdo) a 1 (totalmente de acuerdo). Sin embargo, si un clasificador utiliza dos clases, y el primero tiene una tasa de acierto del 90 % y el segundo 50 %, la tasa de acierto

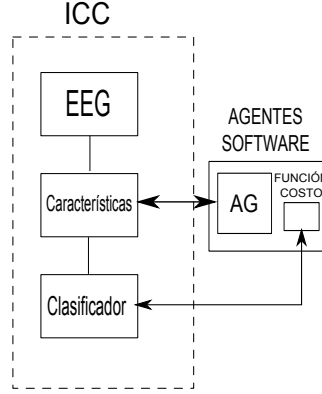


Figura 3.4: aBCI propuesta basada en agentes de software

promedio sería de 70 %. Este resultado no es tan bueno como un clasificador que tiene 70 % para ambas clases. Entonces, es deseable tener un índice que muestre el grado de concordancia que existe por encima de lo esperado debido al azar. En este sentido, el índice utilizado en este trabajo es el propuesto por Cohen [22], llamado coeficiente de Kappa (k), que se define en la ecuación 3.20.

$$Kappa = \frac{\sum_{i=1}^q p_{ii} - \sum_{i=1}^q p_{i-p-i}}{1 - \sum_{i=1}^q p_{i-p-i}} \quad (3.17)$$

donde $\sum_{i=1}^q p_{ii}$ es la tasa de acierto, y $\sum_{i=1}^q p_{i-p-i}$ es el porcentaje de error esperado debido al azar.

En la ecuación 3.18 la función de costo se define como:

$$cost_f = \frac{W_{h_1} \cdot Ofeat}{Nfeat} + \frac{W_{h_2} \cdot Kappa}{Cohen} + \frac{W_{h_3} \cdot Acc}{Acc_d}, \quad (3.18)$$

donde W_{h_1} es el peso para el número de características, $Ofeat$ es el número de características deseadas y $Nfeat$ es el número de elementos de cada característica para la selección. W_{h_2} es el peso de coeficiente Kappa y el coeficiente $Kappa$ es una medida de la concordancia de las etiquetas y predictor (clasificador); $cohen$ es el criterio de Cohen [22] para evaluar una buena concordancia ($Kappa > 0,61$). W_{h_3} es el peso para la tasa de acierto, Acc es la

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

tasa de acierto seleccionado para medir la exactitud del clasificador, y Acc_d es la exactitud requerida.

- **Selección de Características:** la selección de características es realizada para reducir el costo computacional en los sistemas BCIs. Las características que presenten la máxima tasa de acierto y el menor número de ellas, junto con el máximo valor de Kappa, son las seleccionadas. La función de costo permite ponderar, el número de características y el valor deseado de tasa de acierto. En la figura 3.5 se presenta un modelo de representación neuronal para una BCI, bajo el esquema de selección de características y clasificador.

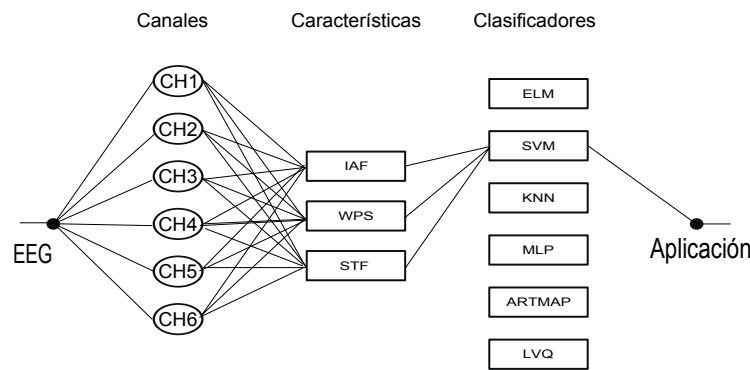


Figura 3.5: Representación en modelo neuronal para una aBCI.

- **Ajuste de Parámetros:** el ajuste de parámetros debe permitir modificar el sistema para presentar el mejor modelo para un caso particular de un sistema BCI.

Siendo la aBCI una extensión de las BCIs tradicionales estas contienen un nuevo bloque llamado SA, el cual modifica los parámetros para la extracción de características y el proceso de clasificación, mejorando el desempeño de la BCI. En el caso de la interacción del SA con la BCI para la modificación de los parámetros en el proceso de clasificación en la Figura 3.6 se muestra la estructura para esta interacción. Las partes asociadas a una aBCI son:

1. Pre-procesamiento

Este bloque busca incrementar la relación señal/ruido o decrementar

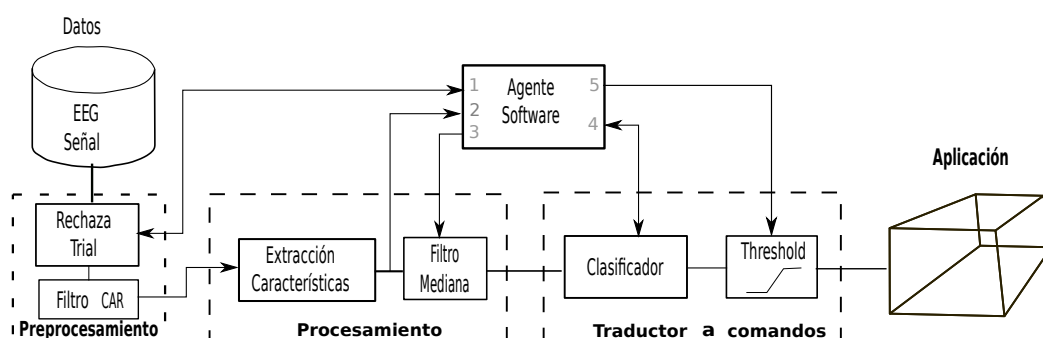


Figura 3.6: Diagrama esquemático de una aBCI con ajuste de parámetros usando SA.

los errores de clasificación por *trials* contaminados en el proceso de adquisición de señales, por dos diferentes rutas:

- **Rechazo automático de *trial*.** Cuenta el número de muestras contaminadas en cada *trial* para obtener el porcentaje de muestras con artefactos o valores atípicos (*outliers*). El porcentaje es usado para definir el umbral de la probabilidad *a posteriori* del clasificador y su aceptación o rechazo del *trial* analizado. el Algoritmo [2] muestra el mecanismo para el rechazo automático de *trial*.

Algoritmo 2. Rechazo automático de *trial*

```

1: function [PERCENTAGE, REJECTION] = TRIAL_RECHAZADO(EEG,criterion)
2:   EEG(channels, Samples, Trials)
3:   EEGx ← median(EEG) % obtiene la mediana de la señal EEG signal para todos los canales
4:   EEGx ← subtraction(EEG,EEGx) se le resta a EEGx & la mediana de la señal EEG
5:   EEGp ← EEGx > 0 % obtienen todos los valores positivos
6:   EEGn ← EEGx < 0 % obtienen todos los valores negativos
7:   EEGp ← median(EEGp) % media de EEGp
8:   EEGn ← median(EEGn) % media de EEGn
9:   Thresholdp ← (2+criterion)*EEGp % Umbral positivo
10:  Thresholdn ← (2+criterion)*EEGn % Umbral negativo
11:  countp ← number of samples over threshold positive
12:  countn ← number of samples under threshold negative
13:  count ← countp + countn % Conteo de countp & countn
14:  Percentage ← count/Samples;
15:  if Percentage > criterion then
16:    Rejection ← 1
17:  else
18:    Rejection ← 0
19:  return Percentage, Rejection

```

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

El término *criterion* corresponde al porcentaje apartir del cual rechazamos los *trials*. *Percentage* totaliza el porcentaje de muestras contaminadas y la variable *Rejection* almancena que *trials* estan contaminados o no.

- **Filtro Espacial.** Es usado para eliminar las señales comunes en las proximidades de los electrodos. El filtro CAR es aplicado para incrementar la relación señal/ruido.

2. Procesamiento

El bloque de procesamiento busca la reducción de la dimensionalidad para incrementar el desempeño del proceso de clasificación. Este está compuesto por los siguientes bloques funcionales:

- **Extracción de características.** Es un método para la representación de los datos. Los datos EEG que son almacenados durante un experimento contienen un gran número de muestras. La representación de los datos es obtenida por la concatenación en series de tiempo para todos los canales. Los datos son almacenados en un vector de alta dimensionalidad. Sin embargo, este tipo de representación es poco práctica dado el alto costo computacional. Por esta razón, es necesario encontrar un subconjunto de muestras para una mejor representacion de las señales EEG para una tarea específica.
- **Filtro Mediana.** La mediana es el valor que separa un conjunto de datos en dos partes iguales, una mitad superior e inferior. El filtro mediana recorre la señal, entrada por entrada, reemplazando cada entrada por la mediana de las entradas vecinas. El patrón de entradas vecinas (número de elementos) es llamado “ventana”, la cual se desliza, entrada por entrada, sobre la señal. El principal objetivo es obtener un vector de características suavizado sin *outliers* e incrementar el desempeño del sistema.

3. Traductor a comandos

- **Clasificador.** En este bloque, es contruido un mapa, donde las entradas (luego de la extracción de características) y cada clase (tareas mentales) son relacionadas. El clasificador permite determinar la clase para una entrada basado únicamente en sus características.
- **Umbral.** Este bloque es un comparador. El clasificador debe suministrar información de la clase y la probabilidad con la que obtuvo esta clasificación. Este comparador utiliza el valor umbral para determinar si un valor dado por el clasificador es enviado o no a la salida del sistema.

4. Agente de software

Un SA es una pieza de código para realizar una tarea específica para un usuario u otras aplicaciones. Los SAs son autónomos e interactúan con su entorno para poder aprender de él [63]. En la Figura 3.7 se muestra la estructura de un SA. El SA define el valor para rechazar los *trials* contaminados y define el umbral para el clasificador. Es decir, el SA mantiene el control del flujo de los datos, por ejemplo si la señal de entrada es rechazada el clasificador no genera señal de salida. El SA trabaja en los dos modos de operación, en la fase *off-line* y en el modo *on-line*.

- *Off-line.* En este modo, el SA envía el número de elementos para el filtro mediana y almacena los modelos de los clasificadores
- *On-line.* En este modo, el SA genera un vector aleatorio con el número de elementos para el filtro mediana. Cada *trial* calcula el filtro mediana y selecciona el clasificador para cada valor particular. Luego, el SA realiza una comparación entre la probabilidad obtenida por el clasificador y la probabilidad obtenida por el clasificador seleccionado en el modelo de clasificadores almacenados. Una matriz con los elementos para el filtro mediano es guardado. En el algoritmo [3] se presenta el SA en pseudo-código.

Algoritmo 3. Implementación del Agente de Software

```
1: function SOFTWARE_AGENT(mode)
2:   Vector=random(1:mdf) % Genera un vector con mdf elementos
3:   if mode = 0 then
```

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

```

4:   for i:=1 to mdf do
5:       [Percentage Rejection] = REJECTION_TRIAL(EEG,criterion)
6:       feat = feature_extraction(EEG(Rejection))
7:       feat_med=median_filter(feat,i)
8:       model(i) = Training(feat_med, Labels(Rejection))
9:       Save model
10:  else
11:      [Percentage Rejection] = REJECTION_TRIAL(EEG,criterion)
12:      if Rejection = 0 then
13:          Threshold=Percentage+1/(number classes) % number_classes =0.20 si son 5 clases
14:          feat = feature_extraction(EEG)
15:          feat_med=median_filter(feat,init)
16:          (ds1,Probab1)=Testing(model.BCI,feat_med1) % Probabilidad a posteriori del clasifica-
            dor de la BCI
17:          (ds1,Probab2)=Testing(model(vector(init)),feat_med2) % Probabilidad del Agente Soft-
            ware
18:          if Probab1 < Probab2 then
19:              matrix(mdf_BCI,init)=0 % Limpia la matriz de aprendizaje
20:              model.BCI=model(Vector(init)), mdf_BCI=Vector(init)
21:              ds1=ds2 % Salida para el clasificador
22:          else
23:              matrix(mdf_BCI,init)=1 % Aprendiendo
24:              if init < mdf then
25:                  init=init+1 %incrementa el valor init
26:              else
27:                  init=1, Vector=random(1:mdf)
28:      return Percentage, Rejection

```

Definiciones: *mode* =0 modo *off-line*; *mode*=1 modo *on-line*; *criterion* es el umbral para el rechazo de *trials*; *mdf* es el número de elementos para el filtro mediana; *model* es el modelo para los clasificadores; *Th* es el umbral de salida; *Perc* es el porcentaje de *trials* contaminados y *Rej* es la bandera para el rechazo de *trial*; *feat* es el vector de características y *Probab* es la probabilidad *a posteriori* del clasificador para las diferentes clases.

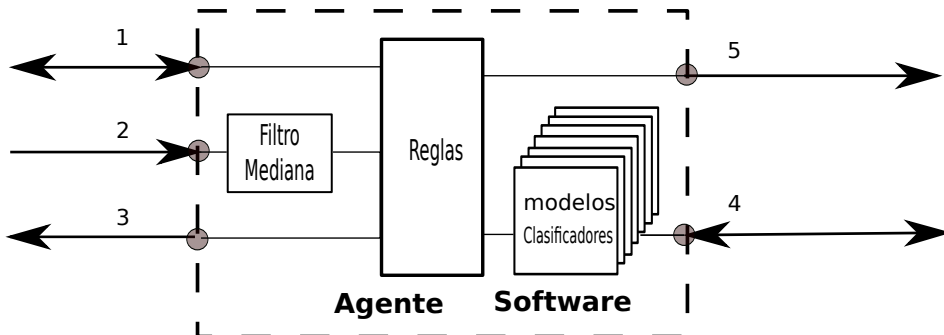


Figura 3.7: Diagrama esquemático de un SA

3.5.2.2 Agente de Software con umbral estático

Otro abordaje en la implementación de SA consiste en interactuar con el clasificador y modificar sus parámetros de salida. En el caso de utilizar un clasificador SVM o K-NN podemos tener un parámetro interesante relacionado con el número de elementos modelado por los vectores de soporte o en número de vecinos cercanos y obtener de este un valor en porcentaje de la probabilidad *a posteriori* del clasificador. En la figura 3.8 se presenta un esquema donde se define un umbral a partir del cual la salida del clasificador es validada, en caso contrario el sistema BCI presenta salida nula. La principal ventaja de este esquema es la posibilidad de incrementar la tasa de acierto del sistema y el costo se ve reflejado en la posible disminución de la tasa de transferencia de información, la cual puede afectarse en los casos en los que el sistema no clasifica.

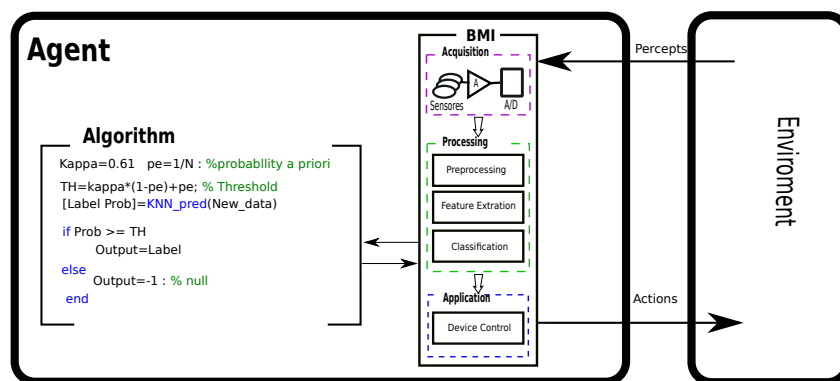


Figura 3.8: aBCI propuesta basada en agentes de software con umbral estático

3.5.2.3 Agente de Software con umbral dinámico

Una forma de dotar de inteligencia al SA consiste en inferir situaciones favorables o no provenientes de la interacción usuario-interfaz. Esta interacción puede ser modelada directamente de las características temporales de la señal. En la figura 3.9 se presenta un esquema de una aBCI basado en SA, donde se define un umbral dinámico el cual está relacionado con las desviaciones de la señal de entrada. Para medir estas desviaciones se parte de la premisa que la señal presenta unas características básicas las cuales presentan pequeñas desviaciones a lo largo del tiempo,

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

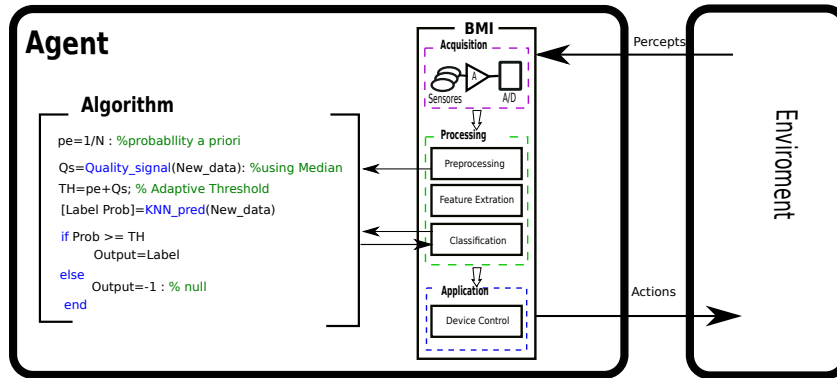


Figura 3.9: aBCI propuesta basada en agentes de software con umbral dinámico

es decir, no presentan cambios abruptos significativos. Para el cálculo de estas desviaciones hacemos uso de la mediana de la señal, luego consideramos que una señal en condiciones de normales se encuentra contenida en el doble del valor de la mediana y que los valores por fuera de este se contabilizan y se obtiene su valor porcentual. El umbral definido contempla un porcentaje asociado al número de clases del sistema sumado al porcentaje de señales por fuera de dos veces el valor de la mediana de la señal de entrada. Con este umbral el cual es dinámico, pues depende de las condiciones de la señal de entrada, se ha propuesto un esquema de una aBCI con umbral dinámico usando SA.

3.5.2.4 Agente de Software con aprendizaje activo

El funcionamiento de un modelo asociado al aprendizaje activo basado en agente software se divide en cuatro fases:

1. **Fase de inicialización:** En esta fase se inicializan los parámetros del sistema y se establecen las señales de control. Mediante las señales de control, conocidas como umbral *a priori*, exactitud aceptada y un respaldo de los datos de entrenamiento, se controla el flujo de datos a través del agente y se pasa a las otras fases.
2. **Fase de Exploración:** En la fase de exploración se efectúa una operación con los datos de entrada y los pesos asociados a la calidad de la señal de entrada o algún parámetro utilizado para inferir el proceso de interacción

entre el usuario y la BCI. Estos parámetros pueden ser el nivel de atención del usuario, el nivel de estrés o fatiga, el resultado de esta operación debe indicar qué clase tiene mayor prioridad para ver si los datos de entrada presentan un nivel mayor que el umbral definido. Por ejemplo, se podría calcular la distancia euclidiana entre los datos de entrada y los pesos, la clase ganadora sería aquella cuyo peso estuviese más cerca de los datos de entrada y por lo tanto sería la primera a la que se le intentaría asociar dicho patrón.

3. **Fase de explotación:** En esta fase el vector de entrada es evaluado en el clasificador y si la probabilidad *a posteriori* es mayor, se selecciona la salida de acuerdo a esta condición e incrementa el valor umbral facilitando la robustez del sistema. El objetivo es obtener una medida de similitud entre el modelo del clasificador de entrada y el clasificador prototipo si se incluye en nuevo dato.
4. **Fase de búsqueda:** De no representar el nuevo dato del vector de entrada, este establece si lo remueve de los datos de entrenamiento considerando el dato más alejado perteneciente a esa clase o elimina un dato aleatorio.

El aprendizaje se inicia sólo al final del procedimiento de búsqueda, en el cual se satisface con una coincidencia de reconocimiento. Si ninguna coincidencia ha sido encontrada, no hay actualización del modelo de entrenamiento. Existen dos métodos básicos para entrenar: lento y rápido. En el método lento el grado de entrenamiento de los pesos hacia el vector de entrada se calcula para valores continuos y por lo tanto depende del tiempo durante el cual el vector de entrada esté presente. Con el método rápido, se utilizan actualizaciones para calcular el grado de ajuste de peso. Si bien el aprendizaje rápido es eficaz y eficiente para ciertas tareas, el método de aprendizaje lento es biológicamente posible y puede usarse con agentes en tiempo continuo, es decir, cuando el vector de entrada varía de forma continua.

Una de las características de la memoria humana consiste en su habilidad para aprender nuevos conceptos sin necesitar para ello olvidar otros aprendidos en el

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

pasado. Sería deseable que esta misma capacidad se pudiera conseguir en los sistemas BCI. Sin embargo, muchos de estos sistemas tienden a olvidar informaciones pasadas al tratar de enseñarles otras nuevas o tener que entrenar el sistema nuevamente. El esquema propuesto elimina los elementos que se encuentren más alejados, actualizando el conjunto de datos de entrenamiento del sistema.

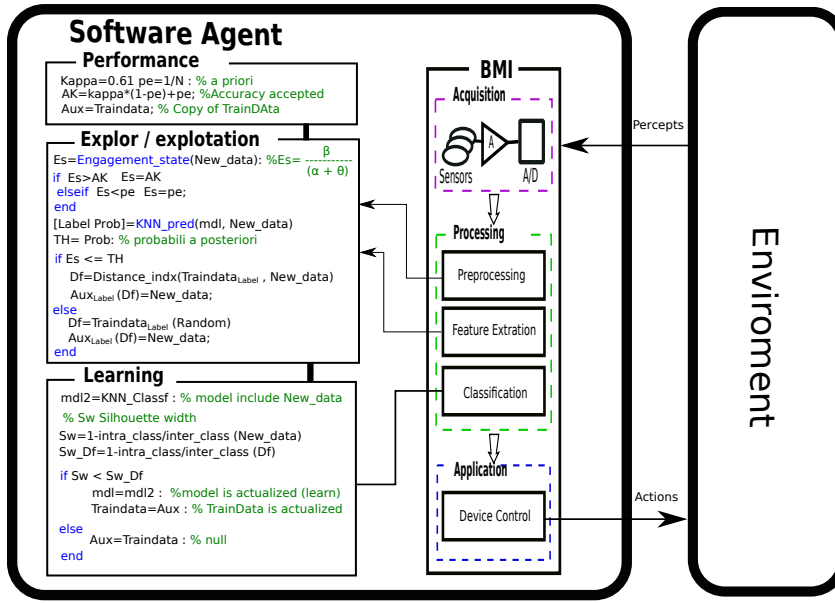


Figura 3.10: aBCI propuesta basada en agentes de software con aprendizaje activo

3.5.3 Medidas de Desempeño

Las medidas de desempeño para un sistema BCI se obtiene principalmente de la matriz de confusión [43], en la Figura 3.11 pueden observarse la matriz de confusión y las medidas de desempeño más usuales son:

- Exactitud. Esta relacionada a la tasa de acierto del clasificador.
- Tasa de transferencia Información. El **ITR** (del inglés, *Information Transfer Rate*) es una medida estándar usada en sistemas de comunicación, la cual cuantifica la tasa de transmisión de información por unidad de tiempo. El ITR depende de la velocidad y la tasa de acierto [58] y es definido por la Ecuacion 3.19.

$$B = (1 - P_u) * (\log_2 N + P \log_2 P + (1 - P) \log_2 (\frac{1-P}{N-1})) , \quad (3.19)$$

donde N es el número de clases, P es la tasa de aciertos de, y P_u es el tasa de clases no identificadas. El ITR es expresado en $[bits/s]$, pero puede ser expresado en $[bits/min]$ multiplicando el resultado por la velocidad seleccionada, por ejemplo, el número de selecciones desarrollada por el sistema puede ser en minutos. En el caso de tareas mentales se encuentra entre los rangos de 5 a 25 $[bits/min]$.

- Coeficiente de Kappa, es un parámetro que representa la concordancia entre el valor objetivo y el valor predicho. En este sentido, el índice aquí utilizado es el propuesto por Cohen [22]. El coeficiente Kappa esta definido en la Ecuación 3.20.

$$Kappa = \frac{\sum_{i=1}^q p_{ii} - \sum_{i=1}^q p_{i-p-i}}{1 - \sum_{i=1}^q p_{i-p-i}}, \quad (3.20)$$

donde $\sum_{i=1}^q p_{ii}$ es la exactitud, y $\sum_{i=1}^q p_{i-p-i}$ es el porcentaje debido al azar. $Kappa > 0,61$ indica un buen nivel de concordancia [22]. Luego, para cinco clases la exactitud esperada debería ser $> 69\%$.

- Sensitividad y especificidad son medidas que proveen información acerca de la habilidad para detectar una clase en particular (condición verdadera positiva o negativa). La sensitividad y la especificidad son calculadas de acuerdo a las Ecuaciones 3.21 y 3.22 respectivamente.

$$Sensitivity = \frac{TP}{TP + FN}, \quad (3.21)$$

$$Specificity = \frac{TN}{FP + TN}, \quad (3.22)$$

donde TP se refiere a verdadero positivo; FN corresponde a falso negativo; TN es verdadero negativo y FP es falso positivo. La sensitividad y la especificidad son usadas para problemas de dos clases; en este caso el valor esperado debería estar alrededor del 81 %.

3. MÉTODOS PARA LA IMPLEMENTACIÓN DE UNA BCI

		Condition (as determined by "Gold standard")			
m=total population		Condition positive	Condition negative		
Test outcome	Test outcome positive	True positive	False positive (Type I error)	$f_p=TP+FP$	Kappa= $\frac{\text{Accuracy}-Pe}{1-Pe}$ Pe, is chance agreements
	Test outcome negative	False negative (Type II error)	True negative	$f_n=FN+TN$	
		$Y_p=TP+FN$	$Y_n=FP+TN$	Accuracy (ACC)= $\frac{\sum \text{True positive} + \sum \text{True negative}}{\text{Total population}}$	
		True positive rate (TPR, Sensitivity, Recall)= $\frac{\sum \text{True Positive}}{\sum \text{Condition positive}}$	True negative rate (TNR, Specificity, SPC)= $\frac{\sum \text{True negative}}{\sum \text{Condition negative}}$		

True positive (TP)
eqv. with hit

true negative (TN)
eqv. with correct rejection

false positive (FP)
eqv. with **false alarm**, **type I error**

false negative (FN)
eqv. with miss **Type II error**

chance agreements (pe)
($Y_p \cdot f_p + Y_n \cdot f_n$)/(P+N)²

sensitivity or **true positive rate (TPR)**
eqv. with hit rate recall

$TPR=TP/P=TP/(TP+FN)$

specificity (SPC) or True Negative Rate
 $SPC=TN/N=TN/(FP+TN)$

accuracy (ACC)
 $ACC=(TP+TN)/(P+N)$

kappa coefficient (kappa)
 $kappa=(ACC-Pe)/(1-Pe)$

$$ITR = \log_2 N + ACC \log_2 ACC + (1-ACC) \log_2 \left(\frac{1-ACC}{N-1} \right)$$

N number of classes

True positive (TP)
eqv. with hit

true negative (TN)
eqv. with correct rejection

false positive (FP)
eqv. with false alarm, type I error

false negative (FN)
eqv. with miss Type II error

chance agreements (pe)
 $(Y_p f_p + Y_n f_n) / (P + N)^2$

sensitivity or true positive rate (TPR)
eqv. with hit rate recall

$TPR = TP / P = TP / (TP + FN)$

specificity (SPC) or True Negative Rate
 $SPC = TN / N = TN / (FP + TN)$

accuracy (ACC)
 $ACC = (TP + TN) / (P + N)$

kappa coefficient (kappa)
 $kappa = (ACC - P_e) / (1 - P_e)$

Figura 3.11: Matriz de Confusión y medidas de desempeño para una BCI

Cada día sabemos más y entendemos menos.

Albert Einstein

CAPÍTULO

4

Pruebas, análisis de datos y resultados

4.1 Plataforma de experimentación para una BCI para tareas mentales

La plataforma de experimentación cuenta con tres modos de operacion:

1. Modo de Captura En este modo se presentan las diferentes tareas mentales con los diferentes protocolos para el proceso de adquisición de las señales.
2. Modo de Análisis Este modo permite realizar el proceso de caracterización de las diferentes tareas mentales, definiendo así las que presenten características bien diferenciables.
3. Modo de Entrenamiento En este modo se seleccionan las diferentes combinaciones en el proceso de eliminación de artefactos, extracción de características, y finalmente la selección del clasificador.

En la figura 4.1 se puede observar la interfaz para la experimentación.

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

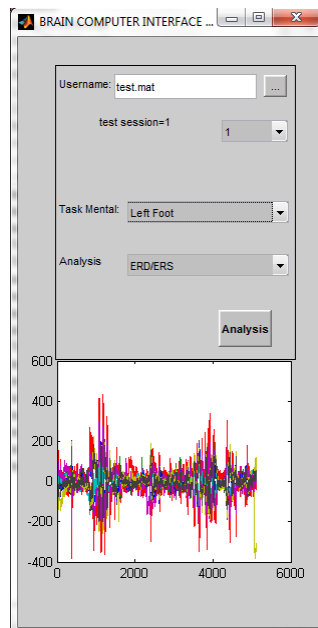


Figura 4.1: Plataforma de experimentación

4.1.1 Análisis ERD/ERS

En este análisis se tiene en cuenta el canal para observaciones detalladas de frecuencia, las características temporales y el análisis topográfico para las siguientes características:

- Banda alfa baja 8–10 [Hz]
- Banda alfa alta 10–12 [Hz]
- Banda alfa 8–12 [Hz]
- Banda beta 13–30 [Hz]
- Valor medio absoluto [V]
- Valor medio cuadrático [V]
- ERD 8–11 [Hz] valor porcentual
- ERS 26–30 [Hz] valor porcentual

En la figura 4.2 se puede observar un tipo de análisis del ERD/ERS.

4.1 Plataforma de experimentación para una BCI para tareas mentales

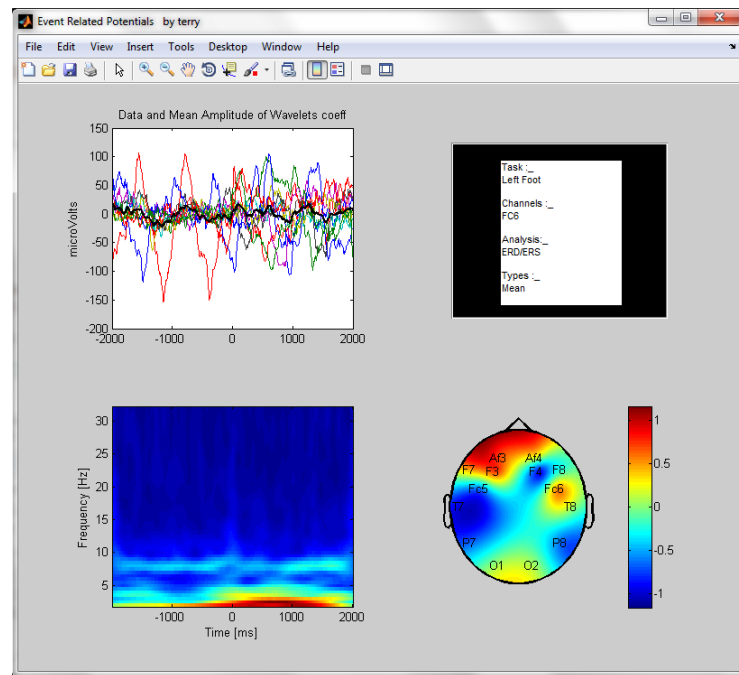


Figura 4.2: Modo de análisis para ERD/ERS, representación topográfica

4.1.2 Análisis de la nube de datos complejos para señales EEG

El denominado análisis de la nube de datos complejos, permite analizar en diferentes instantes de tiempo y para diferentes frecuencias, el valor medio y la varianza antes y después del estímulo, para así obtener algún tipo de información que pueda facilitar el análisis de los datos. En la Figura 4.3 se puede observar un ejemplo de como se observan los datos para este análisis.

4.1.3 Análisis Bandas de Potencia

El análisis de las bandas de potencia permite observar la densidad espectral de potencia en todos los electrodos y también permite visualizar el efecto de ERD/ERS para cada tarea específica.

En la figura 4.4 en la parte a) se puede observar el espectro de potencia para cada canal y en la parte b) la sincronización/desincronización relacionada a

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

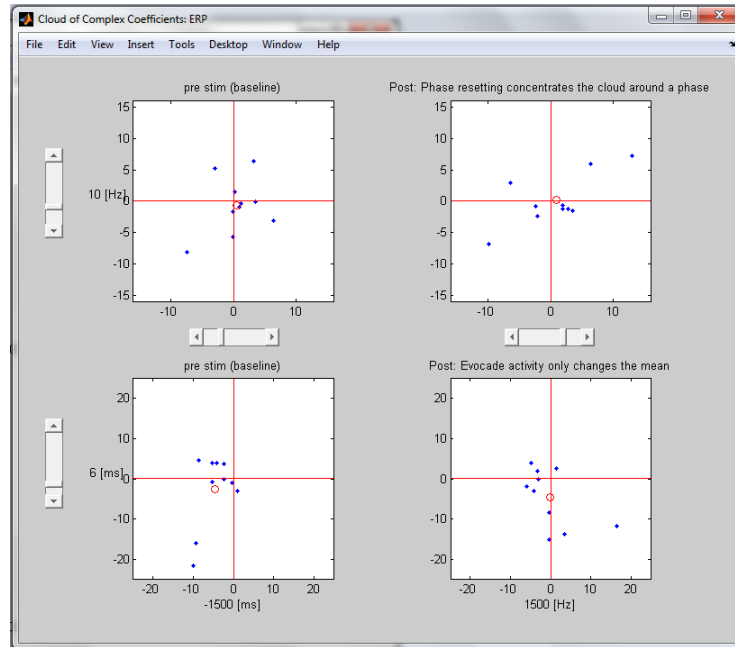


Figura 4.3: Modo de análisis de la nube de datos complejos para un canal

eventos.

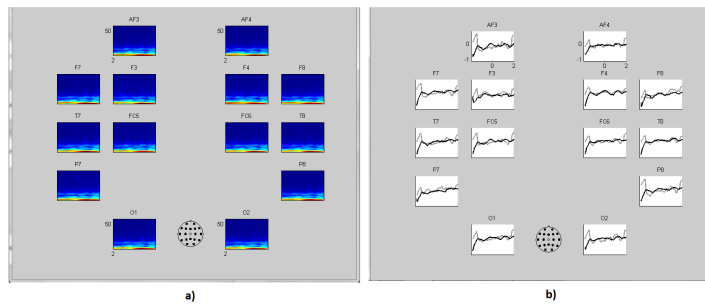


Figura 4.4: Modo de análisis de la densidad espectral de potencia y ERD/ERS en todos los electrodos

4.1.4 Análisis de la estadística compleja para señales EEG

En el estudio de señales EEG existe un análisis de estadístico complejo, ya que estas señales son ricas en información tanto temporal, espacial y frecuen-

4.1 Plataforma de experimentación para una BCI para tareas mentales

cial, para tener un estudio más completo y seleccionar las mejores características para identificar las diferentes tareas mentales se sugiere el uso de esta herramienta, la cual es presentada en la figura 4.5. A continuación se presenta una tabla con las medidas estadísticas que se utilizarán para el análisis temporal-frecuencial de las señales EEG:

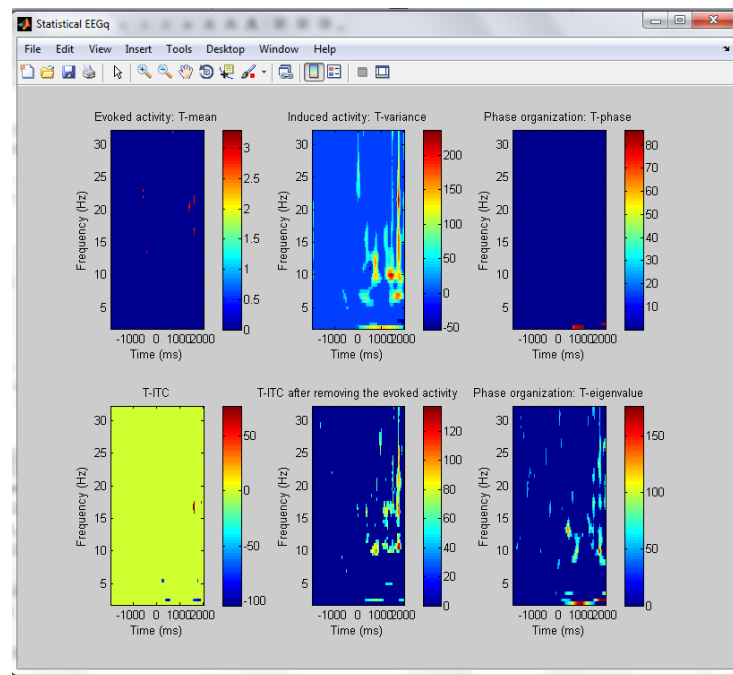


Figura 4.5: Análisis de la cuantitativo de la señal EEG para un canal)

- T-ITC suministra información de la coherencia de fase inter trial o el factor de fase bloqueada o enganchada.
- T-mean Esta relacionado con el valor medio estadístico para señales complejas y los potenciales evocados.
- T-Variance está relacionado con la varianza para señales complejas y actividad oscilatoria inducida.
- T- eigenvalue Prueba la reorganización de la fase usando la prueba de similaridad de los valores propios de la matriz de covarianza entre la parte real e imaginaria de los coeficientes complejos. Puede ser vista

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

como la medida generalizada de la correlación entre la parte real e imaginaria.

- T- phase Prueba para la reorganización de la fase usando el segundo momento trigonométrico, el cual mide la uniformidad de la fase en la distribución bimodal que aparece después de remover el vector por un reinicio de la fase.

4.1.5 Análisis de la coherencia

En la Figura 4.6 se puede observar la relación de coherencia en las diferentes bandas de frecuencia.

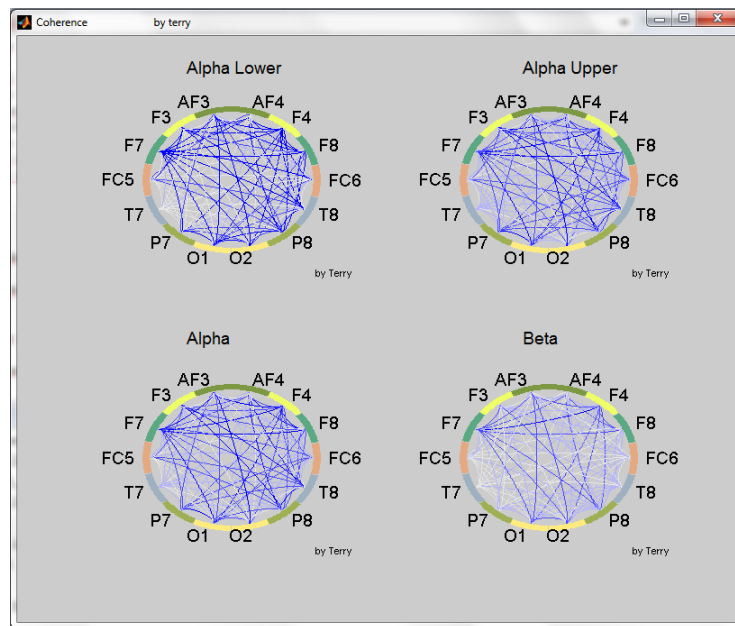


Figura 4.6: Análisis de la coherencia en las banda alfa baja 8–10[Hz], alfa alta 10–12 [Hz], alfa 8–13 [Hz], beta 13–30 [Hz] y las interconexiones de todos los canales

4.1.6 Análisis del factor de fase bloqueada

En la Figura 4.7 se puede observar el factor de fase bloqueada antes de la tarea mental y después de ésta.

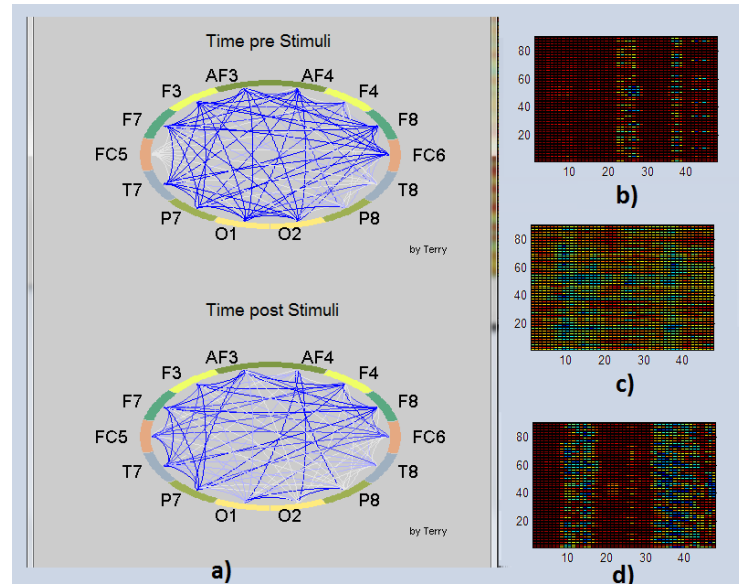


Figura 4.7: Análisis del factor de fase de bloqueo antes y despues de la tarea mental en todas la combinaciones de los canales

4.2 Plataforma de Simulación

En esta plataforma se evalua el esquema de entrenamiento propuesto para una aBCI, se evalua también diferentes base de datos de tares mentales, las cuales son llevadas al esquema de aBCI basada en SA. Son evaluados los esquemas de selección de características, ajuste de parámetros y los agentes presentados en la sesión 3.5.2.

4.2.1 Análisis de base de datos para cinco tareas mentales

La base de datos utilizada fue obtenida en [46]. En este estudio participaron siete sujetos con edades que oscilaban entre 21 y 48 años. Los electrodos utilizados estaban situados en las posiciones O1, O2, P3, P4, C3 y C4 del sistema 10 – 20. Las señales fueron filtradas analógicamente en 0,1 – 100 Hz, muestreadas a 250 Hz y grabadas por 10 s para cada tarea mental. Se realizaron 5 tareas mentales distintas, las cuales son:

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

Tabla 4.1: Resultados de la tasa de aciertos, el coeficiente Kappa, y el mejor clasificador para cada usuario.

Usuario	Característica	Acc.[%]	Kappa	Clasificadores
1	STF	0.75	0.67	ARTMAP
2	[IAF WPS STF]2	0.95	0.95	ARTMAP
3	STF	0.50	0.35	ELM
4	WPS	0.80	0.75	SVM
5	WPS	0.80	0.49	SVM
6	IAF	0.80	0.74	SVM
7	STF	0.80	0.75	SVM

- Línea de Base: No se realiza ninguna tarea mental, el sujeto se relaja y no piensa nada en específico.
- Multiplicación Matemática: El sujeto intenta resolver una multiplicación de varios dígitos.
- Rotación de una Figura: Se debe imaginar la rotación de una figura geométrica.
- Composición de una Carta: Mentalmente se realiza la composición de una carta dirigida a un ser querido.
- Cuenta visual: Usando imaginación visual se realiza el conteo en una pizarra.

En este estudio, todos los datos se dividieron en 5 partes, y una de ellas se tomó como conjunto de validación. Los datos restantes fueron utilizados como conjunto de entrenamiento. Este proceso se repitió 50 veces para la validación del sistema. Cada característica y un conjunto de todas las características fueron utilizadas para entrenar la BCI. En la Figura 4.8 se muestra el desempeño de cada uno de los usuarios. Para la evaluación de desempeño fueron calculados el coeficiente de Kappa y la tasa de acierto para cada clasificador basado en la validación cruzada. En la tabla 4.1 se presentan los valores de tasas de acierto y coeficientes de Kappa para cada usuario. Fue calculado el valor promedio de cada usuario y seleccionado el mejor clasificador. Los resultados presentados en dicho cuadro corresponde al resultado escogido por la aBCI.

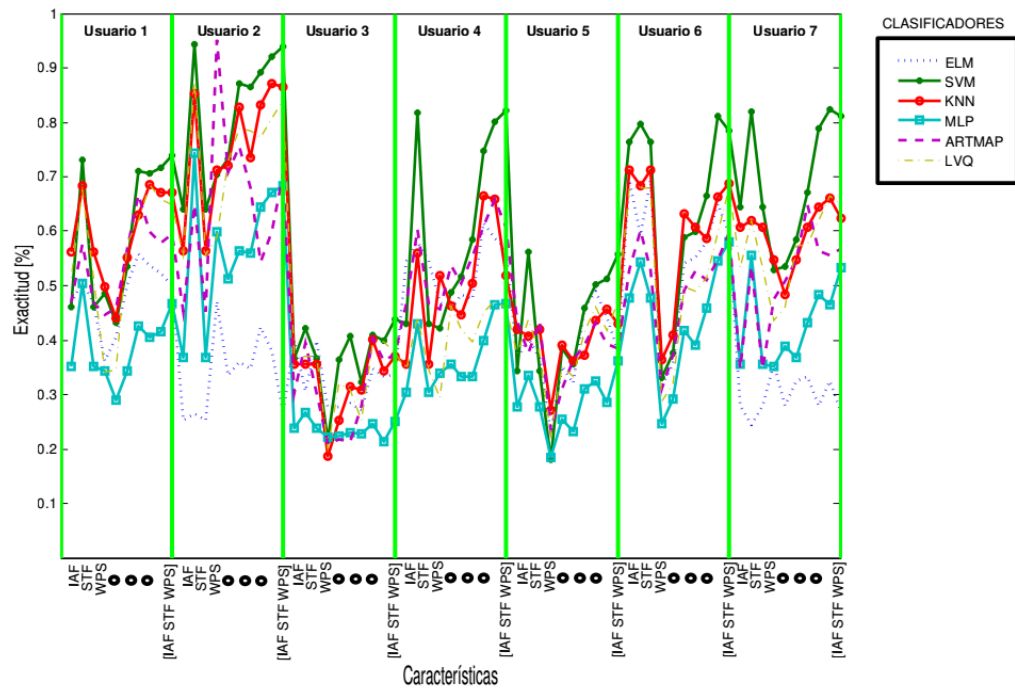


Figura 4.8: Desempeño de los métodos de extracción de característica, la tasa de acierto de los clasificadores para cada usuario.

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

Tabla 4.2: Resultados para todos los métodos de extracción de características.

Característica	Promedio		
	Acc.[%]	Kappa	#Caract.
IAF	0.56	0.44	12
WPS	0.72	0.66	186
STF	0.65	0.56	12
IAF WPS STF	0.54	0.42	2
IAF WPS STF	0.44	0.35	4
IAF WPS STF	0.47	0.37	8
IAF WPS STF	0.50	0.40	16
IAF WPS STF	0.61	0.55	32
IAF WPS STF	0.61	0.55	64
IAF WPS STF	0.35	0.31	128
IAF WPS STF	0.74	0.67	210
ADAPT. BCI	0.74	0.67	60

En la tabla 4.1 se puede observar que los usuarios 3 y 5 presentan los peores desempeños con relación a la tasa de aciertos y al coeficiente Kappa. El usuario 2 presentó el mejor desempeño con relación a la tasa de aciertos y el coeficiente Kappa. Adicionalmente, presenta el menor número de canales a utilizar con relación a los otros usuarios. El clasificador de máquina de vectores de soporte fue el mejor para los usuarios 4, 5, 6 y 7.

En la tabla 4.2 se presenta el valor promedio para todos los métodos de extracción de características implementados. Se puede observar que el desempeño de la aBCI es igual en términos de la tasa de acierto y valor del coeficiente de Kappa, que el vector que contiene todas las características [IAF WPS SFT]. La aBCI utiliza solo 60 elementos de los 210 que componen el vector de característica completo.

En la Figura 4.9 es mostrada la representación en el modelo neuronal de una aBCI para el usuario 2. Este modelo de representación neuronal nos permite relacionar los canales, las características y el clasificador óptimo para este usuario. Cabe resaltar que para este usuario fueron necesarios solo dos canales, obteniendo los mejores desempeños.

El sistema de la aBCI obtuvo una tasa de acierto de 74 %, presentando un valor mínimo de 50 % y 95 % como valor máximo. Usando descomposición empírica en modos, son alcanzados resultados promedios en la clasificación entre 50 % y 79 % [25]. Existen otros trabajos basados en la misma base de

4.2 Plataforma de Simulación

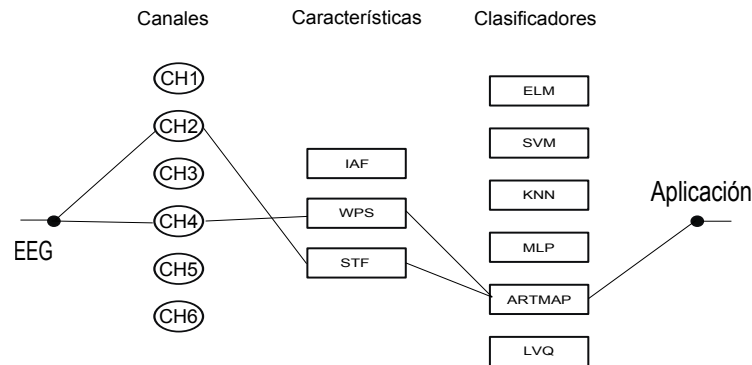


Figura 4.9: Representación en modelo neuronal para una aBCI para el usuario 2.

Tabla 4.3: Desempeño de una aBCI. Medidas de desempeño de la aBCI y la BCI tradicional para todos los usuarios, los valores entre parentesis “()” corresponden a la BCI tradicional.

Usuarios	Sensitividad	Especificidad	Exactitud 2 [%]	kappa_2	ITR [bits/min]	Exactitud 5 [%]	kappa_5
1	0,90 (0,73)	0,98 (0,93)	0,96 (0,89)	0,86(0,66)	6,09 (10.04)	0,90 (0,73)	0,86(0,66)
2	1,00 (0,95)	1,00 (0,99)	1,00 (0,98)	1,00 (0,93)	11,49 (12.94)	1,00 (0,95)	1,00 (0,93)
3	0,58 (0,39)	0,90 (0,85)	0,84 (0,76)	0,46 (0,24)	4,33 (7.63)	0,60 (0,39)	0,46 (0,24)
4	0,94 (0,84)	0,98 (0,96)	0,97 (0,94)	0,91 (0,81)	5,13 (11.32)	0,93 (0,84)	0,91 (0,81)
5	0,67 (0,55)	0,92 (0,89)	0,87 (0,82)	0,58 (0,44)	5,65 (8.52)	0,68 (0,55)	0,58 (0,44)
6	0,92 (0,77)	0,98 (0,94)	0,97 (0,91)	0,91 (0,71)	6,36 (10.48)	0,93 (0,77)	0,91 (0,71)
7	1,00 (0,83)	1,00 (0,96)	1,00 (0,93)	1,00 (0,79)	4,81 (11.19)	1,00 (0,83)	1,00 (0,79)
Promedio	0,86 (0,72)	0,97 (0,93)	0,94 (0,89)	0,82 (0,65)	6,27 (10.30)	0,86 (0,72)	0,82 (0,65)
p value	0,0078	0,0078	0,0078	0,0078	0,0078	0,0078	0,0078
d effect	0.82	0.87	0.83	0.78	3.92	0.86	0.78

datos para la clasificación de tareas mentales [64], [2], [32], y en ellos se muestran resultados promedios de 86 %, 71 % y 72 %, respectivamente. En estos trabajos no son considerados todos los usuarios de la base de datos.

Para evaluar la aBCI basada en SA para la modificación de parámetros utilizamos esta misma base de datos, con la diferencia que el método de extracción de característica usado fue WPS basado en los coeficientes Wavelets. El clasificador utilizado fue SVM, los resultados obtenidos son presentados en la tabla 4.3. Los resultados son contrastados entre la BCI tradicional y la propuesta de aBCI basada en SA con ajuste de parámetros.

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

En la Figura 4.10 se muestra el desempeño para todos los usuarios. Esta presenta la tasa de aciertos y el ITR. La línea intensa corresponde al filtro mediana cuando se configura con un elemento como parámetro (el filtro mediana esta desactivado). La línea punteada corresponde al mejor desempeño para un número específico de elementos en el filtro mediana. El usuario 1 presenta el 73 % de tasa de acierto para una BCI tradicional cuando es usado el filtro mediana ajustado para 13 elementos. Cuando el valor de la probabilidad es alto el valor es seleccionado, el ITR decrece para todos los usuarios. Los usuarios 2 y 7 presentan los mejores desempeños.

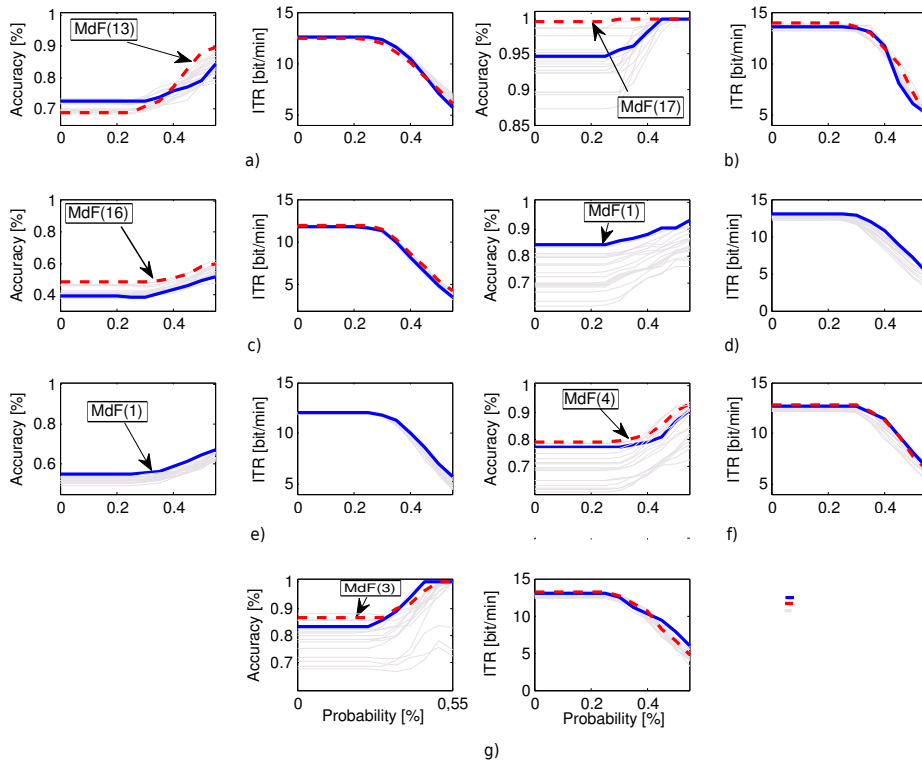


Figura 4.10: Tasa de aciertos e ITR para todos los usuario. Desempeño para una aBCI para todos los usuarios. a) Exactitud (Izquierda) e ITR (Derecha) para el usuario 1; b) Exactitud e ITR para el usuario 2; c) Exactitud e ITR para el usuario 3; d) Exactitud e ITR para el usuario 4; e) Exactitud e ITR para el usuario 5; f) Exactitud e ITR para el usuario 6; y g) Exactitud e ITR para el usuario 7.

La Figura 4.11 muestra el desempeño para todos los usuarios en cada tarea mental. La tarea con menor desempeño es la composición de una carta, y la tareas con el mayor desempeño fueron línea base y rotación. Los usuarios 3 y 5 tuvieron el desempeño más bajo.

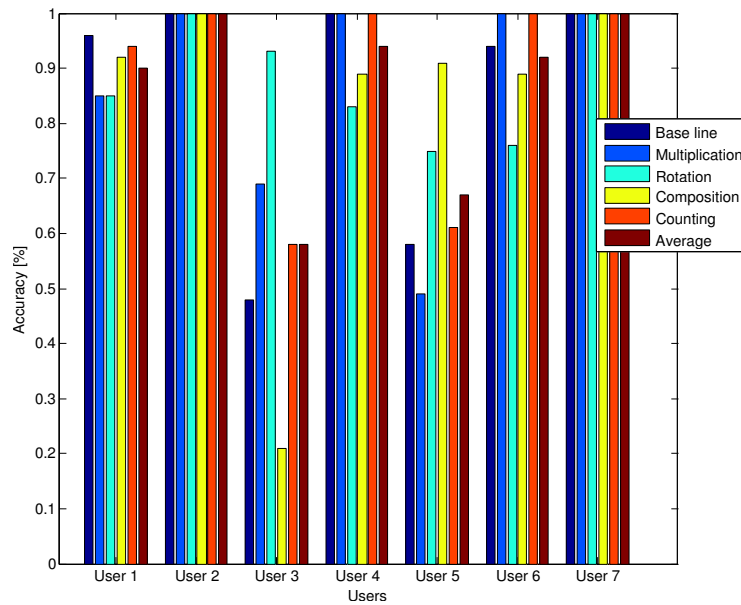


Figura 4.11: Desempeño de todas las tareas mentales para cada usuario

Fue realizado un análisis estadístico para validar el sistema propuesto. La prueba *Wilcoxon Signed-Ranks* (WSR) es una prueba alternativa no paramétrica para una muestra pareada cuando no se conoce si la distribución es normal. La prueba WSR es usada para evaluar la diferencia entre las medias de muestras dependientes (por ejemplo, BCI tradicional y aBCI) [34]. El tamaño del efecto usado por Cohen's permite evaluar que tan diferente son las medias de las muestras. Los valores del tamaño del efecto incluye valores como: pequeño ($d\text{-effect}=0.2$), medio ($d\text{-effect}=0.5$), grande ($d\text{-effect}=0.8$), muy grande ($d\text{-effect}=1.2$), y demasiado grande ($d\text{-effect}=2.0$) [75]. La tabla 4.3 muestra la medida para la evaluación del desempeño de la aBCI y la BCI tradicional. Esta tabla muestra que el usuario 3 presenta una tasa de sensibilidad (0.58) en el sistema propuesto y (0.39) en la BCI tradicional.

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

Tabla 4.4: Rechazo de Trials. Calidad de los datos usando el rechazo de muestras para varios valores.

Usuarios	Criterio de Rechazo							
	10[%]		15[%]		20[%]		25[%]	
	Trials Rechazados	Canales	Trials Rechazados	Canales	Trials Rechazados	Canales	Trials Rechazados	Canales
1	1.00	O1 O2 P3 P4 C3 C4	1.00	O1 O2 P3 P4 C3 C4	0.78	O1 O2 P3 C3 C4	0.18	O1 O2 C3 C4
2	1.00	O1 O2 P3 P4 C3 C4	1.00	O1 O2 P3 P4 C3 C4	0.24	O2 P4 C3 C4	0.00	O1 O2 P3 P4 C3 C4
3	1.00	O1 O2 P3 P4 C3 C4	1.00	O1 O2 P3 P4 C3 C4	0.58	O1 O2 P4 C3 C4	0.08	O1 O2 P4 C3 C4
4	0.98	O1 O2 P3 P4 C3 C4	0.98	O1 O2 P3 P4 C3 C4	0.3	O1 O2 P3 P4 C3 C4	0.02	O2 C3 C4
5	1.00	O1 O2 P3 P4 C3 C4	1.00	O1 O2 P3 P4 C3 C4	0.88	O1 O2 P3 P4 C3 C4	0.52	O1 O2 C3 C4
6	1.00	O1 O2 P3 P4 C3 C4	1.00	O1 O2 P3 P4 C3 C4	0.92	O2 P4 C3 C4	0.18	O2 P4 C3 C4
7	1.00	O1 O2 P3 P4 C3 C4	1.00	O1 O2 P3 P4 C3 C4	0.44	O1 P3 P4 C4	0.08	P4 C4
Promedio	1.00	O1 O2 P3 P4 C3 C4	1.00	O1 O2 P3 P4 C3 C4	0.59	O1 O2 P3 P4 C3 C4	0.15	O1 O2 P4 C3 C4

La tasa de acierto para dos clases (Accuracy₂) es tan alta como la tasa de acierto para cuatro clases, lo cual permitiría ser usada para una BCI de dos estados o para una BCI híbrida. La tasa de acierto y el coeficiente de Kappa para cinco tareas mentales presenta resultados aceptables exceptuando a los usuarios 3 y 5. La significancia estadística (p -value) es <0.05 para todas las medidas de desempeño. La hipótesis nula fue rechazada para todos los casos, teniendo en cuenta el gran valor del tamaño del efecto (d -effect el cual es >0.70). El ITR presenta un alto valor del tamaño del efecto, pero su valor decrece en la aBCI.

El usuario 3 y 5 presentaron el peor desempeño. En este aspecto, cabe comentar que el sistema propuesto es útil para un usuario en particular, y esto puede ser verificada cuando se realiza un análisis acerca de la ausencia / presencia de valores atípicos en las señales EEG. En este sentido, se empleó un método automático para el rechazo de trials contaminados.

En la Tabla 4.4 se muestra los resultados para evaluar el porcentaje de señal con artefactos o valores atípicos. En esta tabla (con el 20 % de la señal contaminada), sólo la mitad de los trials se deben utilizar para el entrenamiento y el proceso de validación. Por otra parte, cuando se utiliza 25 % de la señal contaminada, sólo el 15 % de los trials debe ser rechazado. Utilizando el

25 % como criterio de rechazo, el usuario 5 deberán presentar el mayor porcentaje (52 %) rechazó. Esto nos da una luz sobre lo que produce un bajo rendimiento en el procesamiento de BCI. Los usuarios 1,2,4,6 y 7 presentan buenos resultados. No obstante, el usuario 3 tuvo un bajo porcentaje de rechazo, sin embargo, su rendimiento es bajo. Podría ser debido al hecho de que este usuario no entendía la tarea mental o el sistema no es capaz de modelar sus señales EEG. Vale la pena comentar que el rechazo de trial sólo recopila información acerca de la calidad de la señal, pero no proporciona ninguna información sobre el contenido de la misma.

4.2.2 Análisis de base de datos para cuatro tareas mentales

Los datos utilizados para este análisis fueron registrados con un amplificador de señales EEG g.USBamp (g.tec medical engineering, Austria). Este amplificador tiene 16 canales y las señales fueron muestreadas a una frecuencia de 256 Hz. Los electrodos usados son g.LADYbird. El protocolo experimental permite distinguir entre cuatro estados mentales. Los participantes fueron 4 usuarios de $26,74 \pm 2,63$ años y sin experiencia en el uso de BCI. Las tareas mentales seleccionadas fueron dos relacionadas a imaginación motora y las otras dos a tareas cognitivas, una relacionada a la recitación mental del alfabeto de atras hacia adelante y la otra tarea consistió en realizar el conteo regresivo de 20 a 0. Los electrodos estaban localizados en la región fronto-parietal (región motora) y el área de Wernicke. Los electrodos estaban localizados en las siguientes posiciones (de acuerdo al sistema internacional 10/20): Fz, FC5, FC1, FCz, FC2, FC6, C3, Cz, C4, CP5, CP1, CP2, CP6, P3, Pz Y P4. El sistema de referencia fue monopolar localizado en el lóbulo de la oreja derecha y el sensor de referencia estaba ubicado en la posición AFz.

Un aspecto importante es el proceso de selección de características y un método que puede permitir reducir el costo computacional y el tiempo dedicado a este fin, es la selección de características basado en la anchura de Silhouette (SW), como fue descrito anteriormente, esta relaciona la distancia intra trial (cohesión) y la distancia inter trial (separación) entre un trial y la distribución de los demás en el cluster. En la Figura 4.12 donde para los

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

distintos métodos de extracción de característica son presentados sus distribuciones y las correspondientes SWs.

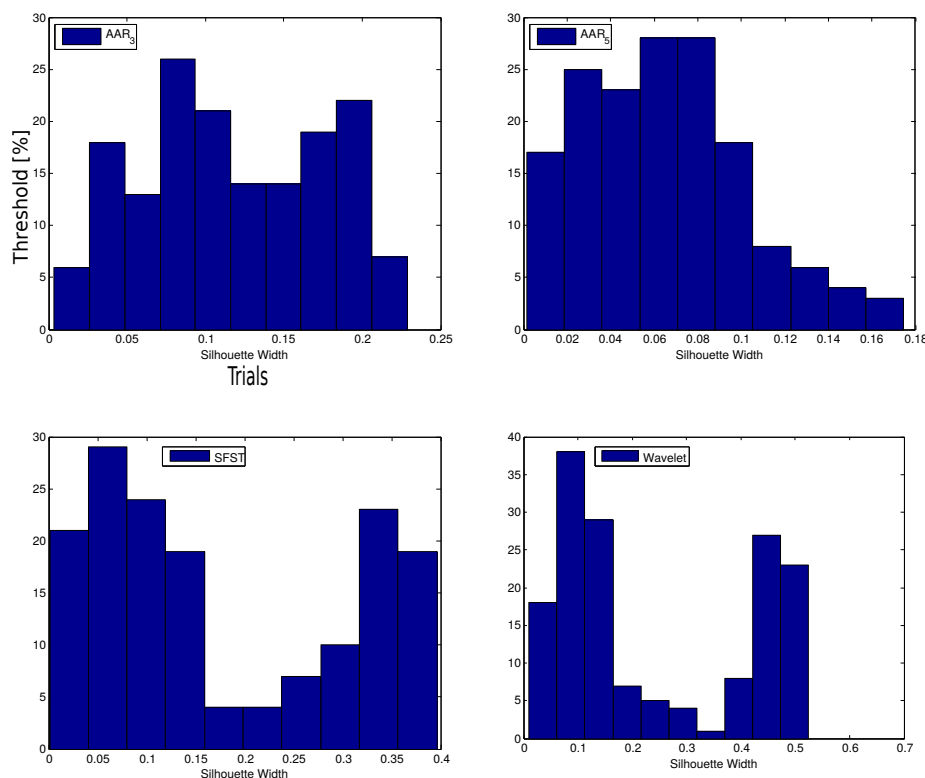


Figura 4.12: Distribución de las características usando SW para el usuario 1.

Del usuario 1 puede observarse que todos los métodos presentan valores de SW muy por debajo de 0.5 y que el método SFTS fue el que presentó una distribución con mayor SW.

En la Figura 4.13 es presentada la distribución para el usuario 2. En esta figura puede observarse que el método con mejor distribución es el SFTS. En la Figura 4.14 puede observarse que el método que presenta la mejor distribución es el WPS basado en Wavelets, cabe anotar que este usuario presenta valores SW mayores a 0.5, lo que puede indicar que en terminos de su tasa de acierto esta puede ser aceptable.

En la Figura 4.15 la distribución para el método SFTS presenta el mayor valor promedio su valor es superior a 0.5, indicando que existe una buena

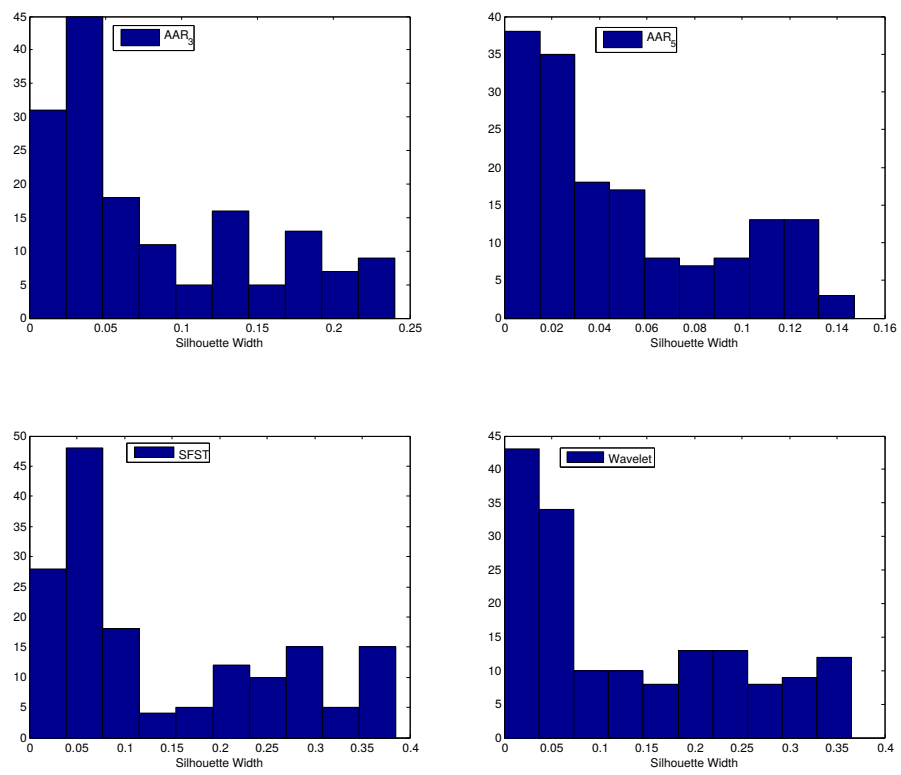


Figura 4.13: Distribución de las características usando SW para el usuario 2.

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

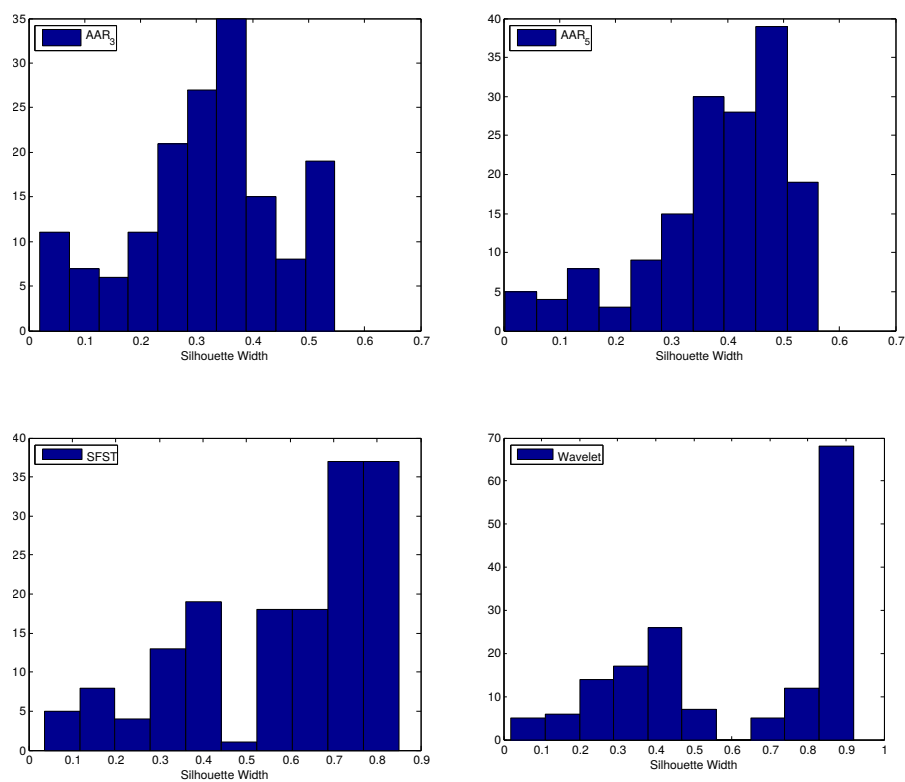


Figura 4.14: Distribución de las características usando SW para el usuario 3.

cohesión y separación en las características, lo que puede traducirse en buenos resultados a la hora de realizar el proceso de entrenamiento y validación de datos en el clasificador.

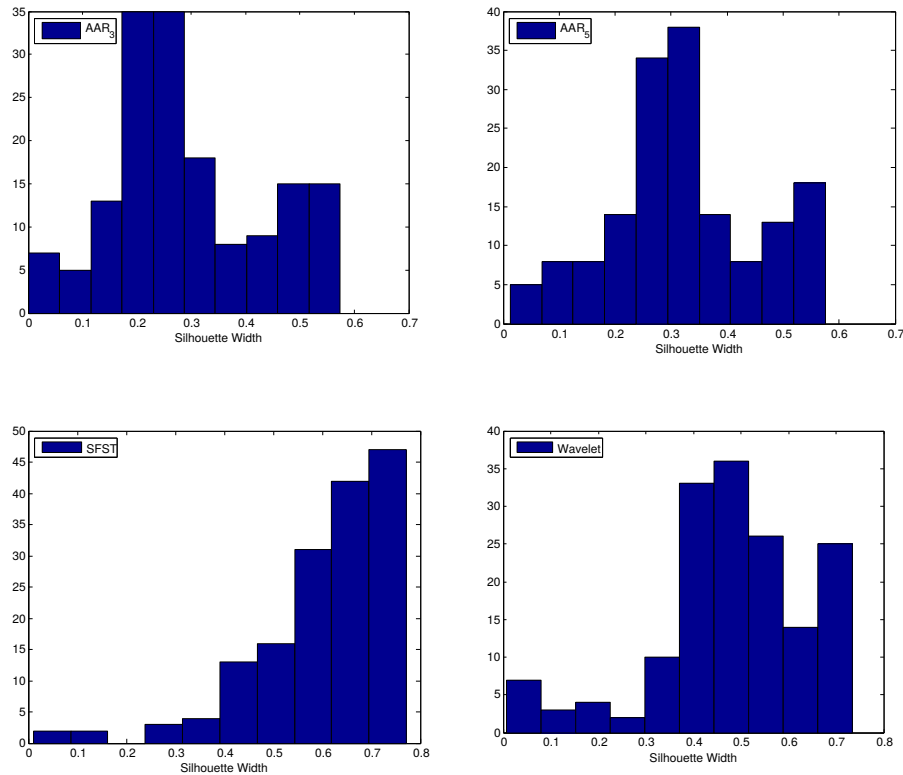


Figura 4.15: Distribución de las características usando SW para el usuario 4.

En la Tabla 4.5 son presentados los valores de tasa de acierto para cada método de extracción de características utilizados con los clasificadores SVM y K-NN, en esta misma tabla también son presentados los valores de SW por cada método y para cada usuario. Es claramente identificable que los usuarios 1 y 2, presentan los índices de desempeño más bajos y que su SW lo confirma. Pues los valores SW en los distintos métodos de extracción de características son inferiores a 0.5. Los usuarios 3 y 4 presentan altas tasas de desempeño y sus valores de SW son mayores a 0.5 mostrando que es posible realizar un análisis previo a los datos y poder seleccionar el método de extracción de características cuyo SW sea superior a 0.5.

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

Tabla 4.5: Tasa de aciertos y SW para cada método de extracción de características

Usuarios	SVM Exactitud [%]				KNN Exactitud [%]				Anchura de Silhouette			
	ARR(3)	ARR(5)	SFTS	Wavelet	ARR(3)	ARR(5)	SFTS	Wavelet	ARR(3)	ARR(5)	SFTS	Wavelet
1	0.4	0.41	0.43	0.36	0.34	0.38	0.48	0.5	0.12	0.06	0.18	0.24
2	0.38	0.39	0.53	0.42	0.6	0.39	0.68	0.54	0.08	0.05	0.14	0.13
3	0.74	0.76	0.74	0.73	0.72	0.75	0.75	0.75	0.32	0.38	0.59	0.61
4	0.88	0.87	0.93	0.91	0.84	0.83	0.9	0.9	0.29	0.32	0.6	0.48
Promedio	0.6	0.61	0.66	0.6	0.62	0.48	0.78	0.67	0.2	0.2	0.38	0.36

Luego de seleccionado el mejor método de extracción de características y el mejor clasificador se implementaron las diferentes propuestas basadas en SA, los cuales son presentados a continuación:

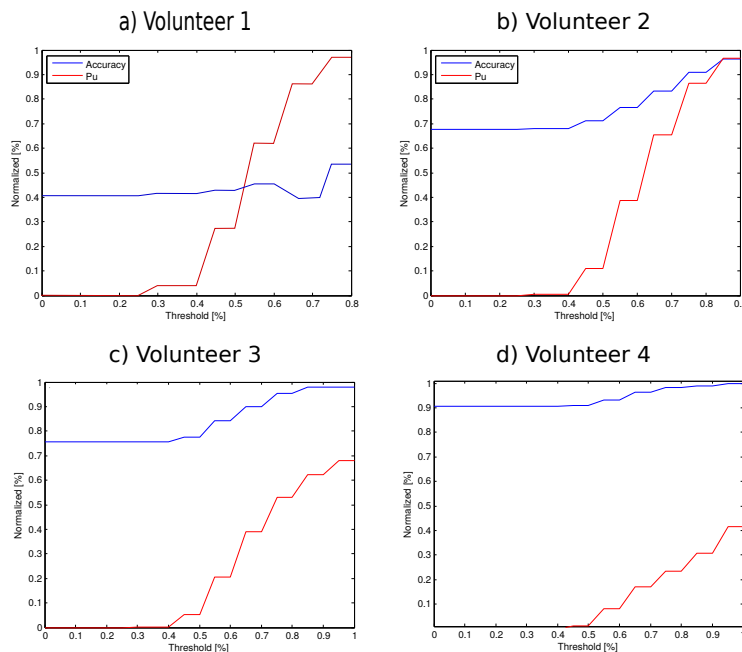


Figura 4.16: Resultados de un SA con umbral estático, Pu es el tiempo sin clasificar.

Tomando en cuenta lo detallado en la sección 3.5.2, iniciamos el análisis de una aBCI basada en SW con umbral estático, el cual es presentado en la Figura 4.16. Del usuario 1 se puede observar que su tasa de acierto máxima es del 0.6[%] (este valor es normalizado), con un tiempo de no clasificación del 0.96[%], lo que quiere decir que prácticamente el sistema no clasificaría satisfactoriamente los datos de este usuario. El valor referido como umbral esta relacionado con la probabilidad *a posteriori* del clasificador, para ma-

yor claridad, si el 0.6 [%] de los vecinos cercanos muestran que la clase a la que corresponde una muestra, en este usuario estaríamos obteniendo una tasa de aciertos de 0.52 [%] y un tiempo de no clasificación de 0.6 [%]. Para el usuario 2 podemos observar que su máxima tasa de acierto corresponde a 0.9 [%] con un tiempo de no clasificación del mismo valor. Para un umbral o probabilidad *a posteriori* del 0.6 [%] tendríamos una tasa de acierto del 0.75 [%] y un tiempo de no clasificación del 0.4 [%]. En los usuarios 3 y 4 se puede observar que la tasa de acierto para umbrales cercano a 0.6 [%] presentan tasas de acierto superiores al 0.8 [%] y tiempos de no clasificación inferiores al 0.3 [%]. Para el análisis de la aBCI basada en SA con

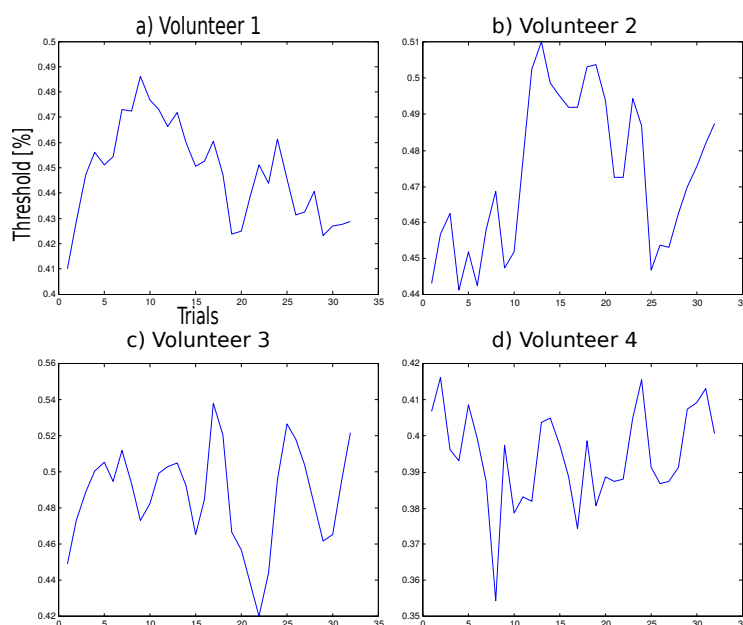


Figura 4.17: Resultados de un SA con umbral dinámico.

umbral dinámico, observamos las variaciones del umbral versus el número de trials. Esta figura muestra para los diferentes usuarios el rango de valores en los cuales el umbral se mueve para los diferentes trials. Para obtener el porcentaje de tasa de aciertos y tiempo de no clasificación hay que llevar el valor del umbral de esta figura y contrastarlo con la Figura 4.16 referida al umbral estático. En la aBCI con aprendizaje activo, tenemos que el umbral es dinámico y que adicionalmente solo en el caso del usuario 4 se observa

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

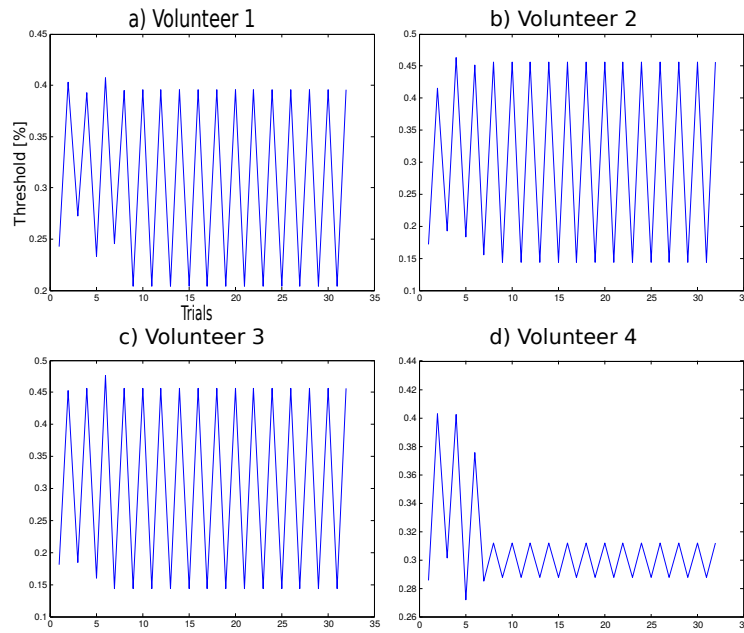


Figura 4.18: Resultados de un SA con aprendizaje activo.

un estrechamiento del valor. Para poder observar el efecto del aprendizaje activo en el usuario 1 nos remitiremos a la Figura 4.19, en esta figura se ha realizado una descomposición en componentes principales y se ha representado el conjunto de datos en las 2 principales componentes. Se muestra el

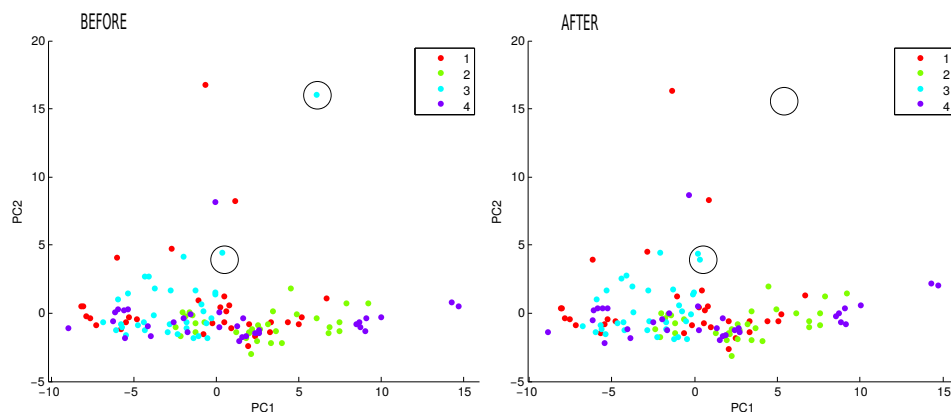


Figura 4.19: Representación en componentes principales de los datos del usuario 1 antes y después del SA con aprendizaje activo.

conjunto de datos antes y después del proceso de aprendizaje, y se muestra también los datos que sufren algún tipo de modificación en uno y otro caso (antes y después del aprendizaje) En el caso del usuario 1 se observa que las variaciones son muy pocas, se ha resaltado la más notoria.

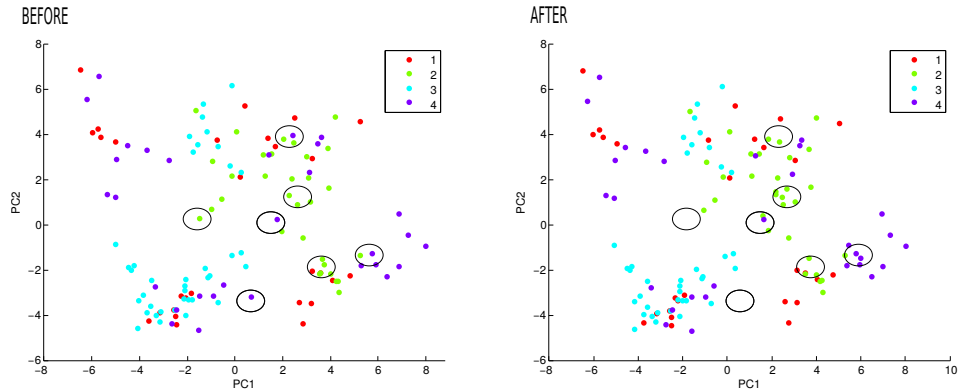


Figura 4.20: Representación en componentes principales de los datos del usuario 2 antes y después del SA con aprendizaje activo.

La Figura 4.20 muestra el usuario 2 presenta una mejor dispersión de los datos y tiene un conjunto mayor de estos afectados en el proceso de aprendizaje activo.

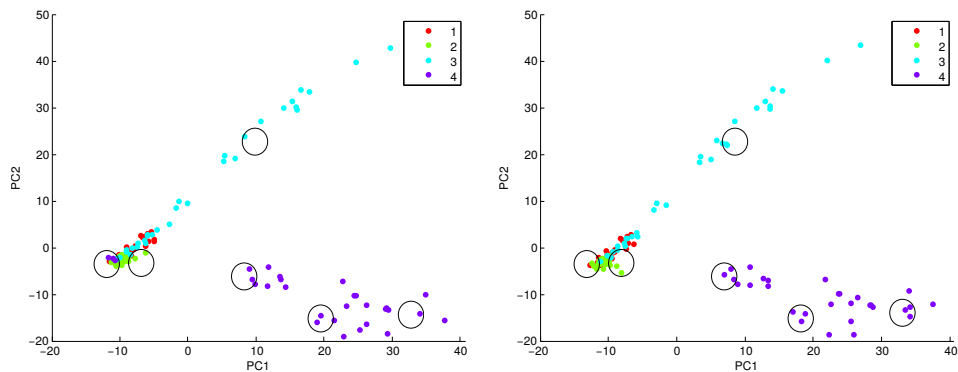


Figura 4.21: Representación en componentes principales de los datos del usuario 3 antes y después del SA con aprendizaje activo.

La Figura 4.21 muestra el usuario 3 presenta una dispersión de datos donde algunas clases están entre mezcladas, luego del proceso de aprendizaje se

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

logra observar una mayor separación entre clases, lo que se traduce en una mejor tasa de aciertos.

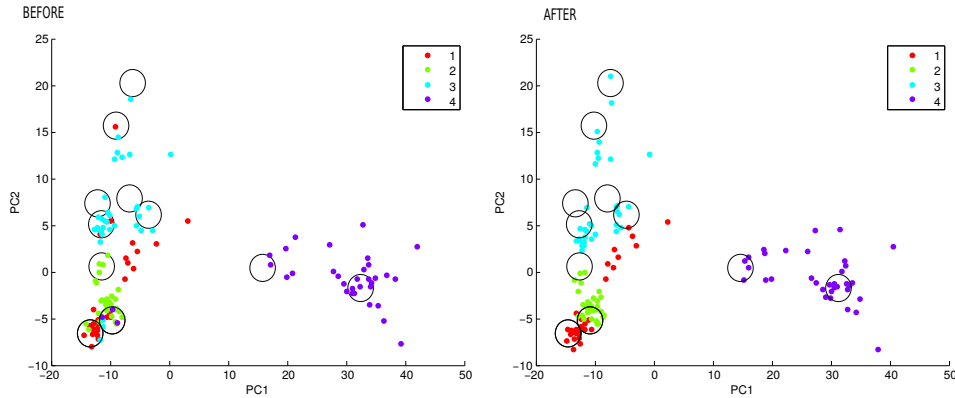


Figura 4.22: Representación en componentes principales de los datos del usuario 4 antes y después del SA con aprendizaje activo.

La Figura 4.22 muestra el usuario 4 presenta la mejor distribución de sus datos, antes del proceso de aprendizaje, aunque algunos datos se encuentran situados muy cerca de las clases que no les corresponden. Luego del proceso de aprendizaje activo, se observa que los datos se encuentran en los cluster que les corresponden.

Tabla 4.6: Tabla comparativa de las implementaciones aBCI basadas en SA para cuatro tareas mentales. BCI es la interfaz tradicional, Th FX es SA con umbral estático, Th Dn es SA con umbral dinámico y Ac Ln es SA con aprendizaje activo.

Usuarios	Paradigma	Sensitividad	Especificidad	Exactitud (2)	Kappa (2)	ITR	Exactitud (4)	Kappa (4)	Pu
Usuario 1	BCI	0.41	0.80	0.70	0.19	63.35	0.41	0.21	0.00
	Th Fx	0.46	0.79	0.70	0.15	6.29	0.39	0.16	0.90
	Th Dn	0.44	0.82	0.73	0.24	36.16	0.45	0.27	0.44
	Ac Ln	0.40	0.81	0.71	0.19	25.97	0.41	0.22	0.59
Usuario 2	BCI	0.66	0.89	0.83	0.55	76.97	0.66	0.55	0.00
	Th Fx	0.81	0.95	0.92	0.77	31.38	0.84	0.78	0.67
	Th Dn	0.70	0.90	0.86	0.61	66.66	0.71	0.62	0.18
	Ac Ln	0.72	0.91	0.87	0.63	55.72	0.73	0.64	0.33
Usuario 3	BCI	0.77	0.92	0.88	0.69	87.18	0.77	0.69	0.00
	Th Fx	0.86	0.97	0.95	0.83	62.79	0.90	0.80	0.39
	Th Dn	0.81	0.94	0.91	0.75	76.66	0.81	0.75	0.13
	Ac Ln	0.81	0.94	0.91	0.75	76.95	0.82	0.76	0.17
Usuario 4	BCI	0.91	0.97	0.95	0.87	104.35	0.91	0.87	0.00
	Th Fx	0.95	0.99	0.98	0.94	90.81	0.96	0.95	0.19
	Th Dn	0.91	0.97	0.95	0.88	103.3	0.91	0.88	0.01
	Ac Ln	0.91	0.97	0.96	0.89	98.40	0.92	0.89	0.07

En la Tabla 4.6 se muestran los resultados de las diferentes implementaciones de SA para aBCI. Las implementaciones de SA de acuerdo a esta tabla comparativa muestra que los SAs con umbral estático presentan los mejores desempeños con relación a la tasa de aciertos, pero presentan las menores ITRs. No existe diferencia significativa entre las implementaciones SA con umbral dinámico y SA con aprendizaje activo, salvo que la implementación con aprendizaje activo, permite que el usuario pueda evaluar su proceso de entrenamiento al mismo tiempo que usa la interfaz.

4.3 Protocolo de experimentación

El protocolo de experimentación implementado para el estudio y selección de las tareas mentales tiene como principal objetivo, permitir la repetitibilidad de los experimentos, garantizar la adquisición de la señal bajo las mejores condiciones técnicas posibles y buscar que tanto el usuario como las personas involucradas en el proceso tengan el mejor confort posible en la realización de las diferentes pruebas.

4.3.1 Descripción de los equipos

El equipo usado para las pruebas de entrenamiento activo, sección 4.3.5, fue el Emotiv EPOC™. Este equipo consta de 14-canales (128 Hz frecuencia de muestreo; 0.16 - 45 Hz ancho de banda). El equipo fue modificado para poder utilizar los electrodos en las posiciones requeridas para el análisis de tareas mentales ya que la disposición original del equipo esta estudios relacionados a emociones. Para esta modificación fue necesario remover los electrodos, y junto con el *hardware* trasladarlos a un gorro fabricado en Neoprene, En la Figura 4.23 se puede observar la disposición final del equipo modificado. Los catorce electrodos fueron posicionados de acuerdo al sistema 10/20, como puede observarse en la Figura 4.24,

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

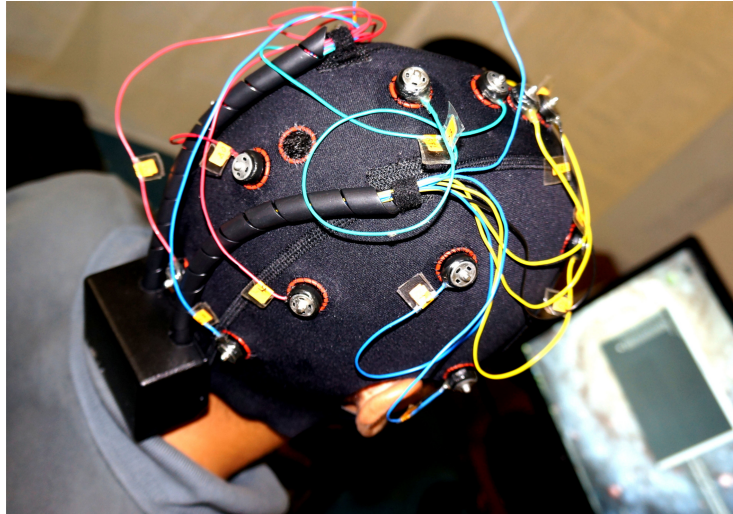


Figura 4.23: Gorro en Neoprene con el *hardware* modificado del Emotiv EPOC™.

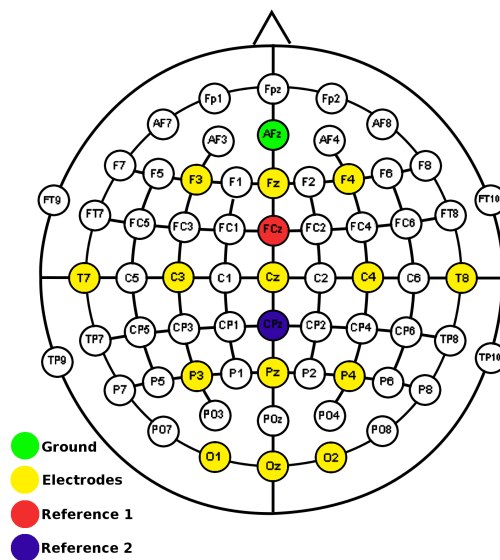


Figura 4.24: Posicionamiento de los electrodos

4.3.2 Descripción de los usuarios

Los usuarios participantes de las diferentes pruebas, fueron estudiantes Universitarios, los cuales no presentaban o registraban algún tipo de discapacidad, las edades oscilaban entre los 24 y 37 años de edad, y accedieron a participar de las pruebas firmando los términos de consentimiento que les fueron presentados oportunamente y fue debidamente explicado para cada participante.

4.3.3 Descripción de la tareas mentales

Las tareas mentales tienen su fundamentación básica, en los principios que rigen las funcionalidades de las distintas regiones del cerebro. Con la finalidad de explorar la gama de posibilidades que este ofrece se establecieron las siguientes tareas mentales:

- Tareas mentales Visuales

Bajo este término se seleccionaron, aquellas que utilizan el canal de representación de la realidad basado en imágenes y que corresponden a las utilidades de la región occipital, parietal y frontal.

- Tareas mentales Auditivas

Estas consistían en imaginar sonidos para utilizar el canal de representación auditivo usando las regiones temporal y frontal específicamente.

- Tareas mentales Kinestésicas

Estas pretendían que el usuario utilizara elementos propioceptivos localizados en la región somato-sensorial del cerebro.

- Tareas mentales Motoras

Estas hacen parte de las tareas mentales kinestésicas, solo que son referidas ampliamente en la literatura y se decidió dar un espacio específico, para imaginar movimiento de las extremidades.

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

4.3.4 Criterios de Selección de la tareas mentales

Para la selección de la tareas mentales que presentaban la mejor separación temporal - frecuencial y espacial, es necesario tener en cuenta los siguientes parámetros:

- Complejidad de la tarea
El análisis inter-trial, el cual afirma que si determina acción es ejecutada repetidas veces de la misma forma debe existir una correlación entre las diferentes repeticiones.
- Tiempo de ejecución de la tarea
Se puede establecer por el análisis de la varianza de la nube de datos en los instantes antes y despues de la ejecución de la tarea.
- Carga cognitiva para ejecutar la tarea
Para determinar el nivel de carga cognitiva se puede utilizar el índice NASA TLX el cual relaciona los niveles de sobre carga en la ejecución de una tarea o actividad cognitiva dada.
- Zonas de mayor activación por tareas
Las zonas mayor activación por tareas se observa a través del análisis de frecuencia-tiempo para cada canal durante la tarea específica.
- Conexiones de las regiones para cada tarea
La ejecución de tareas puede estar representación por la activación secuencial de algunas regiones cerebrales este puede ser analizado a través de la coherencia y el factor de fase bloqueada.

4.3.5 Entrenamiento activo

El entrenamiento activo es una estrategia propuesta para el proceso de aprendizaje de una aBCI, dado que permite obtener una mejor calidad de los datos. El entrenamiento tradicional es implementado para obtener un modelo que mejor represente el conjunto de datos del sistema BCI. Cada experimento requiere un número fijo de muestras de entrenamiento los cuales no son re-alimentados para el usuario. El sistema de entrenamiento activo permite que

4.3 Protocolo de experimentación

el usuario observe la salida del sistema y utilice aquellas muestras que mejoren el desempeño del mismo.

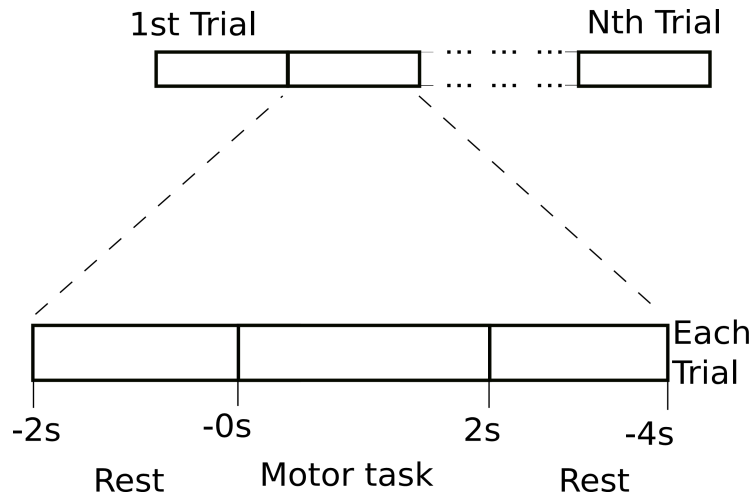


Figura 4.25: Protocolo experimental para entrenamiento activo para aBCI.

En la Figura 4.25 se presenta el protocolo experimental. Las tareas definidas para este tipo de entrenamiento fueron 5 tareas de imaginación motora:

1. Movimiento de la mano derecha
2. Movimiento de la mano izquierda
3. Movimiento de ambas manos
4. Manipulación de un cubo usando ambas manos
5. Movimiento de las dos piernas

Para establecer cuando el sistema considera un conjunto de datos como válidos, cada entrada es clasificada y si el coeficiente de $kappa$ es $> 0,6$ (buena concordancia) el sistema finaliza el proceso de entrenamiento para esta tarea. Si el número de *trials* es > 10 el primer *trial* es descartado y el sistema realizaba un re-aprendizaje (aprendizaje activo con olvido). Luego de entrenado el sistema, este pasa a modo de operación en línea, para validar la prueba y almacenar los *trials* de entrenamiento. Cada tarea mental es entrenada con una fase de relajación o *idle state*. Para seleccionar la mejor tarea mental,

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

fue implementado el índice para sobre carga (NASA TLX). En la figura 4.26 se puede observar las preguntas relacionadas a seis aspectos fundamentales para la ejecución de una tarea, como son: demanda mental, demanda física, demanda temporal, desempeño, esfuerzo y frustración. Cada aspecto tiene relacionado una evaluación subjetiva por parte del usuario, la cual era contrastada por una valoración que permitía cuantificar la sobre carga percibida por el usuario.

The image shows a screenshot of the NASA Task Load Index (NASA TLX) form. The form is titled "NASA Task Load Index" and includes fields for "Name", "Task", and "Date". Below these fields are six horizontal sliders, each representing a different aspect of task load. Each slider has a scale from "Very Low" to "Very High". The sliders are labeled as follows:

- Mental Demand:** How mentally demanding was the task?
- Physical Demand:** How physically demanding was the task?
- Temporal Demand:** How hurried or rushed was the pace of the task?
- Performance:** How successful were you in accomplishing what you were asked to do? (Scale: Perfect to Failure)
- Effort:** How hard did you have to work to accomplish your level of performance?
- Frustration:** How insecure, discouraged, irritated, stressed, and annoyed were you?

An "Ok" button is located at the bottom right of the form.

Figura 4.26: Índice NASA TLX para sobre carga.

En la figura 4.27 se presenta una de las 15 combinaciones de las características para cuantificar la percepción de sobre carga por parte del usuario. La evaluación de los aspectos más relevantes para la ejecución de una tarea se

4.3 Protocolo de experimentación

Nasa TLX

Mental Demand

How much mental and perceptual activity was required (e.g. thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?

Physical Demand

How much physical activity was required (e.g. pushing, pulling, turning, controlling, activating, etc.)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?

Temporal Demand

How much time pressure did you feel due to the rate or pace at which the task or task elements occurred? Was the pace slow and leisurely or rapid and frantic?

Performance

How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals?

Effort

How hard did you have to work (mentally and physically) to accomplish your level of performance?

Frustration Level

How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

Of the two workload measures below, which one contributed the most to the task you just completed?

Mental Demand

or

Physical Demand

Ok

Figura 4.27: Evaluación de un par de características para cuantificar el índice NASA TLX

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

Tabla 4.7: Resultados para entrenamiento activo para una aBCI.

Tareas	Número de Trials	Kappa	Sensibilidad	Especificidad	Exactitud	TLX
1	5	0.6	0.8	0.9	0.9	54.6
2	7	1.0	0.8	0.2	0.5	53.3
3	24	0.8	1.0	0.9	1.0	62.1
4	5	0.6	0.9	0.4	0.7	52.9
5	10	0.6	0.9	0.1	0.5	51.9

realiza haciendo una comparación uno a uno los diferentes aspectos presentados en la Figura 4.26.

En la Tabla 4.7 es presentada cada tarea, el número promedio de trial utilizados, el valor del coeficiente de Kappa, la exactitud y el valor TLX para cada tarea. La tarea con mayor índice NASA TLX requirio un mayor número de repeticiones y la mayor tasa de acierto, lo que permite inferir que cuando una tarea mental presenta una mayor carga cognitiva el usuario requiere una mayor concentración en la ejecución de la misma.

4.4 Discusión y Análisis de Resultados

El proceso de selección de tareas de acuerdo a lo registrado en la literatura básicamente ha surgido de la necesidad de utilizar los canales de representación de la realidad fácilmente utilizados por los usuarios, esto quiere decir que la utilización de tareas basadas en imaginación motora son las de más amplio uso por la facilidad que representan para el usuario. Actualmente, no existe un mecanismo para evaluar y puntuar cuales serían las mejores tareas para ser usada en una interfaz, lo que existe es escoger aquellas que presenten la mejor tasa de aciertos posible. El esquema propuesto de entrenamiento activo, permite escoger el número de *trials* necesarios para que un usuario determinado ejecute de la mejor manera posible una tarea mental, independiente del tipo de canal de representación utilizado (visual, kinestesico, etc). No es posible asevera que aquellas tareas con mayor valoración en terminos del índice NASA TLX serían las mejores, pero hay algo claro en este aspecto, relacionado a la repetitibilidad de la misma. Pues el hecho de presentar un mayor grado de dificultad para el usuario, este pondrá mayor empeño en

ejecutar la tarea. La desventaja de esta selección, está relacionada a la carga cognitiva en la utilización de este tipo de tarea. Este es un aspecto que afecta el confort y usabilidad de la interfaz.

Se obtuvo buenos resultados en el proceso de clasificación de cinco tareas mentales de forma simultánea. La variabilidad, producto de la heterogeneidad de los usuarios es compensada por la adaptación hecha en la interfaz, permitiendo así definir una topología óptima para cada usuario. La función cumplida por el agente de software reduce la carga computacional que supone determinar de forma combinatoria todos los vectores de ensamble para definir el subconjunto de características óptimo. En la presente propuesta se explora la aplicabilidad de los diferentes clasificadores, pues no en todos los casos las máquinas de vectores de soporte resulta siendo la mejor clasificadora. El modelo neuronal para aBCI permite adaptar diferentes topología para varias aplicaciones; cada usuario como cada tarea presentan valores diferentes con relación a la característica y/o clasificador seleccionado. El usuario 2 presentó mejores desempeños usando solo dos canales; estos canales tienen que ver el con la imaginación visual (O2) y el otro canal con la ubicación espacial (P4).

La tasa de acierto es la medida principal de una BCI. Sin embargo, la sensibilidad y especificidad proporcionan información sobre la capacidad de reconocer clases particulares. El coeficiente Kappa proporciona información sobre si la precisión es cercana al valor del azar. Para evaluar el método propuesto se analizaron las medidas de desempeño y la significancia estadística. El agente de software puede interactuar con el medio ambiente mediante la comprensión de las señales del cerebro y la salida del clasificador. Por otro lado, el esquema propuesto aquí permite modificar la extracción de características y parámetros del clasificador. La calidad de la señal es el factor principal en el diseño de un BCI, porque el rendimiento del sistema depende de las características de las señales. En este estudio, se incluyó el rechazo de trials por parámetro de ajuste en el bloque traductor a comandos.

Con relación al método de selección de características utilizando la SW, se puede inferir que es muy útil, principalmente porque reduce el costo compu-

4. PRUEBAS, ANÁLISIS DE DATOS Y RESULTADOS

tacional que supondría realizar todo el proceso de validación cruzada para seleccionar las características que podrían modelar mejor una BCI. La SW al relacionar la cohesión y separación de los trials con relación al proceso de cluster de los datos.

Las implementaciones relacionadas a los agentes software para una interfaz cerebro computador adaptativa muestran diferentes grados de aplicabilidad. Aquellos sistemas con umbral estático permiten ser usados en aplicaciones donde se requiere alta precisión con relación al hecho de generar o no comandos cuando la interfaz consigue discriminar las tareas presentadas. Los sistemas basados en umbral dinámico están más relacionados con aplicaciones básicas de BCI donde se pueden tolerar algunos errores en la clasificación. Finalmente los sistemas basados en aprendizaje activo, son especialmente interesantes para mejorar el proceso de uso de la Interfaz, es decir, como existe la posibilidad de sintonizar el clasificador por medio del uso, esto facilitaría el entrenamiento activo de la interfaz. Esta misma interfaz puede ser interesante su uso en aplicaciones de rehabilitación ya que permitiría observar el grado de evolución de un usuario con relación a la cohesión o separabilidad de sus muestras conforme usa la interfaz. En la tabla 4.8 se presenta un resumen de los resultados más relevantes obtenidos en el desarrollo de esta tesis de doctorado.

4.4 Discusión y Análisis de Resultados

Tabla 4.8: Resumen de los resultados obtenidos.

Resultado	Método de evaluación	Aplicación
Entrenamiento activo	NASA TLX	Selección de tareas
Clasificación 5 tareas agente software (algoritmos genéticos)	Exactitud ITR Kappa Exactitud	Representación modelo neuronal Selección extraccion características selección clasificador
Selección de características	Anchura de Silhouette Exactitud	Reducción en la evaluación de métodos de extracción de características
Agente umbral estático	ITR Exactitud	Alta exactitud y facilidad discriminación de tareas
Agente umbral dinámico	ITR Exactitud	Tolerancia a errores en la clasificación dificultad presentada en el usuario o el sistema de discriminacion
Agente aprendizaje activo	ITR Exactitud Silhouette	Adaptación usuario/interfaz rehabilitación

Ahora bien, este no es el final. Ni siquiera es el principio del fin. Pero es, quizás, el final del principio.

Winston Churchill

CAPÍTULO

5

Conclusiones

5.1 Introducción

Se presentó el diseño de una aBCI donde la adaptabilidad está relacionada al proceso de reconocimiento de clases. En este trabajo doctoral se dio gran énfasis al uso de agentes de software y su posibilidad de integración al esquema de las BCIs tradicionales.

5.2 Conclusiones sobre el problema de investigación

Se presenta un método de diseño de interfaces de usuario donde explícitamente se logra modelar las capacidades de adaptación de una interfaz cerebro computador, incluyendo su diseño dentro del ciclo de vida de una aplicación, es decir, se puede tener una BCI tradicional y se adicionan algunos bloques donde queda claramente diferenciada la interfaz y el módulo de adaptación. Se consiguió la integración total del diseño de la adaptación en el proceso de desarrollo de las interfaces cerebro computador. En este diseño se modelaron los constructores que describen el proceso de adaptación (proceso

5. CONCLUSIONES

de selección de características, ajuste de parámetros del clasificador y aprendizaje activo). Por otra parte, se incluyó el uso de agentes de software para inferir conocimiento a partir de la información recogida de la interacción con el usuario, y este toma decisiones basándose en dicha información. La interacción fue evaluada desde el análisis de la calidad señal de entrada y el nivel de compromiso en la ejecución de la tarea.

5.3 Implicaciones teóricas

Se diseñó una interfaz adaptativa para la comunicación cerebro computador, y fueron presentadas diferentes propuestas para el proceso de integración de agentes de software en aplicaciones de interfaces cerebro computador.

Se presentó un esquema que caracterizó los estados metales, presentando un gran énfasis en una metodología de selección y evaluación de tareas mentales.

La estrategia de integración de agentes software en la identificación de estados mentales para una interfaz adaptativa, trajo consigo la generación de diferentes abordajes para la implementación, desde el esquema básico de selección de características y clasificador hasta el proceso de agentes con menor o mayor grado de interacción con el entorno (tareas mentales y señales EEG).

Se propuso una estrategia de entrenamiento para una interfaz adaptativa, y se logró identificar que el esquema de una aBCI con SA bajo el paradigma de aprendizaje activo puede ser utilizado como sistema de entrenamiento para la misma interfaz.

Se desarrolló un modelo de representación neuronal donde se relacionaban: los canales, los métodos de extracción de características y el mejor clasificador para el reconocimiento de al menos cuatro tareas mentales empleando señales EEG y máquinas de aprendizaje.

5.4 Implicaciones prácticas

El sistema desarrollado permite seleccionar los mejores métodos de extracción de características adaptado para cada usuario, como el o los clasificadores que permitan discriminar al menos cuatro tareas mentales. Los esquemas de agentes software implementados permiten diferentes tipos de aplicaciones y sus funcionalidades van dirigidas a optimizar la tasa de aciertos y el coeficiente de Kappa de la aBCI. El sistema propuesto para el entrenamiento permite tener una aBCI como sistema de entrenamiento activo, ya que hace uso de la interfaz basada en SA y aprendizaje activo para este fin. El proceso de selección de características usando la SW facilita el proceso de búsqueda y reduce el costo computacional en la selección de características.

5.5 Limitaciones

La selección de las tareas mentales dependen de las habilidades cognitivas del usuario y es difícil generalizar. Las mejores tareas mentales están relacionadas a la habilidad del usuario para reproducir los mismos patrones cerebrales en la ejecución de una tarea mental dada y no está relacionada a la carga cognitiva de la tarea, por ejemplo, si se seleccionan tareas que requieran una alta carga cognitiva facilitan la concentración del usuario, pero trae consigo un mayor nivel de fatiga del usuario lo que no permitiría su uso de forma continua. Las estrategias adaptativas, necesitan otros parámetros o indicadores de desempeño que permitan evaluar el grado de adaptabilidad y poder comparar diferentes interfaces adaptativas.

5.6 Investigaciones Posteriores

En los trabajos futuros se proponen validar los esquemas a nivel aplicaciones prácticas de la interfaz desarrollada, así como, realizar implementaciones de los agentes de software usando aprendizaje por refuerzo, en esto se consideran los aspectos formales de este tipo de implementación. También, se propone analizar otro tipo de implementación para agentes de software que tiene

5. CONCLUSIONES

que ver con el uso de lógica difusa, lo que facilitaría el ajuste de parámetros. Se sugiere como trabajo futuro realizar un estudio sobre la anchura de Silhouette como índice de adaptabilidad de interfaz, debido a que este permite relacionar la cohesión y la separación de los datos y de esta forma evaluar el grado de adaptabilidad del usuario en el uso de la interfaz.

5.7 Publicaciones

Se presenta una relación de las producciones científicas generadas, fruto de las investigaciones realizadas en el área de las Interfaces Cerebro computador:

Capítulos de libros publicados

1. VALADÃO, C.; SARMIENTO, J.; GOULART, C.; **CASTILLO-GARCIA, J.**; BASTOS, T. Manipulation Technologies for Developing Cognitive Skills. In: Teodiano Freire Bastos-Filho; Dinesh Kant Kumar; Sridhar Arjunan. (Org.). Devices for Mobility and Manipulation for People with Reduced Abilities (Rehabilitation Science in Practice Series). 1ed.: CRC Press, 2014, v. 1, p. 167-178.
2. AZORIN, J. M.; IÁNEZ, E.; UBEDA, A.; **CASTILLO-GARCIA, J.**; CAICEDO, E.; FERREIRA, A.; MULLER, S.; BASTOS, T.; SARCI-NELLI, M.; LIMA, E. R. Interfaces baseadas na atividade cerebral. In: José María Azorín; Ramón Ceres; Anselmo Frizera Neto; Teodiano Freire Bastos. (Org.). A Interação de Pessoas com Deficiência com o Computador. 1ed.Madri: CYTED, 2014, v. 1, p. 125-146.
3. AZORIN, J. M.; IÁNEZ, E.; UBEDA, A.; **CASTILLO-GARCIA, J.**; CAICEDO, E.; FERREIRA, A.; MULLER, S.; BASTOS, T.; LIMA, E. R. Interfaces basadas en la actividad cerebral. In: José María Azorín; Ramón Ceres; Anselmo Frizera Neto; Teodiano Freire Bastos. (Org.). La interacción de personas con discapacidad con el computador. 1ed.Madri: CYTED, 2013, v. 1, p. 125-145.

4. COTRINA-ATENCIO, A.; **CASTILLO-GARCIA, J.**; TRAUERNICHT, M.; DOLORES, M.; FERREIRA, A.; BASTOS, T. Toward a hybrid Brain-Computer Interface. *Tecnologias, Técnicas e Tendências em Engenharia Biomédica*. Canal6 Editora, 2014.
5. GOULART, C.; POMER-ESCHER, A.; TRAUERNICHT, M.; **CASTILLO-GARCIA, J.**; BASTOS, F. Analysis of emotions based on EEG in autistic spectrum disorder and stress. *Tecnologias, Técnicas e Tendências em Engenharia Biomédica*. Canal6 Editora, 2014.
6. MULLER, S.; **CASTILLO-GARCIA, J.**; TELLO R.; FERREIRA A.; CAICEDO E.; BASTOS T. Brain-Computer Interface applied to a robotic wheelchair and an autonomous car. *Tecnologias, Técnicas e Tendências em Engenharia Biomédica*. Canal6 Editora, 2014.
7. TRAURNICHT M.; **CASTILLO-GARCIA, J.**; COTRINA-ATENCIO A.; GOULART C.; POMER ALEXANDRE; CALDEIRA E.; BASTOS, T. Towards EEG-Based Emotion Recognition Using Brain Computer Interfaces. *Tecnologias, Técnicas e Tendências em Engenharia Biomédica*. Canal6 Editora, 2014.
8. LONGO B.; **CASTILLO-GARCIA, J.**; VILLA C.; DELISLE-RODRIGUEZ D.; BASTOS T. EEG-BCI technologies for Neurorehabilitation. *Tecnologias, Técnicas e Tendências em Engenharia Biomédica*. Canal6 Editora, 2014.

Trabajos completos publicados en congresos

1. BENAVIDES A.; COTRINA-ATENCIO A.; BENAVIDES A.; **CASTILLO-GARCIA, J.**; BASTOS, T. An Ethernet Sniffer for On-line Acquisition of EEG with the BrainNet36 R Device Applied to a BCI. In: 5th IEEE Biosignals and Biorobotics Conference (BRC 2014), 2014, Salvador. Proc. of the 5th IEEE Biosignals and Biorobotics Conference, 2014. p. 6 pg.

5. CONCLUSIONES

2. **CASTILLO-GARCIA, J.**; MULLER, S.; CAICEDO, E.; BASTOS, T. Feature Extraction Techniques Based on Power Spectrum for a SSVEP-BCI. In: IEEE Int. Symp. on Industrial Electronics (ISIE 2014), 2014, Istambul. Proc. of the IEEE Int. Symp. on Industrial Electronics, 2014. p. 1052-1055.
3. COTRINA-ATENCIO A.; **CASTILLO-GARCIA, J.**; BENEVIDES, A.; LONGO, B. B.; FERREIRA, A.; POMER-ESCHER, A.; SOUZA, M. D. P.; BASTOS, T. Computing Stress-Related Emotional State via Frontal Cortex Asymmetry to be Applied in Passive-ssBCI. In: 5th IEEE Biosignals and Biorobotics Conference (BRC 2014), 2014, Salvador. Proc. of the 5th IEEE Biosignals and Biorobotics Conference, 2014. p. 6 pg..
4. LONGO, B. B.; BENEVIDES, A.; **CASTILLO-GARCIA, J.**; BASTOS, T. Using Brain-Computer Interface to Control an Avatar in a Virtual Reality Environment. In: 5th IEEE Biosignals and Biorobotics Conference (BRC 2014), 2014, Salvador. Proc. of the 5th IEEE Biosignals and Biorobotics Conference, 2014. p. 6 pg..
5. **CASTILLO-GARCIA, J.**; MULLER, S.; CAICEDO, E.; SOUZA, Alberto Ferreira de; BASTOS, T. Proposal of a Brain Computer Interface to Command an Autonomous Car. In: 5th IEEE Biosignals and Biorobotics Conference (BRC 2014), 2014, Salvador. Proc. of the 5th IEEE Biosignals and Biorobotics Conference, 2014. p. 6 pg..
6. DELISLE, D.; **CASTILLO-GARCIA, J.**; BASTOS, T.; FRIZERANETO, Anselmo; DELIS, A. L. Using Linear Discriminant Function to Detect Eyes Closing Activities through Alpha Wave. In: 5th IEEE Biosignals and Biorobotics Conference (BRC 2014), 2014, Salvador. Proc. of the 5th IEEE Biosignals and Biorobotics Conference, 2014. p. 6 pg..
7. **CASTILLO-GARCIA, J.**; MULLER, S.; COTRINA-ATENCIO A.; CAICEDO, E.; BASTOS, T. Comparison Among Feature Extraction Techniques Based on Power Spectrum for a SSVEP-BCI. In: International Conference on Industrial Informatics (INDIN2014), 2014, Porto

- Alegre. Proc. of the International Conference on Industrial Informatics (INDIN2014), 2014. p. 1-6.
8. **CASTILLO-GARCIA, J.**; COTRINA-ATENCIO A.; BENEVIDES, A.; DELISLE, D.; LONGO, B. B.; CAICEDO, E.; BASTOS, T. Adaptive BCI Based on Software Agents. In: 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC'14), 2014, Chicago. Proc. of the 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2014. p. 1-4.
 9. COTRINA-ATENCIO A.; BENEVIDES, A.; FERREIRA, A.; BASTOS, T.; **CASTILLO-GARCIA, J.**; MENEZES, M.; PEREIRA, C. Towards an Architecture of a Hybrid BCI Based on SSVEP-BCI and Passive-BCI. In: 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC'14), 2014, Chicago. Proc. of the 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2014. v. 1. p. 1-4.
 10. **CASTILLO-GARCIA, J.**; CAICEDO, E.; BASTOS, T. Interfaz Cerebro Computador Adaptativa Basada en Agentes de Software. In: Congresso Brasileiro de Automática, 2014, Belo Horizonte. Anais do Congresso Brasileiro de Automática, 2014. v. 1. p. 1-6.
 11. COTRINA-ATENCIO A.; BENEVIDES A.; FERREIRA A.; BASTOS T.; **CASTILLO-GARCIA, J.**; MENEZES M.; PEREIRA C. Evaluation of Emotional Components to Improve a SSVEP-BCI. In: Congresso Brasileiro de Automática, 2014, Belo Horizonte. Anais do Congresso Brasileiro de Automática, 2014. p. 1-6.
 12. COTRINA-ATENCIO A.; **CASTILLO-GARCIA, J.**; FERREIRA, A.; BASTOS, T. Towards a SSVEP-BCI Based on Depth of Field. In: 6th International Brain-Computer Interface Conference 2014, 2014, Graz. Proc. of the 6th International Brain-Computer Interface Conference 2014, 2014. v. 1. p. 1-6.
 13. DELISLE, D.; **CASTILLO-GARCIA, J.**; BASTOS T.; FRIZERA A.; DELIS A. L. Detection of Eyes Closing Activities through Alpha Wave

5. CONCLUSIONES

- by Variability Analysis. In: 6th International Brain-Computer Interface Conference 2014, 2014, Graz. Proc. of the 6th International Brain-Computer Interface Conference 2014, 2014. p. 1-6.
14. **CASTILLO-GARCIA, J.**; CAICEDO, E.; LONGO, B. B.; FLORIANO, A.; BASTOS, T. Motor Imagery Based on Wavelet Power Spectrum for a Brain Computer Interface. In: Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014, Uberlândia. Anais do Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014. p. 4 pg.
 15. GOULART, C.; **CASTILLO-GARCIA, J.**; TRAUERNICHT, M.; VALADÃO, C.; Eliete Maria de Oliveira Caldeira; BASTOS, T. Estudo de Estados Emocionais e Mentais de Crianças com Autismo Baseado em EEG na Interação com um Robô Móvel. In: Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014, Uberlândia. Anais do Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014. p. 4 pg.
 16. LONGO, B. B.; FLORIANO, A.; **CASTILLO-GARCIA, J.**; BASTOS, T. A Multi-Proposal Mobile EEG System. In: Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014, Uberlândia. Anais do Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014. p. 4 pg.
 17. DELISLE D.; PARRA C.; **CASTILLO-GARCIA, J.**; MORALES C.; BASTOS, T.; FRIZERA A.; DELIS A. L. Development of an EEG and sEMG Wireless System for a Robotic Walker. In: Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014, Uberlândia. Anais do Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014. p. 4 pg.
 18. FLORIANO, A.; **CASTILLO-GARCIA, J.**; LONGO, B. B.; BASTOS, T. Proposal of a Telepresence Robot Using Brain Computer Interface for People with Motor Disabilities. In: Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014, Uberlândia. Anais do Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014. p. 4 pg.
 19. POMER-ESCHER, A.; Tello, R.; **CASTILLO-GARCIA, J.**; BASTOS, T. Analysis of Mental Fatigue in Motor Imagery and Emotional Stimulation Based on EEG. In: Congresso Brasileiro de Engenharia

Biomédica (CBEB), 2014, Uberlândia. Anais do Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014. p. 4 pg.

20. **CASTILLO-GARCIA, J.**; DELISLE, D.; POMER-ESCHER, A.; FRIZERA NETO, Anselmo; LIMA, E. R.; BASTOS, T. Neurofeedback Tool to Improve the Onset Delay and Sequential Movements During Motor Tasks. In: Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014, Uberlândia. Anais do Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014. p. 4 pg.
21. COTRINA-ATENCIO A.; **CASTILLO-GARCIA, J.**; BENEVIDES, A.; LONGO, B.; FERREIRA, A.; BASTOS, T. Statistical Spectral Power Evaluation of Alpha Rhythms for Hybrid-BCI. In: Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014, Uberlândia. Anais do Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014. p. 4 pg.
22. LONGO, B. B.; **CASTILLO-GARCIA, J.**; BASTOS, T. Controle de um Avatar em um Ambiente Virtual Baseado em uma Interface Cérebro-Computador. In: Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014, Uberlândia. Anais do Congresso Brasileiro de Engenharia Biomédica (CBEB), 2014. p. 4 pg.
23. **CASTILLO-GARCIA, J.**; LONGO, B. B.; FLORIANO, A.; CAICEDO, E.; BASTOS, T. Optimización de una Intefaz Cerebro Computador Basada en Imaginería Motora Usando Emotiv Epoc. In: VI Jornadas AITADIS de Rehabilitación y Tecnologías de Apoyo a la Discapacidad, 2014, Asunción. Memorias de las VI Jornadas AITADIS de Rehabilitación y Tecnologías de Apoyo a la Discapacidad, 2014. p. 4 pg.
24. **CASTILLO-GARCIA, J.**; DELISLE, D.; POMER-ESCHER, A.; CAICEDO, E.; FRIZERA A.; LIMA, E.; BASTOS, T. Propuesta de una ICC como Herramienta de Neuro-Realimentación. In: VI Jornadas AITADIS de Rehabilitación y Tecnologías de Apoyo a la Discapacidad, 2014, Asunción. Memorias de las VI Jornadas AITADIS de Rehabilitación y Tecnologías de Apoyo a la Discapacidad, 2014. p. 4 pg.

5. CONCLUSIONES

25. **CASTILLO-GARCIA, J.**; CAICEDO, E.; BASTOS, T.; CALDEIRA E.; SOUZA A.; CHEEIN F. Propuesta de Interfaz Cerebro Computador para el Comando de un Vehículo Autónomo. In: VII Congreso Iberoamericano de Tecnologías de Apoyo a la Discapacidad - Iberdiscap 2013, 2013, Santo Domingo. Memorias del VII Congreso Iberoamericano de Tecnologías de Apoyo a la Discapacidad - Iberdiscap 2013, 2013. p. 4 pg.
26. COTRINA-ATENCIO A.; BENEVIDES, A.; ALVES, S.; **CASTILLO-GARCIA, J.**; FERREIRA, A.; BASTOS, T. Towards a BCI-Speller Based on Binary Selection. In: 1st International Workshop on Assistive Technology, 2015, Vitória. Proceedings of the 1st International Workshop on Assistive Technology, 2015. p. 221-224.
27. GOULART, C.; **CASTILLO-GARCIA, J.**; VALADÃO, C.; Eliete Maria de Oliveira Caldeira; BASTOS, T. Study of EEG Signals to Evaluate Emotional and Mental States of Children with ASD in the Interaction with Mobile Robot. In: 1st International Workshop on Assistive Technology, 2015, Vitória. Proceedings of the 1st International Workshop on Assistive Technology, 2015. p. 217-220.
28. **CASTILLO-GARCIA, J.**; HORTAL, E.; CAICEDO, E.; BASTOS T.; AZORIN, J. M. Feature Selection Based on Silhouette s Width for Spontaneous Brain Computer Interface. In: 1st International Workshop on Assistive Technology, 2015, Vitória. Proceedings of the 1st International Workshop on Assistive Technology, 2015. p. 162-165.
29. **CASTILLO-GARCIA, J.**, HORTAL, E., CAICEDO, E., IÁÑEZ E., BASTOS, T., AZORIN J. Active Learning for Adaptive Brain Machine Interface Based on Software Agent. 23rd Mediterranean Conference on Control and Automation, Torremolinos, Spain (MED 2015).

Resúmenes publicados en congresos

1. GOULART, C.; **CASTILLO-GARCIA, J.**; VALADÃO, C.; BASTOS, T. EEG Analysis and Mobile Robot as Tools for Emotion Characteriza-

- tion in Autism. In: 5° Congresso Brasileiro de Biotecnologia, 2013, Florianópolis. Anais do 5o Congresso Brasileiro de Biotecnologia, 2013. p. 1.
2. GOULART, C.; **CASTILLO-GARCIA, J.**; VALADÃO, C.; BASTOS, T.; CALDEIRA, E. EEG analysis and mobile robot as tools for emotion characterization in autism. BMC Proceedings, v. 8, p. P85, 2014.
 3. LONGO, BERTHIL; **CASTILLO-GARCIA, J.**; BASTOS, T-. Brain-Computer Interface (BCI) combined with Virtual Reality Environment (VRE) for inferior limbs rehabilitation in post-stroke subjects. BMC Proceedings, v. 8, p. P18, 2014.

Premios

1. *Travel Scholarship IEEE-INDIN2014*, por el artículo **Comparison Among Feature Extraction Techniques Based on Power Spectrum for a SSVEP-BCI**.
2. *Brain Computer Interface Competition no XXIV Congresso Brasileiro de Engenharia Biomédica 2014*, por el artículo **A Setup to a Novel SSVEP-BCI Stimulation Based on Depth of Field**
3. *Menção Honrosa no evento Workshop Internacional de Engenharia Biomédica 2014*, por el artículo **Desenvolvimento de Ambientes Virtuais para ICC Aplicadas a Reabilitação e Tecnologias Assistivas**.

Artículos sometidos en revistas

1. **CASTILLO-GARCIA, J.**, CAICEDO, E., BASTOS, T. Adaptive Spontaneous Brain-Computer Interfaces Based on Software Agents. Journal of NeuroEngineering and Rehabilitation., Agosto 2014.
2. COTRINA-ATENCIO A.; BENEVIDES, A.; **CASTILLO-GARCIA, J.**; FERREIRA, A.; BASTOS, T. Statistical Evaluation of a Novel SSVEP-BCI Stimulation Setup Based on Depth-of-Field. Revista Brasileira de Engenharia Biomédica, Marzo 2015.

5. CONCLUSIONES

3. GOULART, C.; VALADÃO, C.; **CASTILLO-GARCIA, J.**, TRAURNICHT M.; CALDEIRA E.; BASTOS T. Brain signals evaluation of children with ASD and their interaction with a mobile robot. Journal of Autism and Developmental Disorders, Agosto 2015.

Bibliografía

- [1] S. Amiri, R. Fazel-Rezai, and V. Asadpour. A review of hybrid brain-computer interface systems. *Adv. in Hum.-Comp. Int.*, 2013:1:1–1:1, January 2013. URL: <http://dx.doi.org/10.1155/2013/187024>, doi:10.1155/2013/187024. 3
- [2] C. Anderson and Z. Sijercic. Classification of EEG signals from four subjects during five mental tasks. *Int. Conf. on Eng. Appl. of Neural Net. (EANN 96)*, 1996. 32, 63
- [3] X. Artusi, I. Niazi, and M. and Farina D. Lucas. Performance Of A Simulated Adaptive Bci Based On Experimental Classification Of Movement-Related. *IEEE Journal on Emerging and Selected Topics In Circuits and Systems.*, 1(4):480–488, 2011. 10, 12
- [4] J. Azorin. Interfaz cerebral no invasiva para control de un sistema domótico por personas discapacitadas. volume 20, pages 249–254, 2007. URL: www.fundacionmapfre.com. 1
- [5] D. Balakrishnan and S. Puthusserypady. Multilayer perceptrons for the classification of brain computer interface data. In *Bioengineering Conference, 2005. Proceedings of the IEEE 31st Annual Northeast*, pages 118–119, April 2005. doi:10.1109/NEBC.2005.1431953. 33
- [6] P. Bartlett. The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the

BIBLIOGRAFÍA

- network. *IEEE Transactions on Information Theory*, 44(2):525–536, 1998. 33
- [7] R. Bellman. *Adaptive control processes - A guided tour*. Princeton University Press, Princeton, New Jersey, U.S.A., 1961. 25
- [8] A. Benevides, T. Bastos, and M. Sarcinelli. Pseudo-Online Classification of Mental Tasks Using Kullback-Leibler Symmetric Divergence. *Journal of Medical and Biological Engineering.*, 32(6):411–416, 2012. 29
- [9] K. Bennett and C. Campbell. Support vector machines: Hype or hallelujah? *SIGKDD Explor. Newsl.*, 2(2):1–13, December 2000. URL: <http://doi.acm.org/10.1145/380995.380999>, doi:10.1145/380995.380999. 36
- [10] O. Bertrand, F. Perrin, and J. Pernier. A theoretical justification of the average reference in topographic evoked potential studies. *Electroencephalogr Clin Neurophysiol*, 62(6):462–464, 1985. 18
- [11] C. Blakely. A fast empirical mode decomposition technique for nonstationary nonlinear time series. In *Center for Scientific Computation and Mathematical Modeling, University of Maryland, College Park MD 20740 USA*, 2005. 10
- [12] B. Blankertz, C. Schäfer, G. Dornhege, and G. Curio. Single trial detection of eeg error potentials: A tool for increasing bci transmission rates. *Proceedings of the International Conference on Artificial Neural Networks*, 2415(2):1137–1143, 2002. 30
- [13] Benjamin Blankertz, Gabriel Curio, and Klaus-Robert Müller. Classifying single trial eeg: Towards brain computer interfacing, 2002. 30, 36
- [14] J. Bradshaw. Software agents. *Cambridge, MA: AAAI Press/MIT Press Chapter 1*, 1997. 38

- [15] A. Buttfeld, P. Ferrez, and J. Millán. Towards A Robust Bci: Error Potentials and Online Learning. *IEEE Transactions on Neural Systems and Rehabilitation Engineering.*, 14(2):411–416, 2006. 10, 11
- [16] G. Carpenter, S. Grossberg, and J. Reynolds. *ARTMAP: A Self-Organizing Neural Network architecture for fast supervised learning and pattern recognition in: Artificial Neural Networks*. Elsevier Science Publishers B.V. (North-Holland), 1991. 33
- [17] G. Carpenter, S. Grossberg, and J. Reynolds. Artmap: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks*, 4:565–588, 1991. 35
- [18] J. Castillo-Garcia, E. Hortal, T. Basto, E. Caicedo, and J. Azorin. Feature selection based on silhouette’s width for spontaneous brain computer interface. *International Workshop on Assistive Technology*, pages 1–4, 2015. 27
- [19] F. Cheein, R. Carelli, W. Cardoso, T. Bastos, and F. di Sciasco. Maps managing interface design for a mobile robot navigation governed by a bci. *16th Argentine Bioengineering Congress and the 5th Conference of Clinical Engineering. Journal of Physics: Conference Series 90*, 2007. 1
- [20] S. Chiappa and S. Bengio. Hmm and iohmm modeling of eeg rhythms for asynchronous bci systems. *In European Symposium on Artificial Neural Networks ESANN*, 2004. 20, 32
- [21] G. Coelho, C. Barbante, L. Boccato, R. Atuxx, J. Oliviera, and F. Von-zuben. Automatic Feature Selection for BCI: an Analysis using the Davies-Bouldin Index and Extreme Learning Machines. *IEEE Congress on Computational Intelligence*, 2012. 33
- [22] J. Cohen. A coefficient of agreement for nominal scales. *Educ Psychol Meas.*, 20:37–46, 1960. 41, 51

BIBLIOGRAFÍA

- [23] M. Congedo, F. Lotte, and A. Lecuyer. Classsification of movement intention by spatially altered electromagnetic inverse solutions. *Physics in Medicine and Biology*, 51:1971–1989, 2006. 20, 33
- [24] I. Daubechies. Ten lectures on wavelets. *CBMS-NSF Regional Conference Series in Applied Mathematics*, SIAM., page 357, 1992. 22
- [25] P. Diez, V. Mut, E. Laciár, A. Torres, and E. Avila. Application of the empirical mode decomposition to the extraction of features from EEG signals for mental task classification. In *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, page 2579–2582, Sept 2009. doi:10.1109/IEMBS.2009.5335278. 3, 10, 11, 62
- [26] G. Dornhege, J. Millán, T. Hinterberger, D. McFarland, and K. Müller. Towards brain–computer interfacing. *Cambridge,MA: MIT Press*, 2007. 9
- [27] R. Duda, P. Hart, and D. Stork. Pattern Classification. *Wiley 2nd edition*, 2001. 30, 31
- [28] J. Faller, C. Vidaurre, T. Solis, C. Neuper, and R. Scherer. Autocalibration and Recurrent Adaptation: Towards a plug and play online ERD-BCI. *IEEE Transactions on Neural Systems and Rehabilitation Engineering.*, 20(3):313–319, 2012. 10, 12
- [29] M. Florian, K. Lehnertz, D. Peter, and C. Elger. Mean phase coherence as a measure for phase synchronization and its application to the {EEG} of epilepsy patients. *Physica D: Nonlinear Phenomena*, 144(3–4):358–369, 2000. 21
- [30] D. Flotzinger, M. Pregenzer, and G. Pfurtscheller. Feature selection with distinction sensitive learning vector quantisation and genetic algorithms. In *Proc. IEEE Int. Conf. on Neural Networks*, pages 3448–3451. 25, 28

- [31] J. Friedman. On bias, variance, 0/1—loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery*, 1:55–77, 1997. 25
- [32] D. Garrett, D. Peterson, C. Anderson, and M. Thaut. Comparison of linear, nonlinear, and feature selection methods for eeg signal classification. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 11(2), 2003. 30, 63
- [33] A. Giret, E. Insfrán, O. Pastor, and L. Cernuzzi. OO-METHOD para el desarrollo de Sistemas de Agentes. *Departamento de Sistemas Informáticos y Computación Universidad Politécnica de Valencia*, 2000. 37
- [34] N. Haidous and S. Sawilowsky. Robustness and Power of the Kornbrot Rank Difference, Signed Ranks, and Dependent Samples T-test. *American Journal of Applied Mathematics and Statistics*, 1(5):99–102, 2013. URL: <http://pubs.sciepub.com/ajams/1/5/4>. 65
- [35] B. Hamadicharef, Haihong Zhang, Cuntai Guan, Chuanchu Wang, Kok Soon Phua, Keng Peng Tee, and Kai Keng Ang. Learning eeg-based spectral-spatial patterns for attention level measurement. In *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, pages 1465–1468, May 2009. doi:10.1109/ISCAS.2009.5118043. 1, 3
- [36] E. Haselsteiner and G. Pfurtscheller. Using time-dependant neural networks for eeg classification. *IEEE Trans. on Rehabilitation Engineering*, 8:457–463, 2000. 32, 35
- [37] J. Hohne, M. Schreuder, B. Blankertz, and M. Tangermann. two-dimensional auditory p300 speller with predictive text system. In *European Information and Communication Technologies*, 2010. 1
- [38] C. Huang and C. Wang. A ga-based feature selection and parameters optimization for support vector machines. *Expert Systems with applications.*, 31:231–240, 2006. 28

BIBLIOGRAFÍA

- [39] G. Huang, Huang B., Q. Zhu, and C. Siew. Extreme learning machine: Theory and applications. *Neurocomputing*, pages 489–501, 2006. URL: <http://dx.doi.org/10.1016/j.neucom.2005.12.126>. 33
- [40] I. Iturrate, J. Antelis, and J. Minguéz. Synchronous eeg brain-actuated wheelchair with automated navigation. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 2318–2325, May 2009. doi:10.1109/ROBOT.2009.5152580. 1
- [41] I. Iturrate, J.M. Antelis, A. Kubler, and J. Minguéz. A noninvasive brain-actuated wheelchair based on a p300 neurophysiological protocol and automated navigation. *Robotics, IEEE Transactions on*, 25(3):614–627, June 2009. doi:10.1109/TRO.2009.2020347. 1
- [42] A. Jain, R. Duin, and J. Mao. On bias, variance, 0/1—loss, and the curse-of-dimensionality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4–37, 2000. 25, 33, 36
- [43] N. Japkowicz and M. Shah. *Evaluation Learning Algorithms a Classification Perspective*, volume 1. Cambridge University Press, 32 Avenue of the Americas , New York, 1 edition, 2011. 40, 50
- [44] M. Kaper, P. Meinicke, U. Grossekhoefer, and T. Lingner. Bci competition 2003data set iib: support vector machines for the p300 speller paradigm. *IEEE Transactions on Biomedical Engineering*, 51:1073–1076, 2004. 20
- [45] A. Karim, T. Hinterberger, J. Richter, J. Mellinger, N. Neumann, H. Flor, A. Kübler, and N. Birbaumer. Neural internet:web surfing with brain potentials for the completely paralyzed. In *Eurorehabilitation and Neural Repair*, volume 20, 2006. 1
- [46] Z. Keirn and J. Aunon. A new mode of communication between man and his surroundings. *IEEE Trans. Biomed. Eng*, 37(12):1209–1214, 1990. 59

- [47] Dornhege G. Braun M. Blankertz B. Müller K. Curio G. Hagemann K. Bruns A. Schrauf M. Kincses W. Kohlmorgen, J. Improving human performance in a real operating environment through real-time mental workload detection. *Toward Brain-Computer Interfacing.*, page 409–422, 2007. 1, 3
- [48] T. Kohonen. The Self-Organizing Map. In *In: Proceedings of the IEEE.*, volume 789, pages 1464–1480, 1990. 33
- [49] J. Lachaux, E. Rodriguez, J. Martinerie, and F. Varela. Measuring phase synchrony in brain signals. *Human Brain Mapping*, 8:194–208, 1999. 22
- [50] E. Llaneza, J. Azorin, and E. Fernández. Control de un brazo robot mediante señales eeg y eeg. In *Seminario de la Red RETADIM sobre BCIs*, 2008. 1
- [51] A. Llera, M. Van-Gerven, V. Gómez, O. Jensen, and H. Kappen. On the use of interaction error potentials for adaptive brain computer interfaces. *Neural Networks*, 24:1120–1127, 2011. 10, 12
- [52] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi. A review of classification algorithms for eeg-based brain–computer interfaces. *Journal of Neural Engineering*, 4(2):R1, 2007. 2, 3, 32
- [53] O. Ludwig and U. Nunes. Novel Maximum-Margin Training Algorithms for Supervised Neural Networks. *IEEE Transactions on Neural Networks*, 21:972–984, 2010. 28, 40
- [54] J. Millán. Brain-controlled robots. *IEEE intelligent System*, 23:74–76, 2008. 1
- [55] J. Millán, A. Buttfield, C. Vidaurre, R. Cabeza, A. Schlögl, G. Pfurtscheller, P. Shenoy, P. Rao, and B. Blankertz. Adaptation in brain-computer interfaces. *In toward Brain Computer Interfacing.*, pages 303–326, 2007. 10, 12

BIBLIOGRAFÍA

- [56] J. Millán, J. Mourino, F. Cincotti, M. Varsta, J. Heikkonen, F. Topani, M. G. Marciani, K. Kaski, and F. Babiloni. Neural networks for robust classification of mental tasks. *Proc. 22nd Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society*, pages 1380–1382, 2000. 35
- [57] J. Millán and J. Mouriño. Asynchronous bci and local neural classifiers: an overview of the adaptive brain interface project. *Ieee Transactions On Neural Systems And Rehabilitation Engineering*, 11(2), 2003. 10, 20, 35
- [58] J. Millán, F. Renkens, J. Mourino, and W. Gerstner. Noninvasive brain-actuated control of a mobile robot by human EEG. *IEEE Transactions on Biomedical Engineering*, 51(6):1026–1033, 2004. URL: <http://dx.doi.org/10.1109/TBME.2004.827086>. 50
- [59] B. Moghaddam. Principal manifolds and probabilistic subspaces for visual recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(6):780–788, 2002. 29
- [60] E. Muglerab, M. Benschc, S. Haldera, W. Rosenstielc, M. Bogdancd, N Birbaumerae, and Kübleraf A. Control of an internet browser using the p300 event-related potential. *International Journal of Bioelectromagnetism*, 10(1):56–63, 2008. 1
- [61] C. Neuper, G. Müller-Putz, R. Scherer, and G. Pfurtscheller. Motor imagery and eeg-based control of spelling devices and neuroprostheses. *Progress in Brain Research*, 159:393–409, 2006. 30
- [62] P. Nunez and K. Pilgreen. The spline-laplacian in clinical neurophysiology: a method to improve eeg spatial resolution. *J Clin Neurophysiol.*, 8(4):397–413, 1991. 19
- [63] H. Nwana. Software agents: An Overview. *The Knowledge Engineering Review*, 11(3), 1996. xiii, 39, 45

- [64] R. Palaniappan. Utilizing gamma band to improve mental task based brain-computer interface design. *IEEE Trans. Neural Syst. Rehabil. Eng.*, 14(3):299–303, 2006. 63
- [65] W. Penny, S. Roberts, E. Curran, and M. Stokes. Eeg-based communication: a pattern recognition approach. *IEEE Transactions on Rehabilitation Engeneering*, 8:214–215, 2000. 20, 35
- [66] G. Pfurtscheller and C. Guger. Brain-computer communication system: Eeg-based control of hand orthosis in a tetraplegic patient. *Acta Chir. Austriaca*, 31:23–25, 1999. 25
- [67] G. Pfurtscheller, C. Neuper, A. Schlögl, and K. Lugger. Separability of eeg signals recorded during right and left motor imagery using adaptive autoregressive parameters. *IEEE Transactions on Rehabilitation Engeneering*, 6, 1998. 20
- [68] R. Picard, E. Vyzas, and J. Healey. Toward machine emotional intelligence: analysis of affective physiological state. *IEEE Transactions on Pattern Analysis and Machine Intelligence.*, 23(10):1175–1191, 2001. 24
- [69] S. Raudys and A. Jain. Small sample size effects in statistical pattern recognition: Recommendations for practitioners. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3):252–264, 1991. 25
- [70] V. Raykar, B. Krishnapuram, J. Bi, M. Dundar, and R. Rao. Bayesian Multiple Instance Learning: Automatic Feature Selection and Inductive Transfer. In *In: Proceedings of the 25th International Conference on Machine Learning. Helsinki, Finland*, pages 1–8., 2008. 40
- [71] V. Razvan. Autonomous artificial intelligent agents. *Center for Cognitive and Neural Studies (CONEURAL)*, 2003. 38
- [72] P. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987. 27

BIBLIOGRAFÍA

- [73] T. Rutkowski, D. Mandic, A. Cichoki, and A. Przybyszewski. EMD Approach to Multichannel EEG Data -The Amplitude and Phase Synchrony Analysis Technique. *D.-S. Huang et al. (Eds.): ICIC 2008, LNCS 5226. Springer-Verlag Berlin Heidelberg.*, pages 122–129, 2008. 23
- [74] W. Samek, C. Vidaurre, K. Müller, and M. Kawanabe. Stationary common spatial patterns for brain–computer interfacing. *Neural Eng* doi:10.1088/1741-2560/9/2/026013, 9, 2012. 10, 13
- [75] S. S. Sawilowsky. New effect size rules of thumb. *Journal of Modern Applied Statistical Methods*, 8(2):597–599, 2009. 65
- [76] G. Schalk, J. Kubanek, K. Miller, N. Anderson, E. Leuthardt, J. Ojemann, D. Limbrick, D. Moran, A. Gerhard, and J. Wolpaw. Decoding two dimensional movement trajectories using electrocorticographic signals in humans. *Journal of Neural Engineering*, 4:264–275, 2007. 1
- [77] P. Shenoy, M. Krauledat, B. Blankertz, P. Rajesh, Rao, and K. Müller. Towards adaptive classification for bci. *Journal of Neural Engineering*, 3, 2006. 10, 11
- [78] P. Somol, J. Novovicova, and P. Pudil. *Pattern Recognition Recent Advances*, chapter Efficient Feature Subset Selection and Subset Size Optimization, pages 75–98. InTech, 2010. 40
- [79] S. Torres, W. Cardoso, T. Bastos, and M. Sarcinelli. Brain-computer interface based on visula evoked potentials to command autonomous robotic wheelchair. *Journal of Medical and Biological Engineering*, pages 407–416, 2010. 1
- [80] L. Trejo, K. Knuth, R. Prado, R. Rosipal, K. Kubitz, R. Kochavi, B. Matthews, and Y. Zhang. EEG-Based Estimation of Mental Fatigue: Convergent Evidence for a Three-State Model. In DylanD. Schmorow

- and LeahM. Reeves, editors, *Foundations of Augmented Cognition*, volume 4565 of *Lecture Notes in Computer Science*, page 201–211. Springer Berlin Heidelberg. doi:10.1007/978-3-540-73216-7_23. 1, 3, 5
- [81] L. Trejo and M. Shensa. Feature extraction of event-related potentials using wavelets: An application to human performance monitoring. *Brain and Language*, 66(19):89–107, 1999. 22
 - [82] D. Valvuená, M. Cyriacks, and A. Graser. Brain-computer interface for high-level control of rehabilitation robotic systems. *Proceedings of the 2007 IEEE 10th international Conference on Rehabilitation Robotics*, pages 619–625, 2007. 1
 - [83] N. Vapnik. *Statistical Learning Theory*. Wiley., 1998. 36
 - [84] C. Vidaurre, A. Schlögl, R. Cabeza, R. Scherer, and G. Pfurtscheller. A fully on-line adaptive bci. *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING*, 53, 2006. 10, 11
 - [85] C. Vidaurre, A. Schlogel, B. Blankertz, M. Kawanabe, and K. Müller. Unsupervised adaption of the lda classifier for brain-computer interfaces. *In Proceedings of the 4th international brain-computer interface workshop and training course 2008*, 53:122–127, 2008. 10, 12
 - [86] C. Vidaurre, A. Schlögl, R. Scherer, and R. Cabeza. Performance Of A Simulated Adaptive Bci Based On Experimental Classification Of Movement-Related. *IEEE Journal On Emerging And Selected Topics In Circuits And Systems*, 4(3):411–416, 2011. 3
 - [87] E. Vyzas and R. Picard. Affective pattern classification. In *Proc. AAAI 1998 Fall Symp., Emotional and Intelligent: The Tangled Knot of Cognition*, Oct 1998. 25
 - [88] Deng J. Wang T. and He B. Classifying eeg-based motor imagery tasks by means of time frequency synthesized spatial patterns. *Clinical Neurophysiology*, 115:2744–2753, 2004. 20

BIBLIOGRAFÍA

- [89] M. Wanli, T. Dat, L. Trung, L. Hong, and Z. Shang-Ming. Using eeg artifacts for bci applications. In *Neural Networks (IJCNN), 2014 International Joint Conference on*, pages 3628–3635, July 2014. doi: 10.1109/IJCNN.2014.6889496. 19