



UNIVERSIDAD DE VALLADOLID

ESCUELA TÉCNICA SUPERIOR  
INGENIEROS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

GRADO EN INGENIERÍA DE  
TECNOLOGÍAS DE TELECOMUNICACIÓN

**Diseño y aplicación de técnicas de análisis de  
electroencefalogramas para sistemas Brain Computer  
Interface basados en potenciales evocados**

Autor:

**Pablo Núñez Novo**

Tutor:

**Dr. D. Roberto Hornero Sánchez**

Valladolid, 14 de Julio de 2014



---

TÍTULO: **Diseño y aplicación de técnicas de análisis de electroencefalogramas para sistemas Brain Computer Interface basados en potenciales evocados**

AUTOR: **Pablo Núñez Novo**

TUTOR: **Dr. D. Roberto Hornero Sánchez**

DEPARTAMENTO: **Teoría de la Señal y Comunicaciones e Ingeniería Telemática**

---

### **Tribunal**

---

PRESIDENTE: **Dr. D. Roberto Hornero Sánchez**

VOCAL: **Dr. D. Jesús Poza Crespo**

SECRETARIA: **Dra. D<sup>a</sup> María García Gadañón**

---

FECHA: **14 de Julio de 2014**

CALIFICACIÓN:

---



# Resumen del TFG

Los sistemas *Brain Computer Interface* (BCI) facilitan canales para enviar mensajes y órdenes al mundo externo sin realizar ninguna actividad muscular. Estos programas pueden habilitar a personas con discapacidades neuromusculares severas como la esclerosis lateral amiotrófica, ataque cerebral y lesiones de la médula espinal para comunicarse y operar programas como procesadores de texto. Para conseguirlo, se usan una serie de señales electrofisiológicas registradas en el cuero cabelludo en las cuáles está codificada la intención del usuario. En este estudio nos centramos en una de ellas: los potenciales evocados P300.

En este trabajo realizamos un estudio de algoritmos de clasificación de características en BCI basados en potenciales evocados P300. Se proponen tres métodos: discriminante lineal de Fisher (FLDA), *Spectral Regression Kernel Discriminant Analysis* (SRKDA) y un conjunto de máquinas de vectores de soporte (SVM). Estos algoritmos se prueban sobre el conjunto de datos II de la III Competición BCI proporcionados por el centro Wadsworth (*NYS Department of Health*). Los resultados se comparan con los del algoritmo ganador, el cual consistía en un conjunto de 17 SVM con selección de canales y con otros métodos del estado del arte que usaron la misma base de datos.

Los resultados muestran que dos de los métodos propuestos (FLDA y SRKDA) obtienen una precisión (97 %) mayor que la del método ganador (96.5 %). Comparados con otros métodos del estado del arte estos dos algoritmos obtienen unos resultados excelentes que los colocan en segundo y tercer lugar de la clasificación global.

## Palabras clave

*Brain Computer Interface* (BCI), electroencefalograma (EEG), potenciales evocados P300, conjunto de datos II de la III Competición BCI, discriminante lineal de Fisher (FLDA), *Spectral Regression Kernel Discriminant Analysis* (SRKDA), máquinas de vectores de soporte (SVM).



# Abstract

Brain Computer Interface (BCI) systems provide channels for sending messages and commands to the external world without any muscular activity. These programs may enable people with severe neuromuscular disorders, such as amyotrophic lateral sclerosis, brainstem stroke and spinal cord injury to communicate and operate programs such as word processors. To achieve this, a variety of electrophysiological signals recorded from the scalp in which the user's intentions are coded are used. In this study we focus on one of these: P300 evoked potentials.

This paper conducts a study on feature classification algorithms in P300-based BCIs. Three different approaches, Fisher's Linear Discriminant (FLDA), Spectral Regression Kernel Discriminant Analysis (SRKDA) and an ensemble of Support Vector Machines (SVM) are considered. These algorithms are tested on dataset II of the III BCI Competition (2003) provided by Wadsworth Center (NYS Department of Health). The results are compared with those of the winning algorithm, which consists of an ensemble of 17 SVM with channel selection, as well as and with other state of the art methods which use the same database.

The results show that two of our proposed methods (FLDA and SRKDA) achieve a higher accuracy (97 %) than the method that won the competition (96.5 %). Compared to other state of the art methods these two algorithms provide excellent results that place them on second and third place of the global classification.

## Keywords

Brain Computer Interface (BCI), electroencephalogram (EEG), P300 evoked potentials, BCI competition III dataset II, feature classification, Fisher's Linear Discriminant (FLDA), Spectral Regression Kernel Discriminant Analysis (SRKDA), Support Vector Machines (SVM).



# Agradecimientos

Al Grupo de Ingeniería Biomédica de la Universidad de Valladolid, en especial a Roberto Hornero por confiar en mí para realizar este proyecto y a Luis Fernando Nicolás, por su inestimable ayuda a lo largo del mismo.



# Índice general

<b>Lista de acrónimos</b>	<b>1</b>
<b>1. Introducción</b>	<b>3</b>
1.1. Señales Biomédicas . . . . .	3
1.2. Electroencefalograma (EEG) . . . . .	3
1.3. Potenciales evocados . . . . .	4
1.4. Brain Computer Interface . . . . .	6
1.5. Objetivos del Trabajo Fin de Grado . . . . .	7
1.6. Estructura del Trabajo Fin de Grado . . . . .	7
<b>2. Sistemas BCI basados en potenciales evocados P300</b>	<b>9</b>
2.1. Introducción a los BCI . . . . .	9
2.2. Paradigmas de control BCI . . . . .	9
2.3. Señales de control en los BCI . . . . .	10
2.3.1. Potenciales corticales lentos (SCP) . . . . .	11
2.3.2. Ritmos sensoriomotores . . . . .	11
2.3.3. Potenciales evocados visuales (VEP) . . . . .	12
2.3.4. Potenciales evocados P300 . . . . .	13
2.4. Procesado de señales BCI . . . . .	14
2.4.1. Adquisición de la señal . . . . .	14
2.4.2. Procesado de la señal . . . . .	14
2.4.3. Aplicación . . . . .	15
2.5. Estado del arte en el campo de los potenciales evocados P300 . . . . .	15
2.5.1. Métodos de adquisición de señal . . . . .	15
2.5.2. Paradigmas de presentación de estímulos . . . . .	16
2.5.3. Algoritmos de extracción y clasificación . . . . .	17
2.5.3.1. III Competición BCI (2004) . . . . .	20
2.5.3.2. Conjunto de datos de la competición . . . . .	22
2.5.4. Aplicaciones . . . . .	24
<b>3. Métodos de clasificación de características</b>	<b>27</b>
3.1. Objetivos . . . . .	27
3.2. Método ganador del conjunto de datos II de la III Competición BCI . . . . .	27
3.2.1. Preprocesado y extracción de características . . . . .	27
3.2.2. Métodos de clasificación . . . . .	28
3.3. Preprocesado y extracción de características aplicados . . . . .	30

3.4. Clasificación propuesta . . . . .	30
3.4.1. Discriminante lineal de Fisher . . . . .	30
3.4.2. Spectral Regression Kernel Discriminant Analysis (SRKDA) . . . . .	33
3.4.3. Conjunto de máquinas de vectores de soporte (SVM) . . . . .	35
3.5. Método de obtención de parámetros óptimos . . . . .	39
3.6. Validación del método propuesto . . . . .	39
<b>4. Resultados</b>	<b>41</b>
4.1. Introducción . . . . .	41
4.2. Parámetros óptimos de los clasificadores . . . . .	41
4.2.1. LDA . . . . .	41
4.2.2. SRKDA . . . . .	41
4.2.3. Conjunto de SVM . . . . .	46
4.3. Resultados de los clasificadores . . . . .	47
4.3.1. LDA . . . . .	48
4.3.2. SRKDA . . . . .	48
4.3.3. Conjunto de SVM . . . . .	48
<b>5. Discusión de los resultados</b>	<b>51</b>
5.1. Introducción . . . . .	51
5.2. Comparación con otros métodos . . . . .	52
5.3. Limitaciones de este estudio . . . . .	54
<b>6. Conclusiones y líneas futuras</b>	<b>55</b>
6.1. Conclusiones . . . . .	55
6.2. Líneas futuras . . . . .	56
<b>Referencias</b>	<b>57</b>
<b>Anexos</b>	<b>63</b>
<b>A. Listado del código desarrollado en Matlab</b>	<b>65</b>
<b>B. Pliego de condiciones</b>	<b>73</b>
<b>C. Presupuesto</b>	<b>79</b>

# Índice de figuras

1.1.	Ejemplo de electrocardiograma (ECG). . . . .	4
1.2.	Esquema de disposición de los electrodos en el sistema internacional 10-20. [34]. . . . .	5
1.3.	Potencial evocado visual (VEP) registrado en el cuero cabelludo occipital [7]. . . . .	6
2.1.	SCP en función de la intención de mover un cursor hacia arriba (negativos) o hacia abajo (positivos) [6]. . . . .	11
2.2.	Ritmo sensoriomotor $\mu$ usado por un sujeto para el desplazamiento de un cursor hacia arriba o hacia abajo [6]. . . . .	12
2.3.	Potencial evocado P300. El asociado al objetivo presenta un pico positivo de voltaje [6]. . . . .	13
2.4.	Esquema de las etapas de tratamiento de una señal en un BCI [14]. . . . .	14
2.5.	Matriz de caracteres mostrada al usuario [34]. . . . .	21
2.6.	Asignación de valores de la variable StimulusCode a diferentes intensificaciones de filas o columnas [34]. . . . .	22
3.1.	Comparativa entre LDA y FLDA. . . . .	32
3.2.	Hiperplano óptimo de separación de dos clases linealmente separables. . . . .	36
3.3.	Tipos de errores en la clasificación. 1) Patrón dentro de la región de separación pero correctamente clasificado. 2) Patrón incorrectamente clasificado [48]. . . . .	37
4.1.	Área bajo la curva ROC para un barrido del parámetro de regularización del LDA. El valor óptimo es de 13001. Sujeto A. . . . .	42
4.2.	Área bajo la curva ROC para un barrido del parámetro de regularización del LDA. El valor óptimo es de 5001. Sujeto B. . . . .	42
4.3.	Curva ROC para el parámetro de regularización óptimo del Sujeto A. LDA. . . . .	43
4.4.	Curva ROC para el parámetro de regularización óptimo del Sujeto B. LDA. . . . .	43
4.5.	Área bajo la curva ROC para un barrido del parámetro $t$ del SRKDA. El valor óptimo es de 1000. Sujeto A. . . . .	44
4.6.	Área bajo la curva ROC para un barrido del parámetro $t$ del SRKDA. El valor óptimo es de 900. Sujeto B. . . . .	44
4.7.	Curva ROC para el parámetro $t$ óptimo del Sujeto A. SRKDA. . . . .	45
4.8.	Curva ROC para el parámetro $t$ óptimo del Sujeto B. SRKDA. . . . .	45

---

4.9. Área bajo la curva ROC para un barrido del parámetro $C$ del conjunto de SVM. El valor óptimo de $C$ es 30. Sujeto A. . . . .	46
4.10. Área bajo la curva ROC para un barrido del parámetro $C$ del conjunto de SVM. El valor óptimo de $C$ es 30. Sujeto B. . . . .	47
4.11. Comparativa porcentaje de aciertos de caracteres en función del método de clasificación y el número de iluminaciones utilizadas para la misma. .	49

# Índice de tablas

2.1.	Resultados para diferentes paradigmas de clasificación y extracción. . . .	20
2.2.	Resultados de los tres primeros puestos de la competición BCI 2003 [50].	23
2.3.	Precisión de algoritmos que usan la base de datos de la III Competición BCI. . . . .	24
4.1.	Resultados para 15 iluminaciones sobre el conjunto de datos de entrenamiento en función del número de SVM. . . . .	47
4.2.	Resultados obtenidos mediante la clasificación usando LDA para distinto número de iluminaciones. . . . .	48
4.3.	Resultados obtenidos mediante la clasificación usando SRKDA para distinto número de iluminaciones. . . . .	48
4.4.	Resultados obtenidos mediante la clasificación usando un conjunto de 19 SVM para distinto número de iluminaciones. . . . .	48
5.1.	Comparación de resultados de 5 y 15 iluminaciones para los distintos métodos de clasificación . . . . .	51
5.2.	Comparación de resultados con el método ganador. . . . .	52
5.3.	Comparación de resultados con otros métodos del estado del arte ordenados por porcentaje de aciertos para 15 iluminaciones. . . . .	53



# Índice de acrónimos

<b>AAR</b>	Adaptive Autoregressive Model
<b>ALS</b>	Amyotrophic Lateral Sclerosis
<b>BOLD</b>	Blood oxygen level dependent
<b>BCI</b>	Brain Computer Interface
<b>CBP</b>	Checkerboard Paradigm
<b>CNN</b>	Convolutional Neural Networks
<b>CSP</b>	Common Spatial Pattern
<b>ECG</b>	Electrocardiograma
<b>EEG</b>	Electroencefalograma
<b>EG</b>	Electrograma
<b>EMG</b>	Electromiograma
<b>ENG</b>	Electroneurograma
<b>EOG</b>	Electrooculograma
<b>ERD</b>	Event-Related Desynchronization
<b>ERG</b>	Electroretinograma
<b>ErrP</b>	Error Evoked Potential
<b>FLDA</b>	Fisher's Linear Discriminant Analysis
<b>fMRI</b>	Functional Magnetic Resonance Imaging
<b>JADE</b>	Joint Approximate Diagonalization of Eigenmatrices
<b>KDA</b>	Kernel Discriminant Analysis
<b>LDA</b>	Linear Discriminant Analysis
<b>LWT</b>	Lifting Wavelet Transform

<b>MEG</b>	Magnetoencefalograma
<b>MICS</b>	Mutual Information Channel Selection
<b>PCA</b>	Principal Component Analysis
<b>ROC</b>	Receiver Operating Characteristic
<b>RCP</b>	Row-Column Paradigm
<b>SCP</b>	Single Cell Paradigm
<b>SCP</b>	Slow Cortical Potential
<b>SMR</b>	Sensorimotor Rhythm
<b>SOBI</b>	Second Order Blind Identification
<b>SRKDA</b>	Spectral Regression Kernel Discriminant Analysis
<b>SSVEP</b>	Steady State Visually Evoked Potential
<b>SVM</b>	Support Vector Machine
<b>VEP</b>	Visually Evoked Potential

# Capítulo 1

## Introducción

### 1.1. Señales Biomédicas

La ingeniería biomédica (IB) es la aplicación de los principios de la tecnología al campo de la medicina y biología. Este campo combina el diseño y las capacidades de resolución de problemas de la ingeniería con las ciencias médicas para avanzar en el tratamiento de enfermedades, incluyendo diagnóstico, monitorización y terapia. Es un campo de estudio relativamente reciente y la investigación y el desarrollo son por tanto de una gran importancia en el mismo.

Las señales biomédicas son observaciones de procesos fisiológicos en organismos, como por ejemplo ritmos neuronales o cardíacos, imágenes de órganos, etc. El procesamiento de estas señales consiste en el análisis de las mismas para obtener información que pueda ayudar a tomar decisiones en diversos campos de la medicina [1].

Estas señales se pueden clasificar a su vez según su descripción matemática en cuatro tipos: deterministas, estocásticas, fractales y caóticas [2], y según su origen en eléctricas, acústicas, mecánicas, biomagnéticas, ópticas, bioquímicas, impedancias e imágenes. Las señales de tipo estocástico son aquellas para las cuales es imposible predecir su valor futuro incluso conociendo valores pasados debido a la presencia de componentes aleatorios. Normalmente las señales biomédicas son de este tipo por diversas razones, tales como el ruido generado por los instrumentos de medición o por la propia naturaleza aleatoria de las mismas.

Una parte importante de las señales biomédicas son las de tipo bioeléctrico, las cuales pueden ser debidas a actividad espontánea o a como respuesta a un estímulo. Entre las que se originan por actividad espontánea destacan el electroencefalograma (EEG), el electrocardiograma (ECG) (Figura 1.1), el electrograma (EG), el electromiograma (EMG), el electroneurograma (ENG), el electrooculograma (EOG) y el electroretinograma (ERG).

En este trabajo nos centraremos en el EEG, que describimos a continuación.

### 1.2. Electroencefalograma (EEG)

El EEG consiste en el registro de la actividad eléctrica en el cuero cabelludo producida por las neuronas del encéfalo mediante la medición de las fluctuaciones de voltaje que

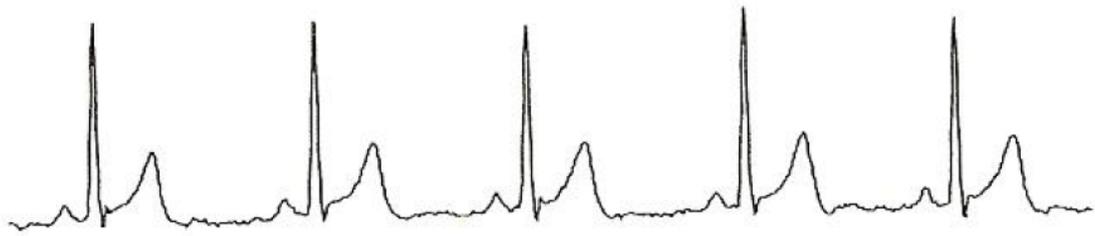


Figura 1.1: Ejemplo de electrocardiograma (ECG).

resultan de las corrientes iónicas entre las neuronas del cerebro [3]. Se usa activamente en la investigación en el campo de la neurociencia. Algunas de sus aplicaciones en el ámbito clínico son la distinción entre ataques epilépticos y otros episodios como síncope o migraña y entre encefalopatías y estados psiquiátricos como la catatonia.

En un EEG convencional, el registro se obtiene colocando una serie de electrodos en el cuero cabelludo mediante un gel conductor para reducir la impedancia. Los nombres y la posición de los electrodos están determinados por el sistema internacional 10-20 [4] para la mayoría de aplicaciones de investigación. Dicho sistema fue desarrollado con el objetivo de estandarizar la reproducibilidad de tal forma que los estudios sobre un sujeto pudieran ser comparados a lo largo del tiempo y con otros sujetos. Cada electrodo tiene una letra para identificar el lóbulo cerebral: Frontal (F), Temporal (T), Parietal (P) y Occipital (O), y un número para identificar el hemisferio. Los nodos de referencia usan la letra A. Los electrodos situados en el hemisferio izquierdo tienen números impares y los situados en el derecho, pares. El número  $z$  se usa para los electrodos localizados en el eje central. La Figura 1.2 muestra el esquema de electrodos del sistema 10-20.

Las señales recogidas por estos electrodos tienen unas amplitudes del orden de los  $\mu\text{V}$  y están comprendidas en el espectro entre 0.5 y 100 Hz.

Los EEG suelen estar afectados por señales eléctricas de origen no cerebral denominadas artefactos. La amplitud de dichos artefactos puede llegar a ser grande en comparación con la amplitud de las señales corticales de interés. Algunos de los artefactos biológicos más comunes son los inducidos por el ojo (parpadeos, movimientos del globo ocular), cardíacos y musculares.

Este trabajo se centra en el estudio de las señales P300, las cuales forman parte del conjunto de potenciales evocados introducidos a continuación.

### 1.3. Potenciales evocados

Los potenciales evocados son potenciales eléctricos registrados del sistema nervioso humano que aparecen tras la presentación de un estímulo, lo que los distingue de los potenciales espontáneos.

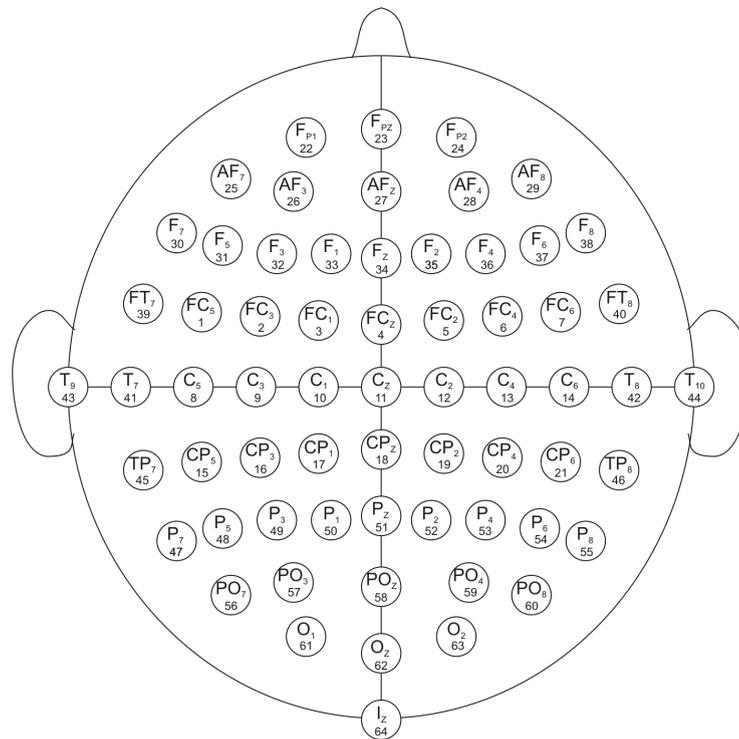


Figura 1.2: Esquema de disposición de los electrodos en el sistema internacional 10-20. [34].

La amplitud de los potenciales evocados suele ser pequeña, en un rango de entre un microvoltio a unos pocos microvoltios. Debido a esta baja amplitud suele ser necesario el promediado de estimulaciones repetidas del mismo estímulo, ya que de esta forma se puede eliminar el ruido, de naturaleza aleatoria y por tanto diferente en cada estimulación [5]. En función del tipo de estímulo que los provoca (véase [6]), algunos de los potenciales evocados se pueden clasificar en:

- Potenciales evocados visuales: cambios en el potencial producidos en la corteza visual tras una estimulación luminosa. Su uso principal es el estudio del nervio óptico.
- Potenciales evocados auditivos: estos potenciales son generados en la cóclea y siguen un camino que pasa por el nervio coclear hasta llegar al córtex cerebral. Son evocados por estímulos auditivos.
- Potenciales evocados somatosensoriales: se usan para comprobar el correcto funcionamiento de la médula espinal. Se registran estimulando nervios periféricos como el tibial, normalmente mediante estímulos eléctricos.

En la Figura 1.3 se puede observar un ejemplo de potencial evocado visual.

La información que aportan los potenciales evocados es utilizada, mediante los sistemas que se introducen a continuación, para el desarrollo y diseño de aplicaciones orientadas a mejorar la vida de las personas discapacitadas.

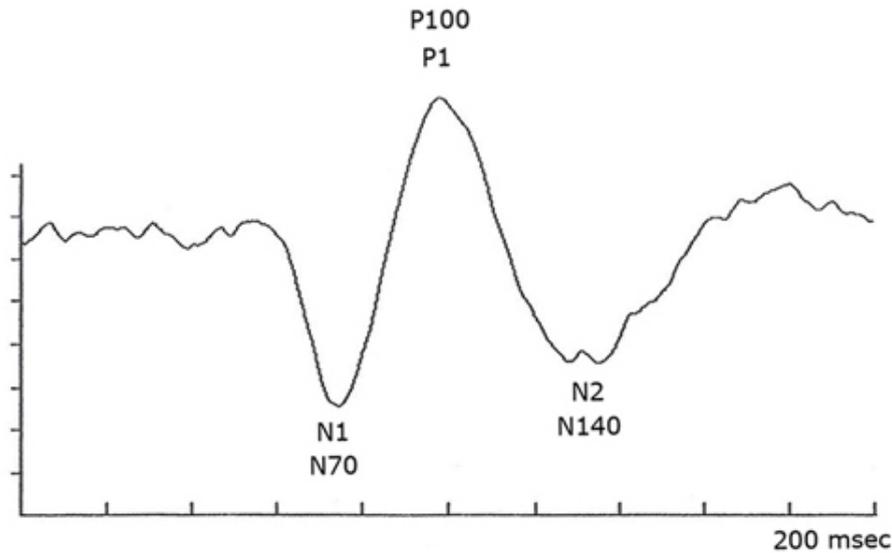


Figura 1.3: Potencial evocado visual (VEP) registrado en el cuero cabelludo occipital [7].

## 1.4. Brain Computer Interface

Brain Computer Interface (BCI) es un sistema de comunicación mediante el cual mensajes u órdenes que un individuo envía al mundo exterior no pasan por los caminos habituales de salida del cerebro de los nervios periféricos y músculos. Se pueden considerar, por tanto, como sistemas que permiten al cerebro interactuar con el medio sin hacer uso de dichos nervios y músculos. Los BCI se pueden clasificar en dos clases: dependientes e independientes [6].

Los BCI dependientes son aquellos que se basan en actividad cerebral eléctrica provocada mientras que los independientes son aquellos que se basan en señales que aparecen de forma espontánea. Un ejemplo de los primeros son los BCI basados en VEP en los que el usuario mira una letra concreta de una matriz y la señal generada depende de la dirección de la mirada. Entre los segundos destacan los BCI basados en potenciales evocados P300, los cuales aparecen independientemente de la intención del usuario.

Los BCI basados en EEG traducen la intención del usuario, la cual se encuentra codificada en la actividad cerebral, en acciones de distinto tipo.

Tras ser registradas, las señales se muestrean y digitalizan. Dichas señales son entonces procesadas en dos etapas: extracción de características y clasificación de características. La extracción consiste en la selección de la información más relevante de las señales para que en la posterior etapa de clasificación sea más sencillo discriminar las mismas en clases mediante métodos estadísticos.

Uno de los principales usos de los BCI es el servir como sistemas de comunicación y control para personas con discapacidades motoras severas como lesiones de la médula espinal o esclerosis lateral amiotrófica. Mediante el BCI estos usuarios pueden expresar sus deseos a los cuidadores o incluso operar sistemas procesadores de texto o sillas de ruedas.

## 1.5. Objetivos del Trabajo Fin de Grado

El objetivo principal de este trabajo es realizar un estudio comparativo de métodos de clasificación de características sobre las señales BCI que constituyen el conjunto de datos de la III Competición BCI (2004) proporcionado por el Centro Wadsworth (*NYS Department of Health*) [8]. El conjunto de datos sobre el que se trabaja está proporcionado por los organizadores de la competición y se explica con detalle en el capítulo 2. Para alcanzar este objetivo general se deben cumplir los siguientes objetivos específicos:

1. Estudiar los sistemas BCI, sus paradigmas de control, señales asociadas y métodos de tratamiento de las mismas, con hincapié en los P300.
2. Realizar un repaso del estado del arte en el campo de los P300 en los ámbitos de adquisición de señales, presentación de estímulos, clasificación y extracción y aplicaciones.
3. Describir y aplicar los tres algoritmos de clasificación usados en este estudio : *Linear Discriminant Analysis* (LDA), *Spectral Regression kernel Discriminant Analysis* (SRKDA) y *Support Vector Machines* (SVM).
4. Obtener los parámetros óptimos de los tres algoritmos de clasificación mediante validación cruzada usando la herramienta Matlab® y usando como criterio el área bajo la curva ROC (Receiver Operating Characteristic).
5. Aplicar dichos métodos a las señales de la competición para obtener la clasificación de las mismas en dos clases: objetivo y no objetivo. La programación y aplicación de los algoritmos se lleva a cabo mediante Matlab®.
6. Usar la clasificación obtenida para determinar los caracteres objetivo que buscaban los usuarios y establecer el porcentaje de aciertos.
7. Realizar una discusión de los resultados y compararlos con los de otros estudios.
8. Extraer conclusiones y determinar posibles líneas futuras de acción.

## 1.6. Estructura del Trabajo Fin de Grado

Este trabajo se divide en 6 capítulos: Introducción, Sistemas BCI basados en potenciales evocados P300, Métodos de clasificación de características, Resultados, Discusión de los resultados y Conclusiones y líneas futuras.

En el capítulo de introducción se establece un primer contacto con las señales biomédicas con especial hincapié en el EEG y se introduce el concepto de potencial evocado. A su vez se define el concepto de BCI, su finalidad y tipos. Finalmente se explican los objetivos y la estructura del trabajo.

El segundo capítulo introduce en profundidad los BCI basados en potenciales evocados P300, así como un pequeño repaso de otros tipos de señales BCI. También se explican las etapas de procesado de las señales. Finalmente se hace un repaso del estado del arte del

campo de la investigación sobre P300, se introducen las competiciones BCI y se explica los objetivos de la misma y la base de datos con la que se trabaja en este proyecto.

En el tercer capítulo se introduce la metodología que se va a seguir en este estudio: primero el preprocesado y la extracción de características usadas por el mismo; después se explican los tres métodos de clasificación utilizados y cómo se obtuvieron los parámetros óptimos de cada uno de ellos.

El siguiente capítulo detalla los resultados obtenidos al aplicar los métodos anteriores a las señales de la competición, con gráficas y tablas de interés.

A continuación, en el capítulo quinto se comparan dichos resultados con los del método ganador y con los de otros métodos actuales que usan la misma base de datos. En el último capítulo se exponen las conclusiones a las que se ha llegado y las limitaciones del estudio. Tras ello se exponen posibles líneas de investigación futuras que podrían abrirse a partir de dichas limitaciones.

## Capítulo 2

# Sistemas BCI basados en potenciales evocados P300

### 2.1. Introducción a los BCI

Un gran número de enfermedades pueden impedir el correcto funcionamiento de los canales neuromusculares por los que el cerebro se comunica y controla su entorno externo, como la esclerosis lateral amiotrófica (ALS), ataques cerebrales, lesiones de la médula espinal, parálisis cerebral y otras enfermedades [6]. Se calcula que estas enfermedades afectan a cerca de dos millones de personas solo en los Estados Unidos [6]. Las personas que estén afectadas de forma más grave pueden perder todo movimiento voluntario de los músculos, incapacitándolos para cualquier tipo de comunicación.

Los sistemas BCI están diseñados para restaurar las funciones sensibles de estas personas, permitiendo que puedan comunicarse e interactuar con el exterior de diversas formas [9].

A lo largo de los últimos 20 años se ha producido un gran incremento en la investigación sobre BCI para comunicación y control. Además un mejor entendimiento del funcionamiento del cerebro, mejores equipos y mayor conocimiento de las necesidades de las personas con discapacidad han ayudado al avance de la misma.

En este capítulo se explicarán los aspectos principales de los sistemas BCI: las señales de control posibles, con hincapié en los potenciales evocados P300, las etapas de procesamiento en los sistemas BCI basados en P300 debido a que son en los que nos centramos en este trabajo y un repaso por el estado del arte en la investigación, incluyendo la descripción de la III Competición BCI (2004) y su base de datos, que se usará en el estudio posterior. Dicho repaso por el estado del arte incluye a su vez apartados dedicados a aplicaciones novedosas de los BCI P300.

### 2.2. Paradigmas de control BCI

Existen dos aproximaciones al control de los sistemas BCI, aunque casi todos los sistemas usan una mezcla de ambas [9]:

- (1) aprender a controlar voluntariamente la actividad cerebral mediante aprendizaje y entrenamiento tras el cual el sujeto controla voluntariamente una actividad cerebral específica y
- (2) aprendizaje de máquina (*Machine Learning*), en el cual el que aprende no es el sujeto sino el propio algoritmo.

El primer paradigma (*Biofeedback Approach*) es un procedimiento que mediante retroalimentación de un parámetro aparentemente autónomo intenta adquirir un control voluntario sobre dicho parámetro. Los sujetos reciben información visual, auditiva o táctil sobre su actividad cardiovascular, temperatura, conductancia de la piel, actividad eléctrica cerebral (EEG), etc. y se les pide que incrementen o disminuyan la actividad de interés, lo que eventualmente lleva a un control de dicha actividad.

El segundo paradigma (*Machine Learning Approach*) consiste en unos algoritmos que se adaptan individualmente al usuario que realiza la tarea. Los algoritmos de aprendizaje requieren ejemplos a partir de los cuales puedan inferir la estructura estadística subyacente de un estado cerebral. Por ello se pide a los sujetos que produzcan un estado cerebral repetidamente en una sesión de calibración.

En la práctica los BCI no van a basarse solamente en uno de estos dos paradigmas, aun así no está claro como unirlos en sincronía. Es importante hacer notar, además, que el 20 % de la población no es capaz de clasificar los patrones de activación cerebral, independientemente de la aproximación al entrenamiento [9].

### 2.3. Señales de control en los BCI

Los BCI actuales difieren en cuanto a cómo se registra la actividad cerebral, cómo se entrena a los sujetos, cómo las señales se transforman en comandos y qué aplicación se provee al usuario. Existen dos métodos principales de registro: invasivos y no invasivos [9].

Los métodos invasivos miden la actividad cerebral en la corteza cerebral o debajo de la misma. Estos métodos tienen muchas ventajas en cuanto a la calidad de las señales y sus dimensiones, pero por otra parte requieren cirugía y tienen problemas potenciales de estabilidad de los implantes e infecciones a largo plazo, por lo que para que se decida por uso estos deben ser sustancialmente mejores que los no invasivos.

La actividad eléctrica cerebral se registra de forma no invasiva mediante el EEG, el magnetoencefalograma (MEG), nivel dependiente de oxígeno en la sangre (blood oxygen level dependent BOLD), o las imágenes de resonancia magnética funcional (fMRI) entre otros.

En un BCI típico los participantes son presentados con estímulos o se les pide que realicen actividades mentales específicas mientras la actividad eléctrica cerebral se registra mediante EEG. Algunas características del EEG son reguladas por el usuario (potenciales corticales lentos o SCP, ritmos sensoriomotores o SMR) mientras que otras aparecen por estimulación visual, táctil o auditiva (potenciales evocados visuales (VEP), potenciales evocados P300). A continuación describimos algunas de estas señales.

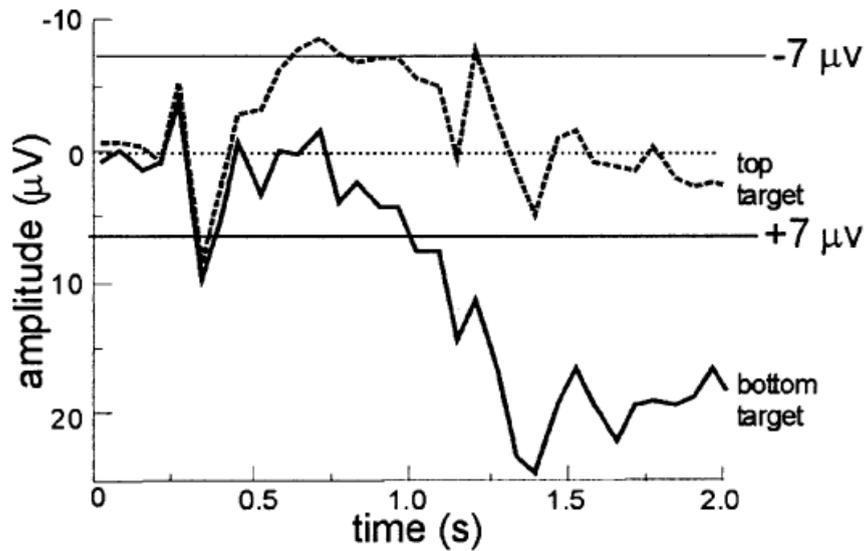


Figura 2.1: SCP en función de la intención de mover un cursor hacia arriba (negativos) o hacia abajo (positivos) [6].

### 2.3.1. Potenciales corticales lentos (SCP)

Los potenciales corticales lentos (SCP) son una de las características del EEG de menor frecuencia. Consisten en cambios de voltaje que ocurren entre 0.5 y 10 segundos. Los negativos se asocian con el movimiento y otras funciones relacionadas con la activación cortical mientras que los positivos están relacionados con activación cortical reducida [6]. En la figura 2.1 se pueden observar SCP en función de la intención de un usuario.

Los SCP se registran por electrodos en el vertex y con referencia en los mastoides. Los SCP se extraen mediante filtrado y se realimentan al usuario mediante una pantalla que muestra una elección en la parte de arriba y una en la de abajo. La selección dura 4 segundos: durante los dos primeros el sistema mide el nivel de voltaje inicial del usuario y durante los dos siguientes el usuario selecciona una de las dos aumentando o disminuyendo el voltaje por una cantidad fijada [6]. El voltaje se muestra como un cursor vertical.

### 2.3.2. Ritmos sensoriomotores

En las personas despiertas, las zonas corticales primarias a menudo presentan actividad cerebral EEG en el rango de los 8-12 Hz cuando no están realizando actividad sensorial ni motora. Esta actividad se conoce como ritmos  $\mu$  cuando se centran en el córtex somatosensorial o motor y ritmos  $\alpha$  cuando se centran en el córtex visual [6].

La actividad  $\mu$  agrupa una serie de ritmos que se distinguen por localización, frecuencia y relación con salidas motoras o entradas sensoriales concurrentes. Estos ritmos  $\mu$  se asocian normalmente con los ritmos  $\beta$  en el rango de 18-26 Hz. Algunos de estos ritmos son armónicos de los  $\beta$ , pero otros son separables.

El interés por estos ritmos radica en el hecho de que disminuyen cuando hay movi-

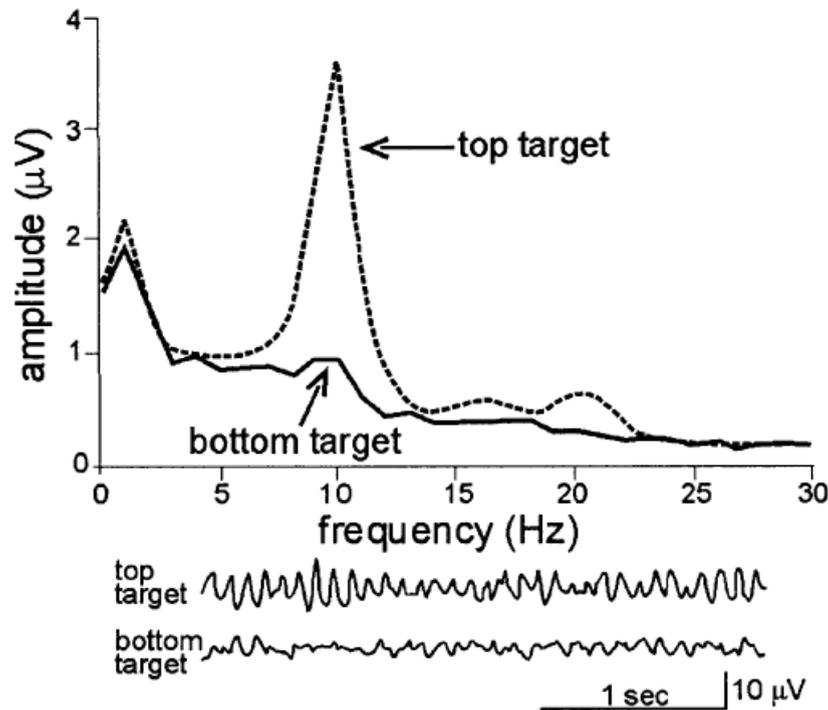


Figura 2.2: Ritmo sensoriomotor  $\mu$  usado por un sujeto para el desplazamiento de un cursor hacia arriba o hacia abajo [6].

miento o se prepara el movimiento: este fenómeno se conoce como “desincronización con el evento relacionado o *event-related desynchronization* (ERD)”. Lo opuesto ocurre después del movimiento o con la relajación. Lo más relevante es, sin embargo, que no es necesario el movimiento en sí para que aparezca un ERD, basta con la imagen motora en el pensamiento. En la figura 2.2 se observa un ejemplo de ritmo sensoriomotor.

### 2.3.3. Potenciales evocados visuales (VEP)

Tras una estimulación visual, se pueden registrar potenciales evocados en el córtex visual en el lóbulo occipital (electrodos 01, 02, 0z en el sistema internacional 10-20 [4]). Un potencial evocado visual se hace estacionario (*steady-state visually evoked potential* o SSVEP) cuando la tasa de los estímulos es superior a 6 Hz [10]. Cuando un sujeto fija su mirada en un objetivo parpadeante, la amplitud del SSVEP aumenta en la frecuencia fundamental del objetivo y sus armónicos de segundo y tercer orden. La resolución en frecuencia de los SSVEP es de 0.2 Hz y el ancho de banda en el que se pueden detectar con fiabilidad es entre 6 y 24 Hz.

Los SSVEP requieren la atención y mirada pero no el entrenamiento del usuario, ya que la respuesta cortical aparece por estimulación visual. Para que aparezcan SSVEP, se presentan objetivos con diferentes frecuencias de iluminación en un monitor, con un número de objetivos que varía entre 4 y 48 [9]. Se suelen obtener precisiones de clasificación del 90 % hacia arriba con tasas binarias de 18.37-27.29 bits/min [11]. La desventaja

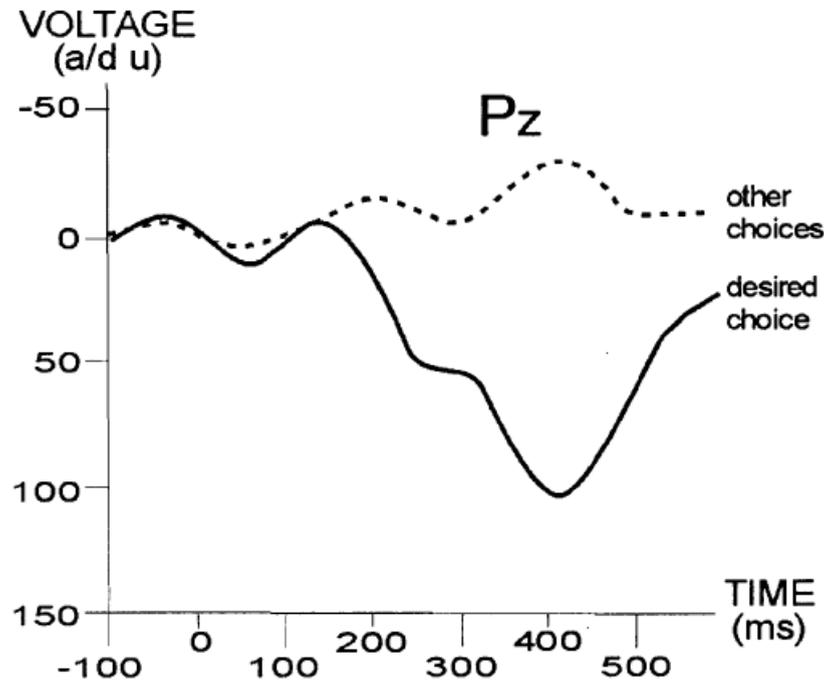


Figura 2.3: Potencial evocado P300. El asociado al objetivo presenta un pico positivo de voltaje [6].

de estos sistemas es que dependen de la capacidad del usuario de mantener la mirada, lo que los hace incompatibles con pacientes con movimiento restringido en el ojo.

#### 2.3.4. Potenciales evocados P300

Los potenciales evocados P300 son picos de voltaje positivos que aparecen en el córtex parietal como respuesta a un estímulo auditivo, visual o somatosensorial infrecuente o significativo a los 300 ms aproximadamente [6]. Aparece normalmente cuando se pide a los usuarios que atiendan a un estímulo infrecuente entre un flujo de estímulos frecuentes [9]. La amplitud del P300 varía en función de características de la tarea como la facilidad de discriminar entre de los estímulos estándar y objetivo, volumen de los tonos en caso de estímulos auditivos, probabilidad del estímulo objetivo, el orden de la secuencia de estímulos y la posición de los electrodos [12]. En la figura 2.3 se muestra un ejemplo de P300.

La aplicación típica que se realiza con este tipo de señales es la selección de letras. Para ello se muestra al usuario una matriz de  $6 \times 6$  letras y números. En un intento se van iluminando aleatoriamente las filas y las columnas de forma que, tras doce iluminaciones, cada fila y columna se ha iluminado una vez. Para el elemento que el usuario quiere seleccionar se habrán producido dos P300, uno al iluminarse la fila en que se encuentra y otro al iluminarse su columna. Se promedia entonces la respuesta para cada elemento de la matriz y el potencial que presente un pico de mayor amplitud será el que el usuario tenía como objetivo [13].



Figura 2.4: Esquema de las etapas de tratamiento de una señal en un BCI [14].

## 2.4. Procesado de señales BCI

Los sistemas BCI constan de 3 etapas a la hora de tratar la señal: adquisición, procesado y aplicación del usuario. A su vez el procesado se divide en dos partes: extracción de características y traducción o clasificación de las mismas. En la figura 2.4 se puede ver este esquema [14]. A continuación se describen estas etapas.

### 2.4.1. Adquisición de la señal

La señal de entrada se registra típicamente mediante un EEG en el cuero cabelludo. La señal es adquirida mediante electrodos según el estándar 10-20 [4], amplificada, filtrada y digitalizada para su uso mediante programas informáticos. Es importante además almacenar las señales de forma digital para su posterior análisis *offline*, para poder establecer qué métodos de extracción y clasificación de características son más adecuados. En nuestro estudio trabajamos con un BCI no invasivo (registro sobre el cuero cabelludo). Es importante amplificar las señales porque su amplitud es muy baja, del orden de los  $\mu\text{V}$ .

### 2.4.2. Procesado de la señal

#### a) Extracción de características

Las señales digitalizadas son sometidas a una serie de procedimientos de extracción de características, como filtrado espacial o análisis espectral. Este análisis extrae las características que, en caso de que el procedimiento sea factible, codifican los mensajes u órdenes del usuario. Los BCI pueden usar señales que están en el dominio del tiempo (amplitud potenciales evocados) o en el de la frecuencia (amplitudes ritmos  $\mu$  y  $\beta$ ).

Como el objetivo es conocer la intención del usuario, debe haber una correlación entre los registros electrofisiológicos y la acción que se pretende realizar. La extracción de características debe maximizar la relación señal a ruido, ya que la señal estará afectada por el ruido del sistema nervioso central.

#### b) Traducción o clasificación de características

En esta etapa se transforman las características extraídas de la señal en señales de control que servirán para controlar un dispositivo. Se lleva a cabo mediante un clasificador

que, en función de cómo sea la característica en cada intento, le asocia una selección u otra [6]. El objetivo es que la selección atribuida sea la misma selección que el usuario pretendía realizar.

Los algoritmos lineales son en general más robustos que los no lineales. Los algoritmos lineales asumen que los datos se pueden separar de manera lineal. Para ello dividen el espacio en hiperplanos de separación que generan dos regiones, cada una de las cuales corresponde a una selección distinta. Los algoritmos no lineales, por otra parte, tienen en cuenta más parámetros por lo que la carga computacional es mayor y no son adecuados para una gran cantidad de datos [49].

Es conveniente aplicar inicialmente una regularización de los datos para limitar la influencia de *outliers* y ruido y reducir la complejidad del clasificador. Por otra parte, si se trata de un algoritmo adaptativo, como en el caso de los usados para clasificar P300, es necesario que se adapte a las características de la señal del usuario. Para ello se entrena a los clasificadores en sesiones de entrenamiento en las que las clases son conocidas. Es posible realizar este ajuste de manera *offline*.

### 2.4.3. Aplicación

En esta etapa, una vez detectadas las características y traducidas en señales de control, se usan las mismas para realizar acciones concretas en aplicaciones. Algunos ejemplos serían seleccionar una letra de la matriz o mover un cursor en una pantalla de arriba abajo.

## 2.5. Estado del arte en el campo de los potenciales evocados P300

En esta sección se hace un repaso del estado actual, así como de los últimos avances en el campo de la investigación y desarrollo de BCI basados en potenciales evocados P300. Asimismo se hace una comparación de resultados entre diferentes algoritmos de clasificación.

Los campos que se estudian serán los métodos de adquisición de señal, paradigmas de presentación de estímulos, aplicaciones y algoritmos de extracción y clasificación, con un especial énfasis en el último campo. Asimismo se debate sobre la III Competición BCI (2004) y se hace un estudio comparativo de algoritmos de clasificación y extracción que hagan uso de la base de datos de la misma.

### 2.5.1. Métodos de adquisición de señal

Tradicionalmente, una de las principales trabas a la hora de realizar un BCI que pueda ser aplicado a situaciones de la vida cotidiana es el engorroso proceso preparatorio, que incluye preparación de la piel, aplicación del gel conductor y posicionamiento de electrodos, entre otras cosas.

Además del método tradicional de adquisición consistente en electrodos y gel aplicado sobre el cuero cabelludo existe actualmente un casco desarrollado por la empresa

EMOTIV que cuenta con la ventaja de que no necesita gel y el consiguiente limpiado de la cabeza del usuario. Además cuenta con electrodos semi-rígidos que permiten un posicionamiento más rápido aunque menos preciso.

Se realizó un estudio comparativo (Mayaud *et al.* [15]) del rendimiento del casco EMOTIV con respecto a tres tipos de electrodos (disco, aguja, y cuadrado) en un mismo equipo y sobre diez usuarios. Los resultados obtenidos sugieren que no hay diferencia en cuanto a rendimiento entre el casco y los electrodos de disco, aunque se encontró una cierta tendencia a una caída de rendimiento en el casco EMOTIV tras dos horas de utilización, además de una mayor incomodidad empezando al mismo tiempo que dicha caída. De cualquier forma, es un avance esperanzador a la hora de establecer sistemas de uso común (no estrictamente de investigación).

### 2.5.2. Paradigmas de presentación de estímulos

Los sistemas BCI-P300 se basan en el paradigma de la aparición de un estímulo esperado entre muchos no esperados. A día de hoy existen tres paradigmas visuales principales: el original de la fila/columna (RCP), donde las filas y las columnas de una matriz visual parpadean en orden aleatorio mientras el usuario espera a que aparezca su elección. Otro es el de la celda única (SCP), donde los elementos de la matriz parpadean de forma aleatoria (con el que se obtienen potenciales más grandes a costa de velocidad y precisión) y finalmente el paradigma del tablero de damas (CBP), donde grupos de elementos de la matriz parpadean de forma cuasi-aleatoria en lugar de filas y columnas.

Se ha propuesto una variación al sistema de la fila/columna (Akram *et al.* [16]) que consiste en integrar un mecanismo de sugerencia de palabras mediante una búsqueda en un diccionario y una selección final de la palabra en función de las sugerencias cuando el número de las mismas se ha reducido a un número determinado. Este sistema reduce el tiempo de entrada de las palabras de forma considerable, si bien solo es útil para aplicaciones de escritura.

Un estudio realizado previamente (Taha Ahi *et al.* [17]) sugiere un posicionamiento alternativo del alfabeto en la matriz bajo el supuesto de que la mayoría de los pares objetivo-error están en la misma fila o columna.

El paradigma del tablero de damas ha sido mejorado recientemente por sus creadores (Townsend *et al.* [18]) mediante la propuesta del paradigma de los cinco parpadeos (FFP), con el que se consiguió una tasa de transmisión de información muy superior a la obtenida con el CBP. Básicamente en este paradigma cada elemento de la matriz parpadea cinco veces en lugar de dos como en el CBP en cada secuencia; de esta forma al haber más candidatos es más fácil seleccionar el deseado por el usuario y se requieren menos secuencias.

Además de los paradigmas visuales también existen paradigmas táctiles y auditivos, los cuales se pueden usar con personas cuya discapacidad sea demasiado severa y no puedan ver correctamente. En el campo de los paradigmas auditivos se propuso un sistema (Guo *et al.* [19]) para seleccionar un objetivo entre una secuencia aleatoria de dígitos con el que se obtuvo una precisión del 85 %, mayor que lo que habitualmente se consigue en este tipo de paradigmas.

Una forma novedosa de enviar los estímulos auditivos (Aminaka *et al.* [20]) es a través de las apófisis mastoides de sujeto, las cuales fueron estimuladas por un transductor que emitía señales sinusoidales en tonos que se debían reconocer por patrones.

### 2.5.3. Algoritmos de extracción y clasificación

Uno de los mayores retos a la hora de optimizar el rendimiento de los BCI P300 es mejorar la detección de los potenciales evocados por los estímulos elegidos. Este proceso consiste en la extracción de las características que representan mejor las intenciones del usuario y la clasificación de las mismas en una salida adecuada por un algoritmo.

#### a) Extracción

Recientemente se llevó a cabo un estudio comparativo de tres métodos de extracción de características (Amini *et al.* [21]). A continuación se describen los métodos estudiados:

La “segmentación inteligente” consiste en segmentar en intervalos no uniformes de los datos del EEG adquiridos en los electrodos y calcular la media de cada segmento. La idea detrás de esta estrategia se basa en la suposición de que un submuestreo efectivo debería proporcionar una distinción clara entre clases (objetivo y no objetivo).

El CSP (Common Spatial Pattern) es una técnica de filtrado espacial que encuentra las direcciones de discriminación óptima entre dos clases mediante la varianza: maximiza la varianza de una condición y al mismo tiempo minimiza la de la otra. También es posible combinar los dos métodos anteriores (primero CSP y luego reducción de muestras con segmentación inteligente).

El filtro basado en subespacios (Shahriari y Erfanian [22]) es un filtro espacial diseñado para maximizar el ratio entre las señales sincronizadas con el estímulo objetivo y las señales sincronizadas con el estímulo no objetivo. Los resultados fueron adecuados aunque no especialmente destacados (89.53 % de precisión).

La transformada de ondícula o transformada wavelet es una herramienta de análisis de señales que se puede usar para maximizar la precisión del clasificador P300 gracias al conocimiento de la localización espacial y frecuencial de estos potenciales (González *et al.* [28]). Se usó el algoritmo *Lifting Wavelet Transform* (LWT) como método de extracción de características y se obtuvieron resultados similares a otros métodos (un 67.5 %), pero podrían mejorar con futuros refinamientos: por ejemplo se podrían probar otras topologías con pasos y coeficientes de filtro adicionales. En otro estudio se propuso un método de detección basado en la extracción por filtro de mediana (Li *et al.* [26]).

#### b) Clasificación

El Discriminador Linear de Fisher (LDA) es un mecanismo de clasificación cuyo propósito es crear un vector discriminante que separe dos clases lo mejor posible.

Usando un LDA típico como clasificador en el estudio de Amini *et al.* [21] se llegó a la conclusión de que la segmentación inteligente producía los mejores resultados (95.05 %

de precisión). La combinación de las mejores características de los métodos anteriores dio como resultado una precisión del 97.4 %.

Por otra parte, un estudio sobre el LDA usado en conjunto con tres mecanismos de extracción de características: *Adaptive Autoregressive Model* (AAR), *Joint Approximate Diagonalization of Eigenmatrices* (JADE) y *Second Order Blind Identification* (SOBI) (Turnip y Widyotriatmo [25]) llegó a la conclusión de que el primero de ellos da mejores resultados llegándose a un 96.16 % de precisión.

Un método novedoso de clasificación de P300 es el de las redes neuronales convolucionales (CNN) (Cecotti y Gräser [24]), la topología de la cual se adapta a la detección de potenciales P300 en el dominio del tiempo. Las CNN son redes neuronales (modelos computacionales basados en el sistema nervioso central de animales) normalmente usadas para reconocimiento de imágenes. Consisten en múltiples capas de pequeñas colecciones de neuronas que miran a pequeñas porciones de la imagen de entrada, las cuales luego se superponen para obtener una representación mejor de la imagen original, de esta forma toleran traslaciones de la imagen. El hecho de que estas redes se usen para tareas de clasificación las hace ideales para los P300, especialmente porque las señales EEG son muy variables en el tiempo y respecto a las personas. Los resultados obtenidos fueron buenos para un número pequeño de electrodos (el 95.5 %, algo inferior a la precisión del método ganador de la tercera competición BCI).

Mediante un LDA Bayesiano, una extensión del LDA de Fisher que efectúa una regresión de forma Bayesiana (Li *et al.* [26]), que es computacionalmente eficiente, obtuvo una precisión media del 90 %.

Recientemente están cobrando una especial importancia los sistemas de basados en información mutua (MICS) (Shahriari y Erfanian [29]). Se propuso un método de selección de canal que hace uso de MICS, seleccionando de forma iterativa un canal con la máxima dependencia respecto a la clase objetivo y mínima dependencia con los canales seleccionados previamente. De esta forma se alcanzó una precisión media del 96.9 %.

Un algoritmo que combina Wavelets para extracción de características y una combinación de FLD combinados (Salvaris *et al.* [37]) dio resultados parecidos a los de los ganadores de la III Competición BCI (95 %) pero con un coste computacional muy inferior para la selección de canal.

Los intentos de optimizar el sistema de deletreador P300 en general no tienen en cuenta los conocimientos existentes sobre el lenguaje, es decir, tratan las selecciones de caracteres como elementos independientes elegidos de un grupo sin información previa. Sin embargo, se puede usar la información sobre el lenguaje (Speier *et al.* [27]) para crear suposiciones previas sobre los caracteres a elegir. Modelando el deletreador P300 como un modelo oculto de Markov o HMM (proceso de Markov de parámetros desconocidos que se deducen a partir de los observables) se supone el deletreado como un proceso secuencial donde la selección de cada carácter (cada estado) está influenciada por las anteriores selecciones, y se deduce la secuencia óptima de caracteres (los estados no visibles) usando el algoritmo de Viterbi. Los resultados obtenidos mejoraron notablemente el rendimiento del sistema de comunicación en cuanto la velocidad de introducción de caracteres y la ITR (32.33), aunque el porcentaje de precisión fue del 88.34 %.

Los potenciales evocados por errores (ErrP) en un EEG son generados por la percepción del sujeto de un error. Se estudió la posibilidad de usar estos potenciales evocados para mejorar el rendimiento del deletreador P300 (Combaz *et al.* [23]) mediante la eliminación de falsos positivos: la teoría es que cuando el usuario se diera cuenta de que el sistema no ha reconocido correctamente el carácter deseado aparecería un ErrP, lo cual se usaría para cancelar la selección. La desventaja es que se requiere un largo entrenamiento para que este sistema funcione correctamente y el porcentaje de acierto (90.55 %) no es especialmente destacable.

Se ha llegado a la conclusión (Mirghasemi *et al.* [36]) de que el preprocesamiento de datos, la extracción de características y la clasificación no deberían ser procesos aislados. En este estudio se demostró que para cada algoritmo de clasificación existe otro algoritmo de preprocesamiento adecuado, llegándose a una precisión del 96.77 % con una combinación de filtrado paso banda, Principal Component Analysis (PCA) y FLD.

En la tabla 2.1 se presenta una comparativa de resultados.

Artículo	Nº	Extracción	Clasificación	Base de Datos	Precisión
Amini <i>et al.</i> (2012) [21]	2	Segmentación inteligente	LDA de Fisher	Competición BCI III	95.05 %
Amini <i>et al.</i> (2012) [21]	2	Common Spatial Patern (CSP)	LDA de Fisher	Competición BCI III	82.45 %
Amini <i>et al.</i> (2012) [21]	2	Segmentación inteligente + CSP	LDA de Fisher	Competición BCI III	85.20 %
Amini <i>et al.</i> (2012) [21]	2	Combinación de características anteriores	LDA de Fisher	Competición BCI III	97.40 %
Shahriari y Erfanian (2013) [22]	5	Filtros basados en subespacios	SVM	Propia	89.53 %
Combaz <i>et al.</i> (2011) [23]	9	Filtro Butterworth y submuestreo	SVM	Propia	90.55 %
Cecotti y Gräser (2011) [24]	2	Submuestreo, filtrado y normalización	Redes neuronales convolucionales	Competición BCI III	95.5 %
Turnip y Widyo-triatmo (2013) [25]	8	SOBI	LDA de Fisher	Propia	93.36 %
Turnip y Widyo-triatmo (2013) [25]	8	JADE	LDA de Fisher	Propia	94 %

Artículo	Nº	Extracción	Clasificación	Base de Datos	Precisión
Turnip y Widyotriatmo (2013) [25]	8	AAR	LDA de Fisher	Propia	96.16 %
Li <i>et al.</i> (2012) [26]	2	Filtro de mediana	LDA Bayesiano	Competición BCI III	90 %
Speier <i>et al.</i> (2013) [27]	15	No especificado	Modelo oculto de Markov	Propia	88.34 %
González <i>et al.</i> (2013) [28]	2	Lifting Wavelet Transform	SVM + Particle Swarm Optimization	Competición BCI III	67.50 %
Shahriari y Erfanian (2011) [29]	2	Información mutua	SVM con kernel lineal	Competición BCI III	96.90 %
Mirghasemi <i>et al.</i> (2006) [36]	2	Filtrado paso banda	PCA + FLD	Competición BCI III	96.77 %
Salvaris <i>et al.</i> (2009) [37]	2	Transformada Wavelet	Conjunto de FLDs	Competición BCI III	95 %
Sakamoto y Aono (2009) [38]	2	Supervised Adaptive Downsampling	FDA	Competición BCI III	96.50 %
Onishi <i>et al.</i> (2012) [39]	2	Tensorización mediante expansión de datos	LDA	Competición BCI III	92 %
Çelik y Anca (2010) [40]	2	Aproximación multiresolución	FLD	Competición BCI III	94.50 %
El Dabbagh y Fakhr (2011) [41]	2	Filtro Chebychev orden 8	SVM con pesos	Competición BCI III	97 %
Tomioka y Haufe (2008) [42]	2	Filtrado paso bajo	Regularización canales con norma euclídea columnas	Competición BCI III	96.50 %

Tabla 2.1: Resultados para diferentes paradigmas de clasificación y extracción.

### 2.5.3.1. III Competición BCI (2004)

Las competiciones BCI son eventos periódicos realizados desde 2001 cuyo objetivo es validar métodos de procesamiento de señal y de clasificación para BCI. Se provee a los participantes de una serie de sets de datos registradas de usuarios en sesiones de entrenamiento



Figura 2.5: Matriz de caracteres mostrada al usuario [34].

y test, cada uno de los cuales tiene objetivos específicos.

En el caso que nos interesa (conjunto de datos II de la III Competición BCI, 2004), las señales de dicho conjunto eran un registro completo de potenciales evocados P300 obtenidos mediante el sistema BCI2000 para un total de dos sujetos. El objetivo era predecir el carácter correcto que el usuario deseaba seleccionar entre los del tablero tradicional bajo el paradigma de la fila/columna de Farwell y Donchin [35]. Este algoritmo consiste en que al usuario se le presenta una matriz  $6 \times 6$  de caracteres (ver Figura 2.5). El usuario debía fijarse en los caracteres de una palabra fijada por los investigadores. Todas las filas y columnas de la matriz se iluminaron de forma sucesiva y aleatoria con una frecuencia de 5.7 Hz. Dos de cada 12 intensificaciones de filas o columnas contenían el carácter deseado; en estos casos las respuestas evocadas por los estímulos son diferentes de las que no los contienen.

Estas señales fueron tratadas de la siguiente forma: se digitalizaron con una frecuencia de muestreo de 240 Hz y a continuación se les aplicó un filtrado paso banda de 0.1-60 Hz. Los canales son los habituales del sistema internacional 10-20 [4]. Para la captura se usó el software BCI2000 [34]. Cada sujeto realizó 5 sesiones, consistentes en diferentes secuencias. En cada secuencia el sujeto centraba su atención en una serie de caracteres.

El display evolucionaba de la siguiente forma: primero se mostraba la matriz (Figura 2.5) durante 2.5 segundos (en este tiempo todos los caracteres tenían la misma intensidad). Tras ello, cada fila y columna de la matriz se iluminaban de forma aleatoria durante 100 ms (12 estímulos: 6 filas y 6 columnas). Tras cada intensificación la matriz no se iluminaba durante 75 ms. Cada set de 12 intensificaciones se repetía 15 veces para cada selección de carácter, es decir, cada combinación de fila y columna se intensificaba 15 veces haciendo un total de 180 intensificaciones de cada selección de carácter. Al pasar a la siguiente selección de carácter la matriz no se iluminaba durante 2.5 s para informar al usuario de que tenía que centrarse en el siguiente carácter de la palabra.

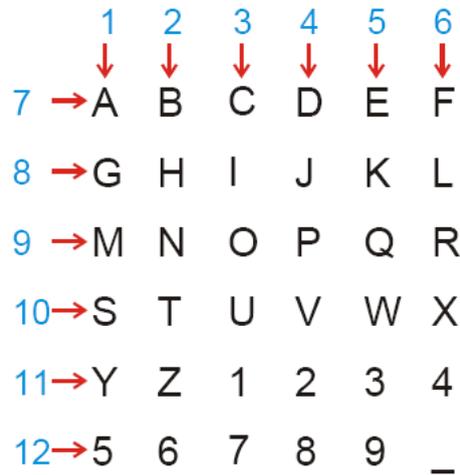


Figura 2.6: Asignación de valores de la variable StimulusCode a diferentes intensificaciones de filas o columnas [34].

### 2.5.3.2. Conjunto de datos de la competición

Para la competición estos datos se convirtieron en 4 ficheros Matlab (\*.mat), un entrenamiento (85 caracteres) y un test (100 caracteres) para cada uno de los dos sujetos A y B. Los datos se guardaron en *precisión simple*. Para cada \*.mat, la señal EEG (obtenida mediante el casco estándar) se organiza en una gran matriz (*Signal*). Es importante saber que para evitar que los participantes en la competición adivinaran las palabras, los caracteres buscados fueron desordenados en los sets. El resto de variables son las siguientes:

**Flashing:** 1 cuando la fila/columna se intensificaba, 0 si no.

**StimulusCode:** 0 cuando no se intensifica ninguna fila/columna.  
 1... 6 para columnas intensificadas de izquierda a derecha.  
 7... 12 para filas intensificadas de arriba abajo (Figura 2.6).

**StimulusType:** 0 cuando ninguna fila/columna se intensifica o la fila/columna intensificada no contiene el carácter deseado.

1 cuando la fila/columna contiene el carácter deseado.

El propósito de esta variable es proveer de un acceso fácil a las etiquetas en los sets de entrenamiento para separar las respuestas que contenían el carácter deseado de las que no.

**TargetChar:** La etiqueta correcta para cada carácter buscado en los datos de entrenamiento.

Las dos últimas variables lógicamente solo están disponibles en las matrices de entrenamiento.

El algoritmo ganador fue el que consiguió un mayor número de aciertos para el conjunto de datos de test. Los tres primeros clasificados se muestran en la tabla 2.2.

<b>Autor</b>	<b>Método</b>	<b>% de aciertos (15 ilumin.)</b>	<b>% de aciertos (5 ilumin.)</b>
1. A. Rakotomamonjy	SVM lineal con selección de canales específicos para cada sujeto	96.50 %	73.50 %
2. L. Yandong	SVM lineal con selección de canales específicos para cada sujeto	90.50 %	55.00 %
3. Z. Zongtan	PCA (Principal Component Analysis)	90.00 %	59.50 %

Tabla 2.2: Resultados de los tres primeros puestos de la competición BCI 2003 [50].

El método ganador fue el propuesto por Rakotomamonjy y Guigue. Dicho método está basado en un conjunto de SVM lineales diferente para cada usuario, usando para construirlos los datos de entrenamiento proporcionados y acompañado de un proceso de selección de canales que se adapta a las diferentes respuestas de cada usuario [8].

En la tabla 2.3 se recogen los resultados, ordenados por porcentaje de aciertos, de una serie de algoritmos que usan el conjunto de datos II de la III competición BCI.

<b>Autor</b>	<b>Método</b>	<b>% de aciertos</b>
1. Amini <i>et al.</i> (2012) [21]	Segmentación inteligente, CSP + LDA de Fisher (Combinación de características)	97.40 %
2. El Dabbagh y Fakhr (2011) [41]	Filtro Chebychev orden 8 + Conjunto de SVMs son pesos	97 %
3. Shahriari y Erfanian (2011) [29]	Información mutua + SVM con kernel lineal	96.90 %
4. Mirghasemi <i>et al.</i> (2006) [36]	Filtrado paso banda + PCA y FLD	96.77 %
5. Rakotomamonjy y Guigue (2008) [8]	Filtrado paso banda + SVM lineal con selección de canales específicos para cada sujeto	96.50 %
6. Sakamoto y Aono (2009) [38]	Supervised Adaptive Downsampling + FDA	96.50 %
7. Tomioka y Haufe (2008) [42]	FPB + Regularización de canales mediante la norma euclídea de las columnas	96.50 %
8. Cecotti y Gräser (2011) [24]	Submuestreo, filtrado y normalización + CNN	95.50 %

	<b>Autor</b>	<b>Método</b>	<b>% de aciertos</b>
9.	Amini <i>et al.</i> (2012) [21]	Segmentación inteligente, LDA de Fisher	95.05 %
10.	Salvaris <i>et al.</i> (2009) [37]	Transformada Wavelet + Conjunto de FLDs	95.00 %
11.	Çelik y Anca (2010) [40]	Aproximación multiresolución + FLD	94.50 %
12.	Onishi <i>et al.</i> (2012) [39]	Tensorización mediante expansión de datos + LDA	92 %
13.	Li <i>et al.</i> (2012) [33]	Filtro de mediana + LDA Bayesiano	90.00 %
14.	Amini <i>et al.</i> (2012) [21]	Segmentación inteligente, CSP + LDA de Fisher	85.20 %
15.	Amini <i>et al.</i> (2012) [21]	CSP + LDA de Fisher	82.45 %

Tabla 2.3: Precisión de algoritmos que usan la base de datos de la III Competición BCI.

#### 2.5.4. Aplicaciones

A continuación se hace un pequeño repaso por algunas aplicaciones recientes de los potenciales evocados P300.

Tradicionalmente la investigación en torno a la aplicación de los P300 se ha centrado exclusivamente en los usuarios con discapacidad motora. Por esta razón resultan interesantes estudios recientes (Rosenfeld *et al.* [30]) en los que se propone usar estos potenciales para otro propósito, como es el de revelar información oculta, bajo el supuesto de que cuando alguien culpable de un crimen vea objetos relacionados con el mismo entre otros irrelevantes se produciría en el sujeto un potencial evocado P300.

Uno de los principales problemas a la hora de controlar vehículos motorizados por BCI basado en P300 es la baja tasa de información y el esfuerzo que implica usarlo de forma continuada. Para evitar estas dificultades (Rebsamen *et al.* [31]) se realizó una silla de ruedas en la cual se selecciona un destino entre una serie de localizaciones prefijadas de forma que ésta se desplaza por caminos virtuales de forma segura, permitiendo al usuario parar en cualquier momento.

En los últimos tiempos están empezando a aparecer sistemas BCI híbridos, como el que combina P300 con potenciales relacionados con desincronizaciones (ERD) (Riechmann *et al.* [32]). Este sistema propone controlar un cursor 2D usando la combinación de ambos potenciales, de una dimensión cada uno, alcanzándose dicho objetivo satisfactoriamente.

Otro sistema BCI híbrido (Li *et al.* [33]) propone combinar P300 con ritmos sensoriomotores para la misma aplicación, de forma que se puedan controlar las dos direcciones del espacio bidimensional de forma simultánea e independiente. Además el sistema permite movimiento en direcciones arbitrarias del cursor, a diferencia del tradicional movimiento discreto, de forma que los P300 controlan los movimientos verticales y los ritmos sensoriomotores determinan la velocidad y dirección del movimiento horizontal.



## Capítulo 3

# Métodos de clasificación de características

### 3.1. Objetivos

El objetivo principal de este trabajo consiste en aplicar diferentes métodos de clasificación de características al conjunto de datos II de la III Competición BCI [34]. Los resultados de esta clasificación se comparan con los obtenidos por el método ganador, propuesto por Rakotomamonjy y Guigue [8], y otros métodos del estado del arte. El objetivo final es superar el porcentaje de clasificación del método ganador y de los métodos con mayor tasa de aciertos en el campo de investigación actual. Al estar centrado nuestro estudio en la clasificación se decidió usar los mismos métodos de preprocesado de señal y extracción de características empleados por dicho método. Por otra parte el tercer método de clasificación que probamos es una implementación simplificada del mismo, por lo que resulta de interés conocerlo. A continuación se describe el método ganador.

### 3.2. Metodología del método ganador del conjunto de datos II de la III Competición BCI

El objetivo de la competición es proporcionar un método que prediga la secuencia de caracteres del conjunto de datos de test de los dos sujetos mediante un algoritmo que puede ser entrenado mediante el conjunto de datos de entrenamiento. El método ganador fue el que obtuvo un mayor porcentaje de aciertos para las 15 iluminaciones de filas y columnas [34].

#### 3.2.1. Preprocesado y extracción de características

En primer lugar, para cada canal, se extraen todas las muestras del conjunto de datos entre 0 y 667 ms tras el inicio de una intensificación. Los potenciales evocados P300 aparecen alrededor de los 300 ms tras la aparición de un estímulo. Por esta razón se considera que esta ventana es suficientemente grande para capturar todas las características temporales para una clasificación eficiente.

Tras ello, cada señal extraída se filtra con un filtro paso banda de Chebyshev de Tipo I y orden 8 con frecuencias de corte de 0.1 y 10 Hz. Finalmente se diezman las señales con un factor de 10, teniendo entonces 14 muestras para cada canal, y se concatenan dichas muestras para los 64 canales. Obtenemos de esta forma una matriz de trabajo de 896 columnas (14 muestras  $\times$  64 canales) y 15300 filas para las señales de entrenamiento (12 columnas/filas  $\times$  15 repeticiones  $\times$  85 caracteres de entrenamiento) cuyas etiquetas son 1 o -1 en función de si la iluminación era del carácter deseado o no.

### 3.2.2. Métodos de clasificación

#### A. Conjunto de máquinas de vectores de soporte (SVM)

Cuando se usa un sistema basado en un conjunto de clasificadores, cada clasificador tiene su propio conjunto de datos de entrenamiento. Es necesario, por tanto, subdividir las 15300 muestras del conjunto de entrenamiento en subconjuntos.

Las particiones usadas se construyen de la siguiente forma: cada carácter se corresponde con 180 señales post-estímulo (12 columnas/filas  $\times$  15 repeticiones). Cada partición de entrenamiento se compone de señales asociadas a 5 caracteres, lo que hace un total de 17 particiones (17  $\times$  5 = 85 caracteres) denominadas  $A_1, A_2, \dots, A_{17}$  para el sujeto  $A$  y  $B_1, B_2, \dots, B_{17}$  para el sujeto  $B$ . Cada partición se compone de 900 señales de entrenamiento de dimensión 896.

En esta estrategia se ha diseñado un sistema de clasificación múltiple para cada sujeto. Cada clasificador individual del sistema es una SVM entrenada en una de las 17 particiones. Se usó SVM debido a que es un buen método de reconocimiento de patrones para problemas de grandes dimensiones. Cada entrenamiento de una SVM incluye un procedimiento de selección de modelo para establecer el parámetro de regularización  $C$ . La función de decisión para una SVM entrenada en la partición  $P_k$  viene dada por la expresión:

$$f_k(\mathbf{x}) = \sum_{i \in P_k} y_i \alpha_i^{(k)} \langle \mathbf{x}, \mathbf{x}_i \rangle + b^{(k)}, \quad (3.1)$$

donde  $\{\alpha_i^{(k)}\}$ ,  $b^{(k)}$  son parámetros obtenidos tras el entrenamiento de las SVM y los  $\mathbf{x}_i$  vectores de post-estímulo con etiquetas  $y_i = \{1, -1\}$ .

#### B. Esquema global de clasificación

Las salidas de estos clasificadores tienen que ser fusionadas de tal forma que se obtenga un solo carácter predicho. De esta forma se transforman los resultados de todos los clasificadores en un solo resultado dentro de las 36 clases posibles (caracteres de la matriz).

El método de clasificación se basa en  $K = 17$  clasificadores que han sido clasificados en uno de los subconjuntos de entrenamiento. Cada clasificador asigna una puntuación  $f_k(\mathbf{x}_{r|c})$ ,  $k = 1, \dots, 17$ , a cada vector de post-estímulo  $\mathbf{x}_{r|c}$  asociado a una fila  $r$  o columna  $c$ .

Después de un número de secuencias  $J$  se considera que la fila o columna más probable es la que maximiza la puntuación:

$$S_{r|c}(\mathbf{x}) = \frac{1}{J} \frac{1}{K} \sum_{j=1}^J \sum_{k=1}^K f_k \left( \mathbf{x}_{r|c}^{(j)} \right), \quad (3.2)$$

donde  $x_{r|c}^{(j)}$  es el post-estímulo asociado a cada columna o fila para la partición  $j$  y  $S_{r|c}$  es la puntuación de cada columna o fila.

Si desarrollamos esta expresión obtenemos:

$$S_{r|c}(\mathbf{x}) = \frac{1}{K} \sum_{k=1}^K \sum_{i \in P_k} y_i \alpha_i^{(k)} \frac{1}{J} \sum_{j=1}^J \left\langle \mathbf{x}_{r|c}^{(j)}, \mathbf{x}_i \right\rangle + b^{(k)}, \quad (3.3)$$

donde se puede observar que se realiza un doble promediado. El primer promedio se calcula en el espacio de los datos. A medida que las secuencias van aumentando se va calculando el promedio de las filas y columnas. El segundo promedio se calcula en el espacio del clasificador.

### C. Procedimiento de selección de canales

Las señales que ofrece la competición en los 64 canales están registradas mediante el gorro estándar del sistema 10-20 [4]. En muchos estudios se emplean únicamente parte de estos canales, aquellos donde se detecta mejor el potencial evocado P300, que normalmente es en la zona centro-parietal.

El método ganador incluye un algoritmo de selección de canales basado en un proceso de reducción de los mismos. En este algoritmo el rendimiento de los clasificadores se evalúa mediante la puntuación:

$$C_{cs} = \frac{t_p}{t_p + f_p + f_n}, \quad (3.4)$$

donde  $t_p$  es el número de verdaderos positivos,  $f_p$  es el número de falsos positivos y  $f_n$  es el número de falsos negativos. El valor de  $C_{cs}$  se evalúa después de cada post-estímulo.

El procedimiento que se sigue es el siguiente: se entrena el SVM lineal con las 896 características proporcionadas por los 64 canales; cada canal se elimina temporalmente, por lo que las 14 muestras correspondientes a ese canal se eliminan y se evalúa el valor de  $C_{cs}$ . Este procedimiento se realiza para los 64 canales. Aquellos canales cuya eliminación proporciona los valores más altos de  $C_{cs}$ , son aquellos que no aportan información relevante al modelo. A continuación, se volvería a proceder de la misma forma pero con un canal menos. Para que el algoritmo sea más rápido, en vez de eliminar canal a canal, se eliminan 4 canales a la vez (aquellos 4 canales cuyo eliminación dé como resultado las 4 mayores puntuaciones de  $C_{cs}$ ). De esta forma, se eliminan primero los canales que proporcionan menos información. La limitación que se le puede encontrar a este procedimiento es que los canales eliminados no vuelven a ser tenidos en cuenta en el mismo.

### 3.3. Preprocesado y extracción de características aplicados

Este estudio parte del mismo procesado previo de las señales que el método ganador, con la diferencia de que en este caso la organización posterior de las señales para su clasificación es diferente.

Las señales de entrenamiento se guardan en sendos ficheros .mat cuyo contenido incluye las siguientes variables:

<i>code</i>	Matriz $85 \times 180$ correspondiente a la fila columna/iluminada en cada muestra de los 85 caracteres.
<i>target</i>	Cadena de 85 elementos en los que se almacena el carácter objetivo del sujeto en cada prueba.
<i>class</i>	Matriz $85 \times 180$ que contiene la clase (1 o -1, que se corresponden con objetivo o no objetivo) que indica si la iluminación correspondía con un carácter objetivo o no.
<i>features</i>	Matriz $85 \times 180 \times 896$ que almacena para cada carácter (85) y cada iluminación (180) las 896 muestras que corresponden a la concatenación de los 64 canales.
<i>name</i>	Cadena con el nombre del sujeto.
<i>matrix</i>	Matriz de caracteres del paradigma usado en la competición.

Las señales de test son similares pero con 100 caracteres en lugar de 85. Las matrices *class* y *features* para este segundo tipo fueron construidas con conocimientos a posteriori de la cadena de caracteres que se buscaba en la competición, con el fin de poder establecer el porcentaje de aciertos tras la clasificación.

### 3.4. Clasificación propuesta

A continuación pasamos a describir a nivel teórico los tres métodos de clasificación usados en este estudio. Los métodos en cuestión son el *discriminante lineal de Fisher* (FLDA), *Spectral Regression Kernel Discriminant Analysis* (SRKDA) y máquinas de vectores de soporte (SVM).

#### 3.4.1. Discriminante lineal de Fisher

Un discriminante es una función que coge un vector de entrada  $\mathbf{x}$  y le asigna a una de  $K$  clases. Los discriminantes lineales son aquellos cuya superficie de decisión es un hiperplano [43]. En nuestro caso el problema se limita a 2 clases y por tanto nos centraremos en el caso  $K = 2$ .

Una forma de ver el problema de clasificación lineal es en términos de la reducción de la dimensionalidad. El FLDA coge un vector de entrada  $x$  de  $D$  dimensiones y lo proyecta en una sola dimensión mediante:

$$y = \mathbf{w}^T \mathbf{x}, \quad (3.5)$$

donde  $\mathbf{w}$  es un vector de pesos. En general, la proyección a una dimensión da lugar a una pérdida considerable de información, y las clases que están muy separadas en el espacio original de  $D$  dimensiones pueden estar muy superpuestas en una sola. De todas formas, ajustando los componentes del vector  $\mathbf{w}$  podemos seleccionar una proyección que maximiza la separación entre clases. Se considera un problema en el cual hay  $N_1$  puntos de la clase  $C_1$  y  $N_2$  puntos de la clase  $C_2$  de tal forma que los vectores media de las dos clases vienen dados por:

$$\mathbf{m}_1 = \frac{1}{N_1} \sum_{n \in C_1} \mathbf{x}_n, \quad \mathbf{m}_2 = \frac{1}{N_2} \sum_{n \in C_2} \mathbf{x}_n. \quad (3.6)$$

La forma más simple de separar las clases cuando se proyectan en  $\mathbf{w}$  es la separación de la media de las clases proyectadas. De esta forma se elige  $\mathbf{w}$  para maximizar:

$$m_2 - m_1 = \mathbf{w}^T (\mathbf{m}_2 - \mathbf{m}_1), \quad (3.7)$$

donde la media de los datos proyectados de la clase  $C_k$  viene dada por:

$$m_k = \mathbf{w}^T \mathbf{m}_k. \quad (3.8)$$

Esta expresión se puede hacer arbitrariamente grande únicamente cambiando la magnitud de  $\mathbf{w}$ . Para solucionar este problema se podría hacer que  $\mathbf{w}$  tuviera longitud unidad, de tal forma que  $\sum_i w_i^2 = 1$ . Usando un multiplicador de Lagrange para resolver el problema de maximización con restricciones se obtiene que  $\mathbf{w} \propto (\mathbf{m}_2 - \mathbf{m}_1)$ . El problema de esta aproximación se puede ver en la figura 3.1. En ella se puede observar que dos clases que están muy separadas en el espacio original de dos dimensiones  $(x_1, x_2)$  tienen una superposición considerable al ser proyectadas sobre la línea que une sus medias. Este efecto aparece por las covarianzas de las distribuciones de las clases, las cuales distan mucho de ser diagonales. La idea del discriminante lineal de Fisher es maximizar una función que dará una gran separación entre las medias de las clases proyectadas a la vez que dar una varianza pequeña dentro de las propias clases, minimizando de esta forma la superposición entre las mismas.

La varianza de las clases de los datos transformados por (3.5) viene dada por:

$$s_k^2 = \sum_{n \in C_k} (y_n - m_k)^2, \quad (3.9)$$

donde  $y_n = \mathbf{w}^T \mathbf{x}_n$ . Se define la varianza total de las clases para el conjunto de datos como  $s_1^2 + s_2^2$ . El criterio de Fisher se define como el cociente entre la varianza entre clases y la varianza total de las clases y viene dado por:

$$J(\mathbf{w}) = \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2}. \quad (3.10)$$

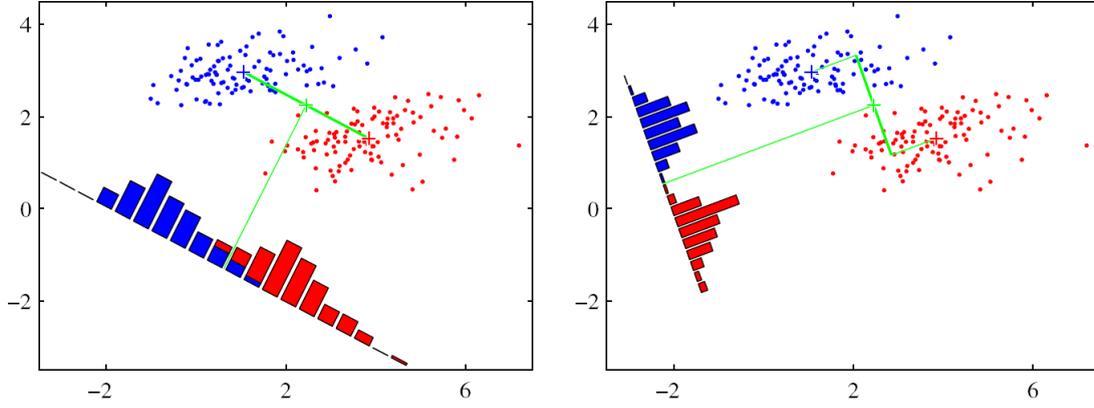


Figura 3.1: En el gráfico de la izquierda se observan las muestras de las dos clases (en rojo y azul) junto con los histogramas que resultan de la proyección sobre la línea que une las medias de las dos clases. Se puede ver una gran superposición entre las clases en el espacio proyectado. El gráfico de la derecha corresponde a la proyección basada en el discriminante lineal de Fisher, con una mejor separación entre clases [43].

Se puede reescribir el criterio de Fisher mediante (3.5), (3.8) y (3.9) de la forma:

$$J(\mathbf{w}) = \frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T S_W \mathbf{w}}, \quad (3.11)$$

donde  $S_B$  es la matriz de covarianza entre clases y viene dada por:

$$S_B = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T \quad (3.12)$$

y  $S_W$  es la matriz de covarianza total de las clases, dada por:

$$S_W = \sum_{n \in C_1} (\mathbf{x}_n - \mathbf{m}_1)(\mathbf{x}_n - \mathbf{m}_1)^T + \sum_{n \in C_2} (\mathbf{x}_n - \mathbf{m}_2)(\mathbf{x}_n - \mathbf{m}_2)^T. \quad (3.13)$$

Derivando (3.11) respecto de  $\mathbf{w}$  se obtiene que  $J(\mathbf{w})$  se maximiza cuando

$$(\mathbf{w}^T S_B \mathbf{w}) S_W \mathbf{w} = (\mathbf{w}^T S_W \mathbf{w}) S_B \mathbf{w}. \quad (3.14)$$

De (3.12) se puede ver que  $S_B \mathbf{w}$  siempre está en la dirección de  $(\mathbf{m}_2 - \mathbf{m}_1)$ . Además, al no importar la magnitud de  $\mathbf{w}$ , solo su dirección, se pueden eliminar los factores de escala  $(\mathbf{w}^T S_B \mathbf{w})$  y  $(\mathbf{w}^T S_W \mathbf{w})$ . Multiplicando ambos lados de (3.14) por  $S_W^{-1}$  se obtiene:

$$\mathbf{w} \propto S_W^{-1} (\mathbf{m}_2 - \mathbf{m}_1). \quad (3.15)$$

Esta ecuación se conoce como el discriminante lineal de Fisher (FLDA), aunque estrictamente no es un discriminante sino una dirección de proyección de los datos a una dimensión. De todas formas los datos proyectados se pueden usar para construir un discriminante eligiendo un umbral  $y_0$  tal que se clasifique un punto como perteneciente a  $C_1$  si  $y(x) \geq y_0$  y a  $C_2$  si no. Una regla habitual es que  $\mathbf{x}$  pertenece a la clase  $i$  si:

$$|\mathbf{w}^T \mathbf{x} - \mathbf{w}^T \mathbf{m}_i| = \min_j |\mathbf{w}^T \mathbf{x} - \mathbf{w}^T \mathbf{m}_j|. \quad (3.16)$$

### 3.4.2. Spectral Regression Kernel Discriminant Analysis (SRKDA)

El algoritmo *kernel discriminant analysis* (KDA) fue propuesto como una extensión del LDA para el caso no lineal. El KDA es un clasificador que, de forma similar al LDA, busca direcciones que mejoran la separación entre clases. La diferencia está en que el KDA considera el problema en el espacio de características  $\mathcal{J}$  inducido por una aplicación no lineal  $\Phi: \mathbb{R}^N \rightarrow \mathcal{J}$ , donde  $N$  es el número de características [45].

La base del KDA es aplicar los vectores de características en un espacio de dimensión alta donde los problemas complejos de clasificación sean más fácilmente separables. La función objetivo del KDA para encontrar las funciones óptimas de proyección  $\mathbf{v}_{opt}$  es la siguiente:

$$\mathbf{v}_{opt} = \operatorname{argm\acute{a}x} \frac{\mathbf{v}^T S_B^\Phi \mathbf{v}}{\mathbf{v}^T S_W^\Phi \mathbf{v}}, \quad (3.17)$$

donde  $\mathbf{v} \in \mathcal{J}$ ,  $S_B^\Phi$  y  $S_W^\Phi$  son las matrices de dispersión o *scatter* (estimador de la matriz de covarianza de la distribución normal multivariante) entre clases en  $\mathcal{J}$ . Es decir:

$$S_B^\Phi = \sum_{k=1}^C M_k \left( \boldsymbol{\mu}_\Phi^{(k)} - \boldsymbol{\mu}_\Phi \right) \left( \boldsymbol{\mu}_\Phi^{(k)} - \boldsymbol{\mu}_\Phi \right)^T, \quad (3.18)$$

$$S_W^\Phi = \sum_{k=1}^C \left[ \sum_{i=1}^{M_k} \left( \Phi(\mathbf{x}_i^{(k)}) - \boldsymbol{\mu}_\Phi^{(k)} \right) \left( \Phi(\mathbf{x}_i^{(k)}) - \boldsymbol{\mu}_\Phi^{(k)} \right)^T \right], \quad (3.19)$$

donde  $C$  es el número de clases,  $\boldsymbol{\mu}_\Phi^{(k)}$  y  $\boldsymbol{\mu}_\Phi$  son los centroides de la clase  $k$  y la media global respectivamente en el espacio de características, y  $M_k$  es el número de vectores de características en la clase  $k$ .

Se puede probar que el problema de maximización planteado se puede resolver eficientemente mediante el truco del núcleo [46]. Para una función de aplicación  $\Phi$  el producto interno  $\langle \cdot, \cdot \rangle$  se puede definir en  $\mathcal{J}$  lo que proporciona el llamado *reproducing kernel Hilbert space* (RKHS)

$$\langle \Phi(\mathbf{x}), \Phi(\mathbf{y}) \rangle = K(\mathbf{x}, \mathbf{y}),$$

donde  $K(\mathbf{x}, \mathbf{y})$  es una función semidefinida positiva del núcleo. Entonces, a partir de la teoría de los núcleos reproducibles, sabemos que cualquier solución  $\mathbf{v}_{opt} \in \mathcal{J}$  tiene que estar dentro del espacio de las muestras de entrenamiento en  $\mathcal{J}$ . Tienen que existir coeficientes  $\alpha_i$  tales que:

$$\mathbf{v}_{opt} = \sum_{i=1}^M \alpha_i \Phi(\mathbf{x}_i), \quad (3.20)$$

donde  $M$  es el número de puntos de datos de entrenamiento.

Sea  $\boldsymbol{\alpha}_{opt} = [\alpha_1, \alpha_2, \dots, \alpha_M]$ , entonces se puede probar que la ecuación (3.17) es equivalente a:

$$\boldsymbol{\alpha}_{opt} = \operatorname{argm\acute{a}x}_{\boldsymbol{\alpha}} \frac{\boldsymbol{\alpha}^T K V K \boldsymbol{\alpha}}{\boldsymbol{\alpha}^T K K \boldsymbol{\alpha}}, \quad (3.21)$$

donde  $K$  es la matriz núcleo  $K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$  y la matriz  $V$  se define como:

$$V_{ij} = \begin{cases} \frac{1}{M_k} & \text{si } \mathbf{x}_i \text{ y } \mathbf{x}_j \text{ están en la misma clase (k).} \\ 0 & \text{si } \mathbf{x}_i \text{ y } \mathbf{x}_j \text{ no están en la misma clase (k).} \end{cases} \quad (3.22)$$

El problema de maximización descrito se corresponde con el siguiente problema de descomposición de autovalores:

$$KVK\alpha = \lambda K\alpha. \quad (3.23)$$

Cada autovector  $\alpha_{opt}$  da la proyección de un nuevo patrón de test  $\hat{\mathbf{x}}$  en  $\mathbf{v}$  en el espacio de características. Para un nuevo vector de datos de ejemplo  $\hat{\mathbf{x}}$  tenemos:

$$\Theta(\hat{\mathbf{x}}, \alpha_{opt}) = \langle \mathbf{v}, \Phi(\hat{\mathbf{x}}) \rangle = \sum_{i=1}^M \alpha_i K(\mathbf{x}_i, \hat{\mathbf{x}}). \quad (3.24)$$

Finalmente,  $\hat{\mathbf{x}}$  es clasificada en base a la distancia euclídea a la proyección de la media de cada clase:

$$\hat{l} = \underset{k}{\operatorname{argmín}} \left\| \langle \mathbf{v}, \Phi(\hat{\mathbf{x}}) \rangle - \langle \mathbf{v}, \boldsymbol{\mu}_{\Phi}^{(k)} \rangle \right\|. \quad (3.25)$$

El algoritmo SRKDA es una mejora de KDA que encuentra la proyección óptima mediante una regresión. La gran ventaja de este algoritmo es que facilita una computación más eficiente ya que no se necesita computación de autovectores implicada en la resolución de la ecuación (3.21). Para obtener el SRKDA, Cai *et al.* [47] probaron el siguiente resultado:

Sea  $\mathbf{y}$  el autovector del problema de autovalores

$$V\mathbf{y} = \lambda\mathbf{y} \quad (3.26)$$

con autovalor  $\lambda$ . Si  $K\alpha = \mathbf{y}$ , entonces  $\alpha$  es el autovector del problema de autovalores de la ecuación (3.23) con el mismo autovalor  $\lambda$ .

Según este teorema, las funciones de proyección se pueden obtener en dos pasos:

- (1) resolver el problema de autovalores en la ecuación (3.26) para obtener  $\mathbf{y}$ ;
- (2) encontrar  $\alpha_{opt}$  solución de  $K\alpha = \mathbf{y}$ .

La solución al problema de autovalores en el primero de estos pasos se puede encontrar de manera trivial explotando la estructura especial de  $V$ . Sin pérdida de generalidad, se puede asumir que los puntos de los datos de entrenamiento están ordenados de acuerdo a sus etiquetas, con lo que la matriz  $V$  tiene una estructura diagonal por bloques:

$$V = \begin{pmatrix} V^{(1)} & 0 & \dots & 0 \\ 0 & V^{(2)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & V^{(C)} \end{pmatrix}, \quad (3.27)$$

donde  $V^{(k)}$ ,  $k \in \{1, \dots, C\}$ , es una matriz  $M_k \times M_k$  con todos los elementos iguales a  $1/M_k$ . Se puede observar que los autovalores de  $V^{(k)}$  son 0 y 1 y que solo tiene un autovector  $e^{(k)} = [1, 1, \dots, 1]^T \in \mathbb{R}^{M_k}$ , asociado con el autovalor 1. Dada la estructura diagonal por bloques de  $V$ , los autovalores y autovectores son la unión de los autovalores y autovectores de sus bloques (rellenando apropiadamente con ceros). De esta forma  $V$  tiene  $C$  autovectores de  $V$  con el mismo autovalor 1. Estos autovectores son:

$$\bar{\mathbf{y}}_k = \left[ \underbrace{0, \dots, 0}_{\sum_{i=1}^{k-1} M_i}, \underbrace{1, \dots, 1}_{M_k}, \underbrace{0, \dots, 0}_{\sum_{i=k+1}^C M_i} \right]^T, \quad k = 1, \dots, C. \quad (3.28)$$

Dado que todos estos autovectores están asociados al autovalor 1 de  $V$ , podemos tomar otros vectores ortogonales  $C$  cualesquiera en el espacio generado por  $\{\bar{\mathbf{y}}_k\}_{k=1}^C$  y definirlos como nuestros  $C$  autovectores. El vector de unos está incluido de forma natural en el espacio. Este vector no tiene ninguna utilidad puesto que la correspondiente función proyectiva llevará todas las muestras al mismo punto. Por esa razón este vector se elige como el primer autovector de  $V$  y los restantes autovectores se encuentran mediante el algoritmo de Gram-Schmidt. Tras ello el vector de unos se puede eliminar dejando exactamente  $C - 1$  autovectores de  $V$ ,  $\{\mathbf{y}_k\}_{k=1}^C$ .

En el segundo paso se obtiene un  $\alpha_{opt}$  para cada autovector de  $V$  resolviendo el correspondiente sistema de ecuaciones lineales  $K\alpha = \mathbf{y}$ . La matriz núcleo  $K$  es semidefinida positiva. Cuando  $K$  es singular, el sistema puede no tener solución o infinitas soluciones. Entonces, una manera posible de obtener un estimador aproximado es adoptar la técnica de regularización:  $(K + \delta I)\alpha_{opt,k} = \mathbf{y}_k$ , donde  $I$  es la matriz identidad y  $\delta > 0$  es el parámetro de regularización. Dado que ahora la matriz  $K + \delta I$  es definida positiva, la factorización de Cholesky se puede usar para computar eficientemente la solución. Finalmente, tras los dos pasos, los nuevos patrones son clasificados proyectando el vector de características con las  $C - 1$  funciones proyectivas  $\{\alpha_{opt,k}\}_{k=1}^C$  de la misma forma que el KDA.

### 3.4.3. Conjunto de máquinas de vectores de soporte (SVM)

Las máquinas de vector soporte se suelen emplear tanto para clasificación como para regresión no lineal. La idea principal es construir un hiperplano de separación de tal manera que el margen de separación entre clases sea máximo. A este hiperplano se le denomina hiperplano óptimo [44].

Para la construcción del hiperplano se supone un conjunto de entrenamiento dado por  $\{\mathbf{x}_i, d_i\}_{i=1}^N$ , donde  $\mathbf{x}_i$  representa el valor de entrada y  $d_i$  el correspondiente valor objetivo. La distancia entre un hiperplano y el punto de los datos más cercano a éste se denomina margen de separación y se denota por  $\rho$ . La frontera de decisión viene especificada por la ecuación:

$$\mathbf{w}^T \mathbf{x} - b = 0, \quad (3.29)$$

donde  $\mathbf{x}$  es el vector de entrada,  $\mathbf{w}$  es el vector de pesos ajustable y  $b$  es el sesgo. Si el objetivo conseguir es el hiperplano óptimo, la función discriminante vendrá dada por la siguiente expresión:

$$g(\mathbf{x}) = \mathbf{w}_0^T \mathbf{x} - b_0. \quad (3.30)$$

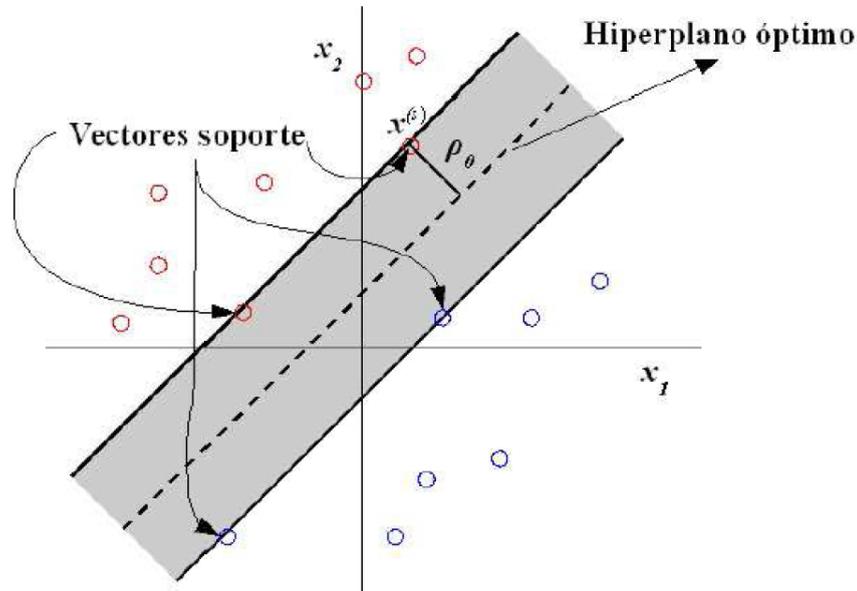


Figura 3.2: Hiperplano óptimo de separación de dos clases linealmente separables. Los círculos rojos representan los patrones de una clase y los azules de la otra [48].

Aquellos puntos  $\{\mathbf{x}_i, d_i\}$  que se encuentran más cercanos al hiperplano óptimo y cuyo margen de separación  $\rho_0$  es máximo, se les denomina vectores soporte y su clasificación es la más complicada. En la figura 3.2 se observa un ejemplo de estos conceptos.

Para la construcción del hiperplano óptimo se suele emplear como método los multiplicadores de Lagrange. El problema de optimización puede ser planteado de la siguiente forma:

*Dado los datos de entrenamiento  $\{\mathbf{x}_i, d_i\}_{i=1}^N$ , encontrar los valores óptimos del vector de pesos  $\mathbf{w}$  y del sesgo  $b$  para que satisfaga el criterio:*

$$d_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \quad \text{para } i = 1, 2, \dots, N \quad (3.31)$$

*y el vector de pesos  $\mathbf{w}$  que minimice la función de coste:*

$$\Phi(\mathbf{w}) = \frac{1}{2} \mathbf{w}^T \mathbf{w}. \quad (3.32)$$

Para resolver este problema se emplea el método de los multiplicadores de Lagrange:

*Dado los datos de entrenamiento  $\{\mathbf{x}_i, d_i\}_{i=1}^N$ , encontrar el multiplicador de Lagrange que maximice la función objetivo:*

$$Q(\boldsymbol{\alpha}) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j d_i d_j \mathbf{x}_i^T \mathbf{x}_j \quad (3.33)$$

*sujeto a las restricciones*

$$\sum_{i=1}^N \alpha_i d_i = 0, \quad (3.34)$$

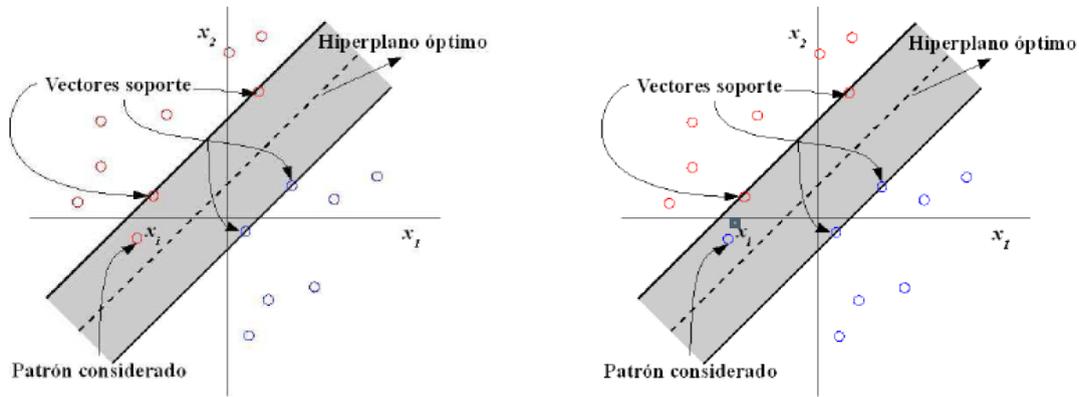


Figura 3.3: Tipos de errores en la clasificación. 1) Patrón dentro de la región de separación pero correctamente clasificado. 2) Patrón incorrectamente clasificado [48].

$$\alpha_i \geq 0 \quad \text{para } i = 1, 2, \dots, N. \quad (3.35)$$

A partir de los multiplicadores de Lagrange óptimos  $\alpha_{0,i}$ ,  $d_i$ ,  $\mathbf{x}_i$  se calculan los parámetros del hiperplano óptimo [44].

$$\mathbf{w}_0 = \sum_{i=1}^N \alpha_{0,i} d_i \mathbf{x}_i, \quad (3.36)$$

$$b_0 = 1 - \mathbf{w}_0^T \mathbf{x}^{(s)} \quad \text{para } d^{(s)} = 1. \quad (3.37)$$

Cuando las clases no se pueden separar linealmente no se puede construir el hiperplano sin que se produzcan errores en la clasificación de algunos de los patrones. Para este caso, el hiperplano óptimo será aquel que minimice la probabilidad de cometer ese error. Se introduce un nuevo parámetro  $\xi_i$  que mide la diferencia desde el patrón considerado al hiperplano óptimo.

El error puede estar definido de dos formas:

- (1) que uno de los datos caiga en la zona de decisión pero en el lado correcto del hiperplano. En este caso la clasificación es correcta y  $0 \leq \xi_i \leq 1$ ;
- (2) que uno de los datos caiga en el lado incorrecto de la frontera de decisión, en cuyo caso  $\xi_i > 1$ .

Se pueden observar ambos errores en la figura 3.3.

El hiperplano óptimo cuando las variables no son linealmente separables se encuentra minimizando la siguiente expresión:

$$\Phi(\mathbf{w}, \xi) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^N \xi_i \quad (3.38)$$

donde  $C$  es un parámetro escalar y positivo determinado de forma experimental que controla la complejidad de la máquina de vector soporte y el número de patrones que se han clasificado incorrectamente.

El problema de optimización se puede resolver empleando, de nuevo, el método de multiplicadores de Lagrange cambiando la segunda de las restricciones (3.35) de la siguiente forma:

$$0 \geq \alpha_i \geq C \quad \text{para } i = 1, 2, \dots, N. \quad (3.39)$$

Si  $C \rightarrow \infty$  estamos ante un problema de clasificación linealmente separable.

Cuando se emplean las máquinas de vector soporte para reconocimiento de patrones el problema varía ligeramente. Como se ha presentado previamente el espacio de entrada se aplica a un espacio de dimensión elevada y una vez en este plano se construye el hiperplano óptimo de separación. Este hiperplano será una combinación lineal de los vectores del espacio de dimensión mayor, no de los del espacio original [44].

El problema de optimización será el siguiente:

*Dado el conjunto de entrenamiento  $\{\mathbf{x}_i, d_i\}_{i=1}^N$ , encontrar el multiplicador de Lagrange  $\{\alpha_i\}_{i=1}^N$  que maximice la función objetivo:*

$$Q(\alpha) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j d_i d_j K(\mathbf{x}_i, \mathbf{x}_j) \quad (3.40)$$

*sujeto a las restricciones*

$$\sum_{i=1}^N \alpha_i d_i = 0, \quad (3.41)$$

$$0 \leq \alpha_i \leq C \quad \text{para } i = 1, 2, \dots, N. \quad (3.42)$$

La ecuación (3.40) difiere de (3.33) en la expresión  $K(\mathbf{x}_i, \mathbf{x}_j)$  que es lo que se conoce como núcleo del producto interno. Existen diferentes tipos de núcleos: polinómicos, gaussianos, exponenciales, etc. En nuestro estudio usamos un núcleo de tipo gaussiano. La expresión que lo define es:

$$K(\mathbf{x}, \mathbf{x}_i) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{2\sigma^2}\right), \quad (3.43)$$

donde  $\sigma$  especifica la anchura de las funciones gaussianas,  $\mathbf{x}_i$  es su centro y  $\mathbf{x}$  es el patrón de entrada. Se suele identificar  $1/(2\sigma^2) = \gamma$ .

Para obtener los resultados de la clasificación mediante un conjunto de SVM entrenados en subconjuntos del conjunto de datos de entrenamiento, de forma totalmente análoga al método ganador.

### 3.5. Método de obtención de parámetros óptimos

Una característica común que tienen todos estos clasificadores es el hecho de que poseen parámetros configurables. Para obtener los valores óptimos de dichos parámetros en nuestro estudio se usó validación cruzada. La validación cruzada es una técnica de validación de modelos para estimar como de preciso será un modelo predictivo en la práctica. El objetivo de la validación cruzada es definir un conjunto de datos para probar el modelo en la fase de entrenamiento y de este modo dar una visión de cómo el modelo actuará con un conjunto de datos independiente.

Una ronda de validación cruzada incluye partir una muestra de datos en subconjuntos complementarios, realizando el análisis en un subconjunto (conjunto de entrenamiento) y validando el análisis en otro subconjunto (conjunto de test). Para reducir la variabilidad se realizan múltiples rondas de validación cruzada usando diferentes particiones y se hace la media de los resultados de la validación.

En una validación cruzada de  $k$  iteraciones, la muestra original se subdivide en  $k$  particiones de igual tamaño. De esos  $k$  subconjuntos, uno se usa como conjunto de validación y los restantes como conjunto de entrenamiento. El proceso se repite  $k$  veces, usando cada uno de los subconjuntos una vez como conjunto de validación. Se realiza la media de los resultados para obtener una estimación. Para este estudio usamos la validación cruzada de 10 iteraciones (*10-fold cross-validation*). Para obtener los parámetros óptimos de los clasificadores realizamos validación cruzada para distintos valores de los mismos con el conjunto de datos de entrenamiento (85 caracteres). Tras el proceso se seleccionan como óptimos aquellos que nos proporcionen mayor área bajo la curva ROC al clasificar. Los parámetros configurables de los clasificadores y sus barridos son los siguientes:

- LDA:** parámetro de regularización “ReguAlpha” (desde 1 hasta 50000 con salto 1000);
- SRKDA:** parámetro de regularización “ReguAlpha” (desde 0.05 hasta 0.1 con salto 0.01) y parámetro  $t$  (desde 800 hasta 2200 con salto 100);
- SVM:** parámetro  $C$ , con valores [1 5 10 15 20 25 30].

La razón por la que se barren estos valores es porque se ha demostrado en la práctica que los valores óptimos suelen encontrarse dentro de este rango.

### 3.6. Validación del método propuesto

Una vez obtenidos los parámetros óptimos para cada clasificador, se entrenan mediante el conjunto de datos de entrenamiento. Tras ello, se aplican los modelos al conjunto de datos de test y se obtiene la probabilidad de que la fila o columna para cada iluminación fuera de clase 1 (objetivo) o -1 (no objetivo). Cada carácter tiene asignada una fila y una columna de la matriz. Para la decisión final se escoge aquel carácter cuya media de probabilidad de ser clase 1 para la fila y la columna que le corresponden sea mayor. El proceso se repite con los 100 caracteres y se calcula el porcentaje de aciertos comparando con los caracteres correctos.

Es importante destacar que los sujetos A y B por lo general tienen distintos parámetros óptimos para los clasificadores.

# Capítulo 4

## Resultados

### 4.1. Introducción

La efectividad de los métodos propuestos se evalúa en el conjunto de datos II de la III Competición BCI, 2004, ya descrito en capítulos previos. Se consideran tres tipos de clasificadores: LDA, SRKDA y conjunto de SVMs. El modelo se optimiza en una sesión de entrenamiento y tras ello se aplica en una sesión de evaluación.

Se obtienen resultados para cada modelo en función del porcentaje de aciertos del total de 100 caracteres, para cada sujeto de forma individual y la media de ambos, en función del número de iluminaciones tenidas en cuenta de 5 a 15.

En el capítulo posterior se comparan los resultados con los de otros estudios de interés.

### 4.2. Parámetros óptimos de los clasificadores

Tras realizar el proceso de selección de parámetros óptimos descrito en el capítulo previo se obtuvieron los siguientes resultados.

#### 4.2.1. LDA

El área bajo la curva ROC que se obtuvo en el proceso de validación cruzada se puede observar en la figura 4.1 para el sujeto A y en la figura 4.2 para el sujeto B. A su vez, en las figuras 4.3 y 4.4 se pueden ver las curvas ROC para el valor óptimo de cada sujeto. El valor óptimo para el parámetro de regularización fue de 13001 para el sujeto A, con un área bajo la curva ROC de 0.7834 y de 5001 para el B, con un área bajo la curva ROC de 0.8711.

#### 4.2.2. SRKDA

El área bajo la curva ROC que se obtuvo en el proceso de validación cruzada se puede observar en la figura 4.5 para el sujeto A y en la figura 4.6 para el sujeto B. A su vez, en las figuras 4.7 y 4.8 se pueden ver las curvas ROC para los valores óptimos de cada sujeto. El valor óptimo para el parámetro de regularización fue de 0.05 para ambos sujetos. El

parámetro  $t$  fue de 1000 para el sujeto A, con un área bajo la curva ROC de 0.7868 y de 900 para el B, con un área bajo la curva ROC de 0.8491.

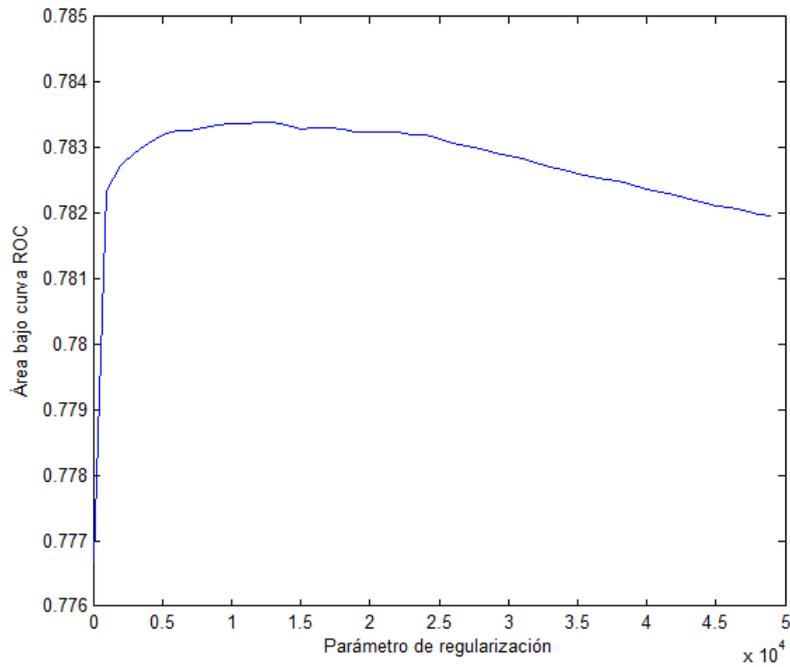


Figura 4.1: Área bajo la curva ROC para un barrido del parámetro de regularización del LDA. El valor óptimo es de 13001. Sujeto A.

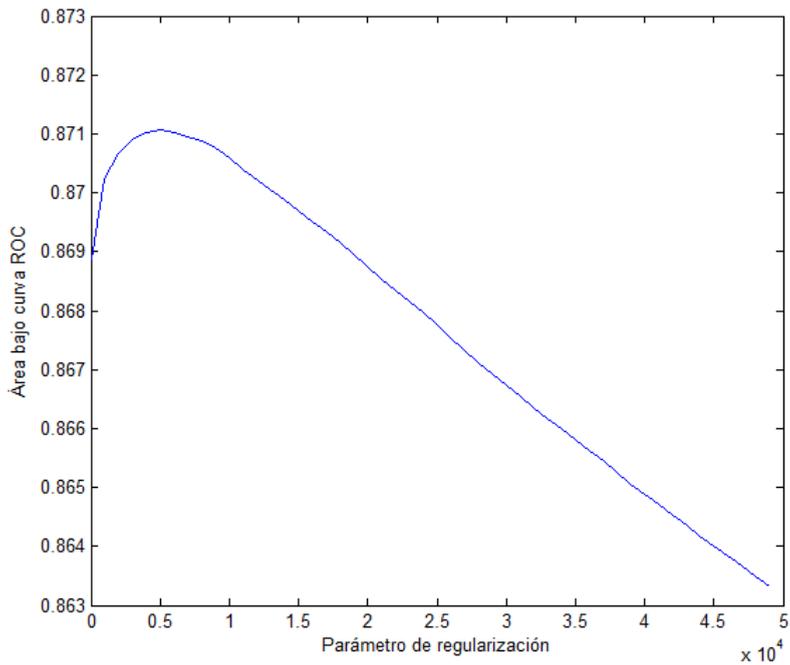


Figura 4.2: Área bajo la curva ROC para un barrido del parámetro de regularización del LDA. El valor óptimo es de 5001. Sujeto B.

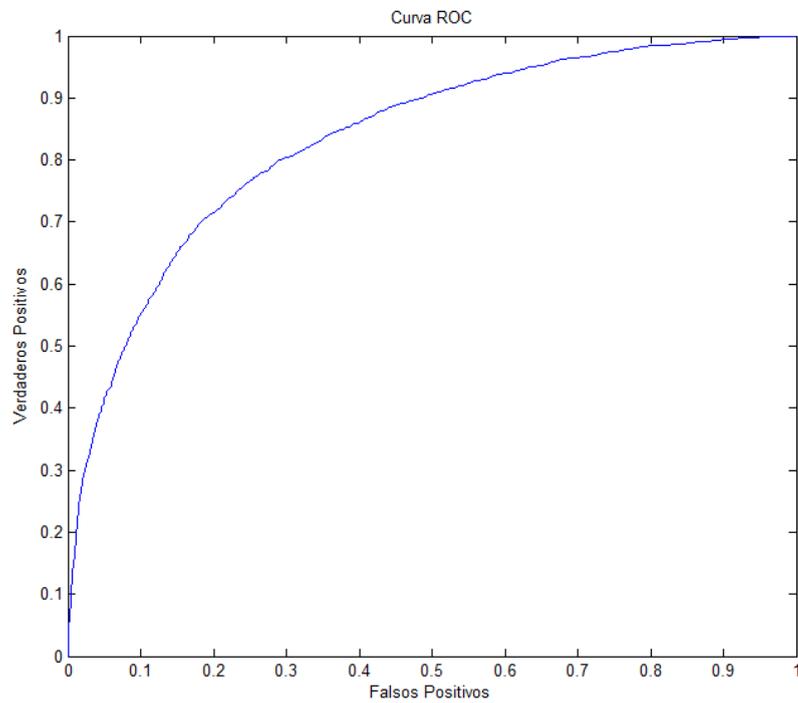


Figura 4.3: Curva ROC para el parámetro de regularización óptimo del Sujeto A. LDA.

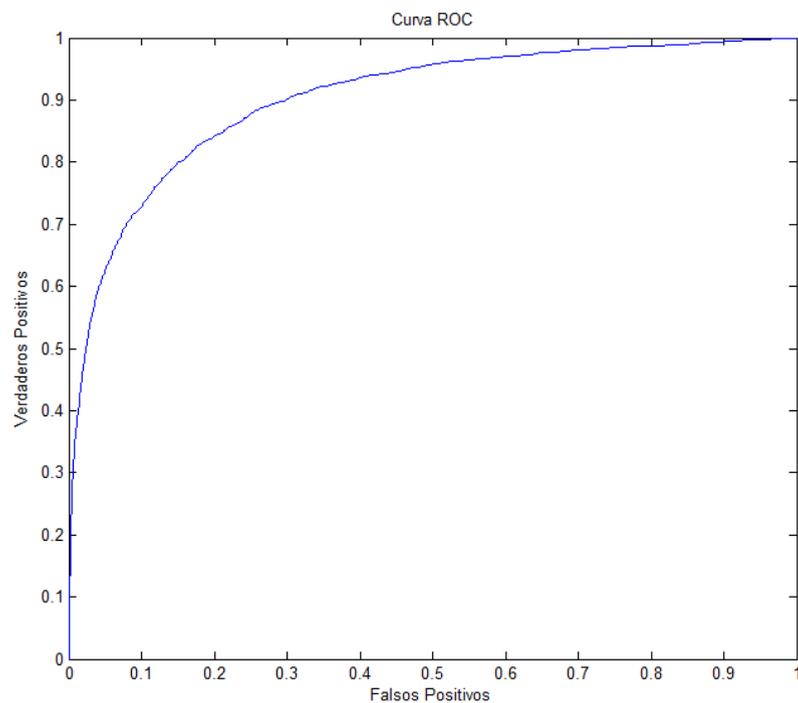


Figura 4.4: Curva ROC para el parámetro de regularización óptimo del Sujeto B. LDA.

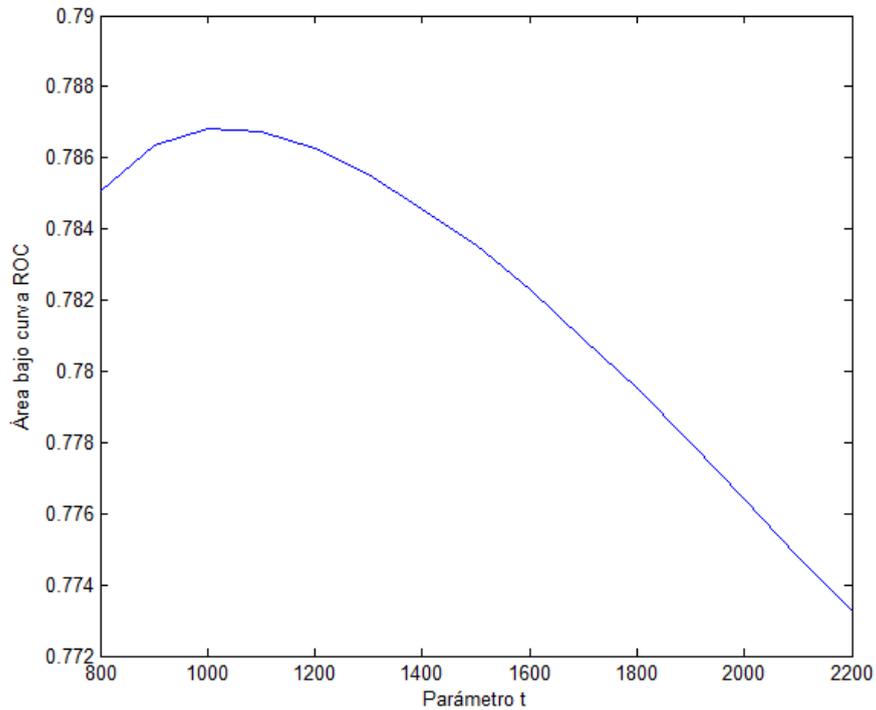


Figura 4.5: Área bajo la curva ROC para un barrido del parámetro  $t$  del SRKDA. El valor óptimo es de 1000. Para una mayor claridad solo se muestra el barrido para el valor del parámetro de regularización de 0.05. Sujeto A.

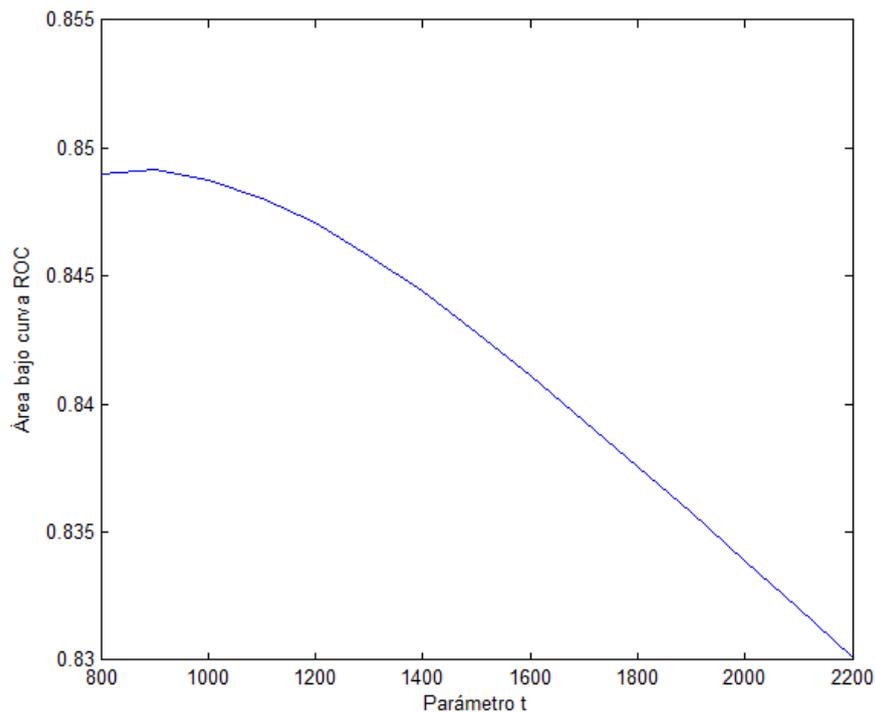


Figura 4.6: Área bajo la curva ROC para un barrido del parámetro  $t$  del SRKDA. El valor óptimo es de 900. Para una mayor claridad solo se muestra el barrido para el valor del parámetro de regularización de 0.05. Sujeto B.

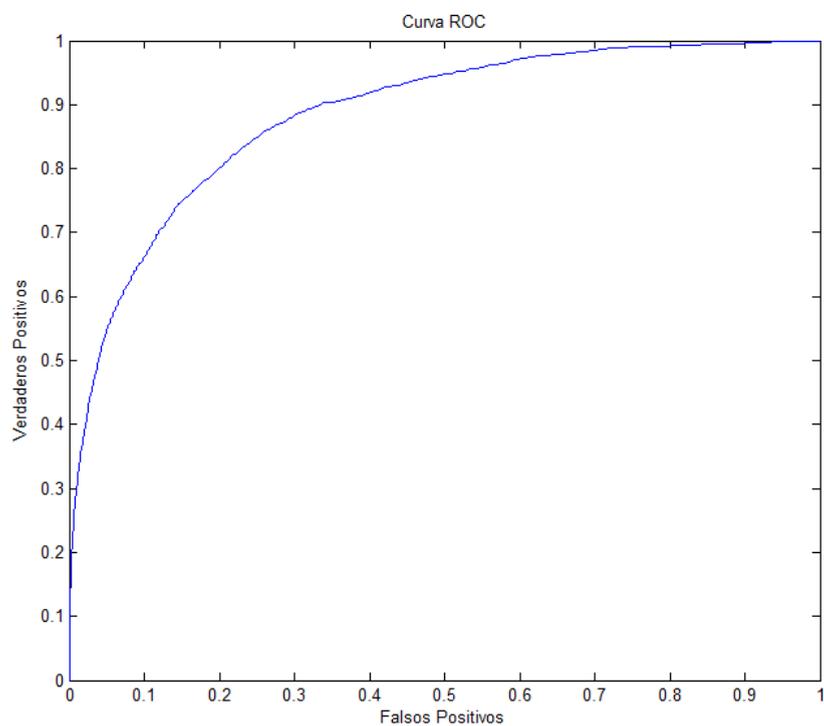


Figura 4.7: Curva ROC para el parámetro  $t$  óptimo del Sujeto A. SRKDA.

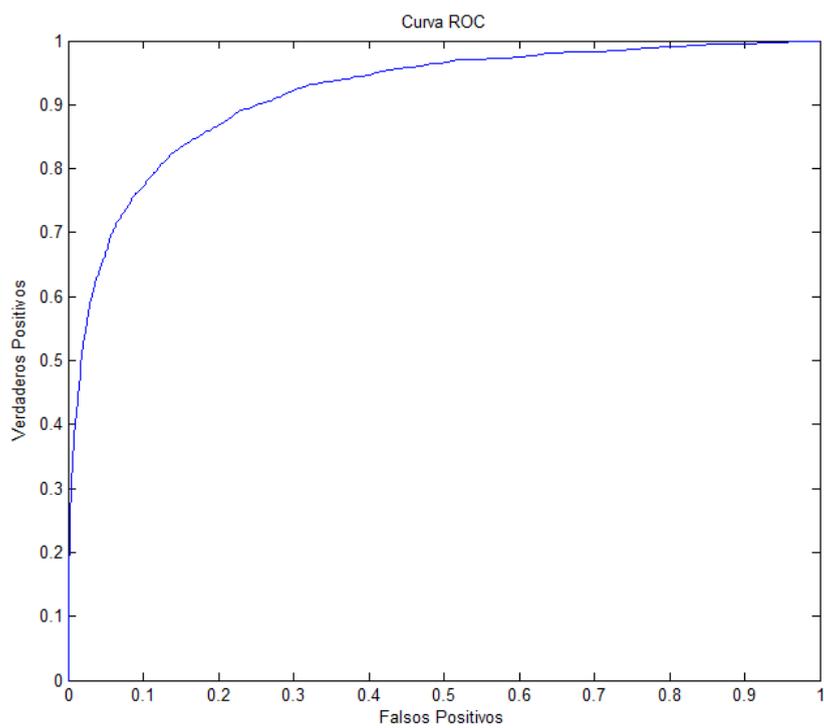


Figura 4.8: Curva ROC para el parámetro  $t$  óptimo del Sujeto B. SRKDA.

### 4.2.3. Conjunto de SVM

El área bajo la curva ROC que se obtuvo en el proceso de validación cruzada se puede observar en la figura 4.9 para el sujeto A y en la figura 4.10 para el sujeto B. El valor óptimo para el parámetro  $C$  fue de 30 para el sujeto A y el B, con un área bajo la curva ROC de 0.7834 para el A y un área bajo la curva ROC de 0.8711 para el B.

Una vez obtenido el parámetro  $C$  óptimo se procedió a determinar el número de SVM que se usarían en el conjunto. Dicho número tiene que ser impar para asegurar que nunca haya un mismo número de votos a favor de un resultado (1 o -1) en la clasificación. Para obtener el número ideal de máquinas se realizaron procedimientos de selección de caracteres sobre el conjunto de datos de entrenamiento, para 15 iluminaciones y barriendo desde 9 hasta 31 SVM. Los números que arrojaron mayor tasa de aciertos fueron 15, 17 y 19. Estos resultados se pueden ver en la tabla 4.1.

Es importante notar que SVM tiene un coste computacional que crece de forma cúbica con el número de señales de entrenamiento [51]. Teniendo en cuenta que cada SVM se entrena con una fracción de las señales de entrenamiento de  $1/N$ , donde  $N$  es el número de SVM del conjunto, es contraproducente usar pocas SVM. Finalmente se decidió usar 19 SVM puesto que entre aquellas con mayor tasa de aciertos era la que más máquinas contenía, por lo que requería un menor tiempo de entrenamiento.

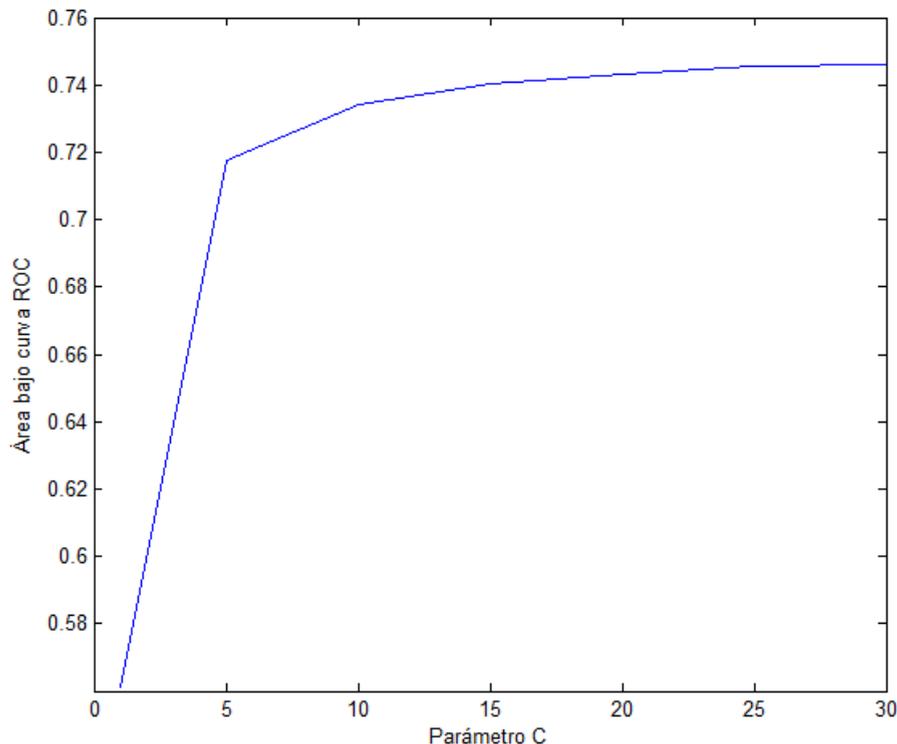


Figura 4.9: Área bajo la curva ROC para un barrido del parámetro  $C$  del conjunto de SVM. El valor óptimo de  $C$  es 30. Sujeto A.

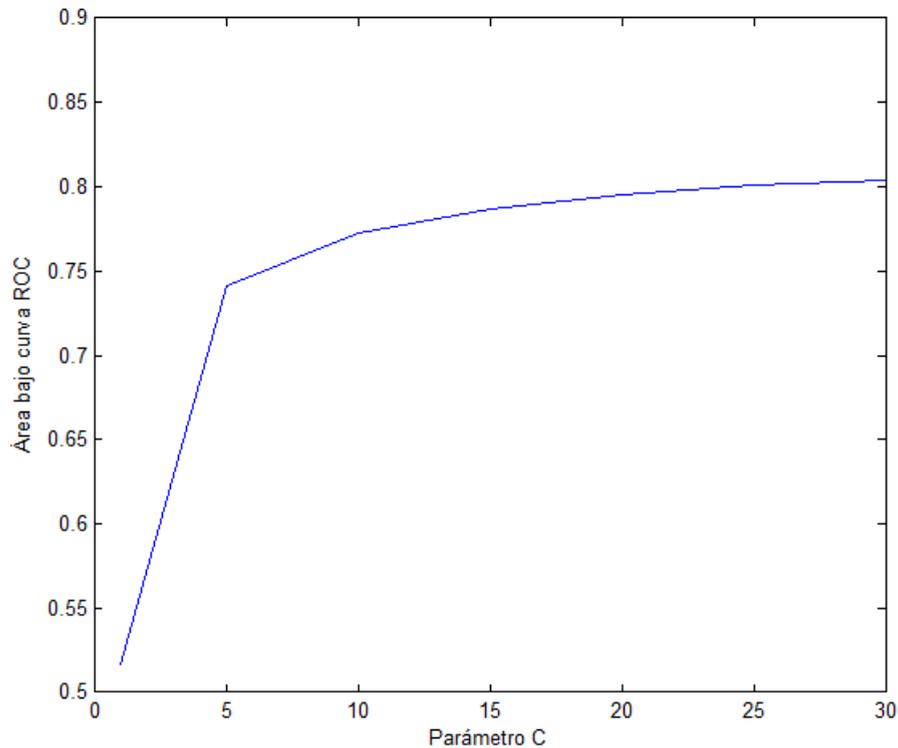


Figura 4.10: Área bajo la curva ROC para un barrido del parámetro  $C$  del conjunto de SVM. El valor óptimo de  $C$  es 30. Sujeto B.

	Número de SVM											
	9	11	13	15	17	19	21	23	25	27	29	31
	Porcentaje de aciertos (%)											
Sujeto A	98.82	98.82	98.82	100	100	100	98.82	97.65	100	98.82	98.82	97.65
Sujeto B	96.47	96.47	95,29	96,47	96,47	96,47	95,29	95,29	95,29	96,47	96,47	96,47
Media	97.64	97.64	97.05	98.23	98.23	<b>98.23</b>	97.64	96.47	97.64	97.64	97.64	97.06

Tabla 4.1: Resultados para 15 iluminaciones sobre el conjunto de datos de entrenamiento en función del número de SVM.

### 4.3. Resultados de los clasificadores

Una vez obtenidos los parámetros óptimos de los clasificadores se procedió a aplicar los mismos al conjunto de datos de la competición. Se realizó el proceso de clasificación para un número de iluminaciones desde 5 hasta el total de 15. Los resultados obtenidos se recogen en las tablas 4.2, 4.3 y 4.4.

### 4.3.1. LDA

La tabla 4.2 muestra los resultados obtenidos usando LDA con los parámetros de regularización óptimos para cada sujeto: 13001 para A y 5001 para B.

	Iluminaciones										
	5	6	7	8	9	10	11	12	13	14	15
	Porcentaje de aciertos (%)										
Sujeto A	65	73	74	82	85	89	95	97	98	97	98
Sujeto B	81	84	89	91	92	94	94	95	93	95	96
Media	73	78.5	81.5	86.5	88.5	91.5	94.5	96	95.5	96	97

Tabla 4.2: Resultados obtenidos mediante la clasificación usando LDA para distinto número de iluminaciones.

### 4.3.2. SRKDA

La tabla 4.3 muestra los resultados usando SRKDA y parámetros de regularización y  $t$  óptimos: 0.05 de regularización para ambos y parámetro  $t$  1000 para A y 900 para B.

	Iluminaciones										
	5	6	7	8	9	10	11	12	13	14	15
	Porcentaje de aciertos (%)										
Sujeto A	77	79	82	83	81	89	93	96	99	99	100
Sujeto B	79	81	84	87	90	92	93	94	95	95	94
Media	78	80	83	85	85.5	90.5	93	95	97	97	97

Tabla 4.3: Resultados obtenidos mediante la clasificación usando SRKDA para distinto número de iluminaciones.

### 4.3.3. Conjunto de SVM

La tabla 4.4 muestra los resultados obtenidos usando 19 SVM y parámetro  $C$  óptimo para cada sujeto.

	Iluminaciones										
	5	6	7	8	9	10	11	12	13	14	15
	Porcentaje de aciertos (%)										
Sujeto A	63	64	77	76	82	86	89	91	90	94	96
Sujeto B	74	72	85	85	89	92	91	92	90	92	94
Media	68.5	68	81	80.5	85.5	89	90	91.5	90	93	95

Tabla 4.4: Resultados obtenidos mediante la clasificación usando un conjunto de 19 SVM para distinto número de iluminaciones.

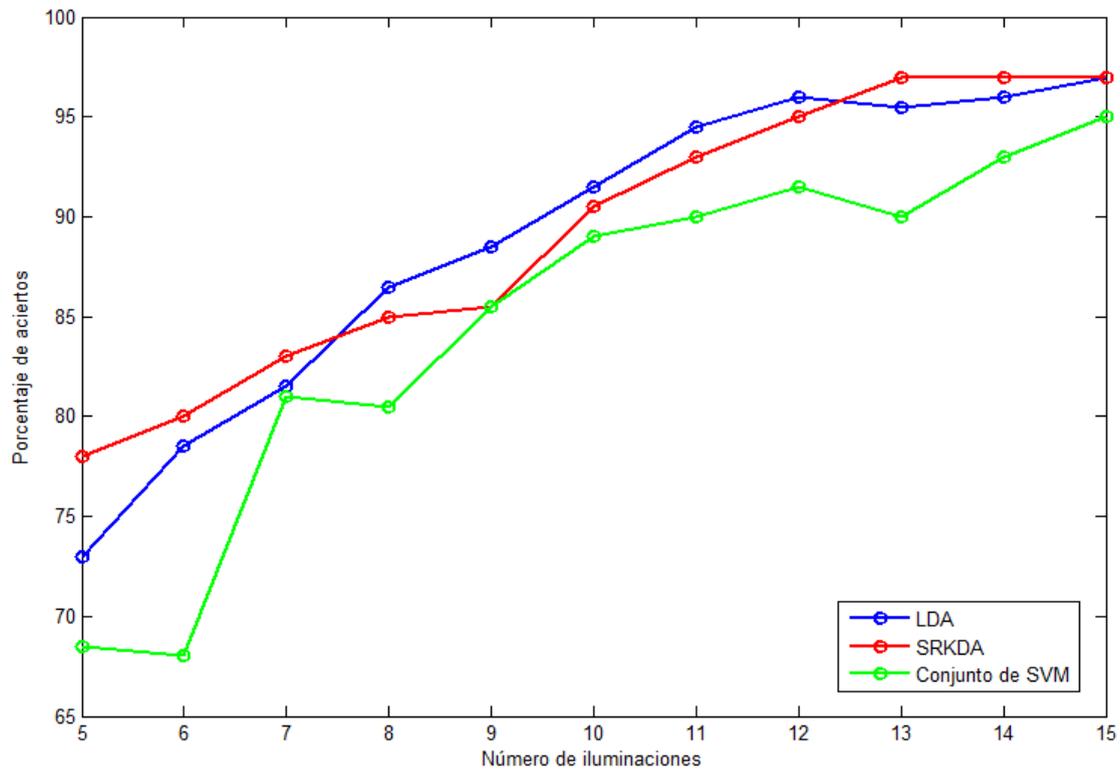


Figura 4.11: Comparativa porcentaje de aciertos de caracteres en función del método de clasificación y el número de iluminaciones utilizadas para la misma.

Finalmente, la figura 4.11 presenta una gráfica comparativa con los resultados medios obtenidos con los tres métodos de este estudio.



# Capítulo 5

## Discusión de los resultados

### 5.1. Introducción

Los resultados de los métodos propuestos se evalúan en el conjunto de datos II de la III Competición BCI, disponible de forma pública. Se consideran tres métodos diferentes de clasificación de características: LDA, SRKDA y un conjunto de 19 SVM. En primer lugar se obtienen los parámetros óptimos y se entrena al modelo mediante el conjunto de datos de entrenamiento, se aplica dicho modelo al conjunto de datos de test y finalmente se decide el carácter mediante la información proporcionada por dicha clasificación. A continuación se muestra la tabla 5.1 comparando los tres métodos para un total de 5 y 15 iluminaciones, ya que en la III Competición BCI se pedían los resultados para dichos números.

Se puede apreciar que el algoritmo que nos proporciona mejores resultados difiere para cada sujeto. Para el A, mediante SRKDA se obtiene un porcentaje satisfactorio de aciertos del 100 % para 15 iluminaciones, pero un 94 % para el B, lo que es más normal. Para el sujeto B, LDA es el método superior, obteniéndose un respetable 96 %. El conjunto de 19 SVM arroja los peores resultados para ambos sujetos, aunque tampoco se pueden considerar pobres en el contexto del estado del arte de la clasificación de P300.

Método	Sujetos					
	A		B		MEDIA	
	Número de iluminaciones					
	5	15	5	15	5	15
LDA	65	98	<b>81</b>	<b>96</b>	73	<b>97</b>
SRKDA	<b>77</b>	<b>100</b>	79	94	<b>78</b>	<b>97</b>
SVM	63	96	74	94	68.5	95

Tabla 5.1: Comparación de resultados de 5 y 15 iluminaciones para los distintos métodos de clasificación de este estudio. En negrita se marcan los valores máximos obtenidos.

Es interesante observar que por lo general el sujeto B obtiene mejores resultados en los tres métodos para 5 iluminaciones, especialmente para LDA, pero para 15 son inferiores que el A, con la excepción del conjunto de SVM. Este caso es especialmente notable en SRKDA con 15 iluminaciones, donde se observa la mayor diferencia de porcentaje de aciertos para un mismo método (100 % frente a 94 %).

Es importante destacar, por otra parte, que a pesar de que SRKDA obtiene los mejores resultados, también es computacionalmente el método más exigente, siendo con diferencia el que más tarda en construir el modelo y más recursos consume para ello, aunque sigue siendo menos costoso que KDA [46]. LDA, por otra parte, es computacionalmente poco exigente y obtiene los modelos de forma rápida.

Incluso así tampoco hay que dar excesiva importancia a la carga computacional y el tiempo requerido para construir los modelos puesto que esto se realiza tras la fase de entrenamiento y solo es necesario hacerlo una vez. Para la clasificación el tiempo requerido es siempre menor, con la excepción del conjunto de SVM, ya que se requiere realizarla para todas ellas y luego obtener un resultado ponderando los de los clasificadores individuales, lo que resulta en un mayor tiempo de procesamiento, aunque se puede reducir mediante computación en paralelo, lo que se hace en este estudio.

LDA y SRKDA proporcionan un porcentaje de aciertos global de 97 % para 15 iluminaciones, por lo que habría un empate en la decisión de cuál de ellos es superior. Sin embargo, y basándonos en el criterio de la III Competición BCI, SRKDA obtiene mejores resultados para 5 iluminaciones, por lo que globalmente es el método superior.

## 5.2. Comparación con otros métodos

En la tabla 5.2 se comparan los resultados de los métodos propuestos con los obtenidos por el método ganador [8].

Método	Sujetos					
	A		B		MEDIA	
	Número de iluminaciones					
	5	15	5	15	5	15
LDA	65	98	<b>81</b>	<b>96</b>	73	<b>97</b>
SRKDA	<b>77</b>	<b>100</b>	79	94	78	<b>97</b>
SVM	63	96	74	94	68.5	95
Rakotomamonjy y Guigue (2008) [8]	72	97	75	<b>96</b>	73.5	96.5

Tabla 5.2: Comparación de resultados con el método ganador. En negrita el método superior en cada caso.

Se observa que mediante SRKDA se ha podido superar al método ganador tanto en porcentaje de aciertos para 15 iluminaciones como para 5, por lo que este método es superior en conjunto. LDA supera también a los ganadores para 15 iluminaciones y está muy cerca para 5, si a esto le sumamos la sencillez y poca carga computacional del mismo se podría considerar superior también. El conjunto de 19 SVM de nuestro estudio obtiene peores resultados que las 17 del de Rakotomamonjy y Guigue, sin embargo nuestro método no incluye el procedimiento de selección de canales óptimos del ganador, por lo que es de esperar esta diferencia.

En cuanto a la comparación con otros métodos del estado del arte que usan la misma base de datos, los resultados, ordenados de mayor número de aciertos a menor, se observan en la tabla 5.3.

Los métodos de este estudio obtienen unos resultados globales de clasificación muy satisfactorios que los sitúan entre los mejores del estado del arte, solo siendo superados por el método propuesto por Amini et al (2012) [21]. Es importante tener en cuenta, sin embargo, que la mayoría de los métodos de esta comparativa se componen tanto de extracción de características como de clasificación de las mismas, al contrario que los realizados en este trabajo, que solamente realizan filtrado y diezmado previo a la clasificación. Es por esta razón que los resultados obtenidos son muy satisfactorios, puesto que existe la posibilidad potencial de mejorar los resultados de los mismos mediante la aplicación de distintos métodos de extracción de características.

Nº	Método	Aciertos ( %)
1	Amini et al (2012) [21]	97.40
2	SRKDA	97.00
3	LDA	97.00
4	El Dabbagh y Fakhr (2011) [41]	97.00
5	Shahriari y Erfanian (2011) [29]	96.90
6	Mirghasemi et al (2006) [36]	96.77
7	Rakotomamonjy y Guigue (2008) [8]	96.50
8	Sakamoto y Aono (2009) [38]	96.50
9	Tomioka y Haufe (2008) [42]	96.50
10	Cecotti y Gräser (2011) [24]	95.50
11	Conjunto de SVM	95.00
12	Salvaris et al (2009) [37]	95.00
13	Çelik y Anca (2010) [40]	95.50
14	Onishi et al (2012) [39]	92.00

Tabla 5.3: Comparación de resultados con otros métodos del estado del arte ordenados por porcentaje de aciertos para 15 iluminaciones.

### 5.3. Limitaciones de este estudio

La principal limitación de este estudio es el hecho de que no se han aplicado métodos de extracción de características más allá de un filtrado y diezmado básicos, por lo que puede que los resultados no sean todo lo satisfactorios que podrían llegar a ser si se realizara alguno.

Otra limitación importante viene dada por el formato de la III Competición BCI: dos sujetos es una muestra de población muy pequeña como para considerar generalizables los resultados de nuestro estudio. Sería deseable contar con un conjunto de datos de mayor tamaño para realizar las pruebas en más sujetos.

Las pruebas realizadas se han hecho sobre bases de datos registradas, no sobre usuarios en tiempo real, por lo que no se ha podido comprobar si son viables para una aplicación real debido a problemas potenciales como un excesivo tiempo de procesamiento previo a la clasificación.

En este trabajo se han tratado solamente tres algoritmos de clasificación de características, el estudio podría haberse ampliado con otros métodos, tales como la regresión logística.

Además, se podrían haber aplicado métodos de extracción de características a las señales de forma previa a su clasificación, lo que previsiblemente mejoraría los resultados obtenidos. A su vez, el método ganador usaba un procedimiento de selección de canales que habría sido interesante probar tanto en el conjunto de SVM como en los otros algoritmos, o incluso desarrollar uno propio con el mismo objetivo.

El conjunto de datos II de la III Competición BCI se obtuvo mediante métodos de adquisición descritos previamente. Las señales podrían haberse extraído con diferentes procedimientos con vistas a ver cuál de ellos proporciona mejores resultados para un mismo algoritmo.

# Capítulo 6

## Conclusiones y líneas futuras

### 6.1. Conclusiones

A lo largo de este trabajo se han estudiado los sistemas BCI, con especial hincapié en aquellos basados en potenciales evocados P300. Estos sistemas permiten que usuarios en situación de discapacidad motora puedan interactuar con dispositivos sin necesidad de los músculos periféricos. Las señales han sido registradas mediante el procedimiento del EEG de tipo no invasivo. Dichas señales han sido potenciales evocados P300, las cuales tienen la ventaja de que no necesitan entrenamiento previo por parte del usuario puesto que son respuestas espontáneas.

El estudio realizado se ha centrado en la aplicación de tres métodos de clasificación de características diferentes a las señales del conjunto de datos II de la III Competición BCI de 2004, proporcionados por el centro Wadsworth (*NYS Department of Health*) [34], que consisten en una serie de potenciales evocados P300 registrados a dos usuarios mediante EEG. Dichas señales se dividen en dos subconjuntos: uno de entrenamiento y uno de test. Se han entrenado los tres métodos (LDA, SRKDA y SVM) con el primer subconjunto y se ha desarrollado un modelo, el cual se ha aplicado posteriormente a las señales de test. Los resultados de la clasificación se han utilizado para elegir el carácter buscado en cada caso. Dichos resultados se han comparado con los obtenidos por el método ganador de la competición [8], consistente en un conjunto de 17 SVM con selección de canales, así como con otros métodos actuales [21, 24, 29, 36–42].

Las principales conclusiones a las que se han llegado se pueden resumir en los puntos siguientes:

- La precisión media en la clasificación obtenida por LDA y SRKDA (97 % para ambos) ha superado a la del método ganador de la competición. En particular SRKDA lo ha hecho tanto para 5 como para 15 iluminaciones (con un 79 % y 97 % respectivamente), pudiéndose considerar por tanto como superior globalmente.
- LDA no ha superado al método ganador para 5 iluminaciones, pero cuenta con la ventaja de ser un método computacionalmente poco exigente y rápido.
- SVM ha proporcionado unos resultados de clasificación inferiores al método ganador (95 %) pero dentro de un rango aceptable. La razón de la diferencia en la tasa de

aciertos probablemente radica en la ausencia de selección de canal en el conjunto de SVM de nuestro estudio.

- Comparando con otros métodos del estado del arte, SRKDA y LDA han quedado en segundo y tercer puesto en cuanto a precisión para 15 iluminaciones, por lo que se pueden considerar sus resultados como satisfactorios y relevantes.
- La respuesta de los sujetos A y B es diferente ante distintos métodos de clasificación. En particular el sujeto A responde mejor a SRKDA y el B a LDA. Los parámetros óptimos de los clasificadores son distintos para ambos.
- LDA ha resultado ser el método más rápido y de menor carga computacional, por lo que su aplicación a sistemas en tiempo real se podría considerar viable.
- SRKDA requiere un tiempo considerable para la construcción del modelo. Esto no es problemático debido a que esta operación se puede realizar entre la sesión de entrenamiento y la de test. El tiempo de clasificación a partir de un modelo ya construido es aceptable aunque mayor que el de LDA.
- El conjunto de SVM requiere mayor tiempo de selección de caracteres que LDA y SRKDA debido a que realiza la misma en función de la clasificación de las señales realizada por cada SVM, teniéndose que esperar a que todas la hayan realizado. Independientemente de ello mediante computación en paralelo se puede reducir la duración de la selección.

## 6.2. Líneas futuras

Tras realizar este estudio es necesario señalar algunas limitaciones que se han ido encontrando y en base a las mismas proponer futuras líneas de investigación.

- Los resultados han sido obtenidos aplicando algoritmos de clasificación directamente a señales EEG filtradas y diezmadas sin realizar ningún procedimiento de extracción de características previo. Sería muy interesante, en vista a los buenos resultados obtenidos, tratar las señales previamente con distintos métodos de extracción de características y comparar los resultados con vistas a mejorar la precisión global.
- Otra limitación importante es el pequeño tamaño de la base de datos de la competición. Para futuros trabajos sería recomendable disponer de señales registradas de más usuarios en un mayor número de sesiones.
- Sería muy interesante probar los métodos propuestos con otras personas y analizar la precisión obtenida, el tiempo necesario para la clasificación y la viabilidad en general de los mismos para aplicaciones en tiempo real.
- En cuanto al conjunto de SVM, sería interesante realizar un procedimiento de selección de canales similar o diferente al del método ganador y comparar los resultados con el fin de obtener un número óptimo de los mismos.

- Con vistas a determinar qué métodos de adquisición de señales son más adecuados para cada algoritmo de clasificación se podría probar a obtener nuevas bases de datos con distintos procedimientos y aplicar LDA, SRKDA y SVM a cada una de ellas. Tras esto se procedería a comparar los resultados y ver cuál es el método de adquisición óptimo para cada uno de los algoritmos de clasificación.
- Finalmente se podría ampliar el estudio de métodos de clasificación de características con otros algoritmos como la regresión logística.

En conclusión, los resultados de este estudio se pueden considerar satisfactorios aunque limitados: se ha conseguido el objetivo de superar al método ganador en 2 de los 3 métodos de clasificación estudiados y además se han situado en puestos altos de la clasificación global de métodos del estado del arte. Por otra parte, debido principalmente a la ausencia de algoritmos de extracción de características, estos resultados son potencialmente mejorables en futuros estudios, lo que resulta prometedor.



# Referencias

- [1] “Biomedical Signal Processing”, About Biomedical Engineering, IEEE Engineering in Medicine & Biology Society, <http://www.embs.org/about-biomedical-engineering/our-areas-of-research/biomedical-signal-processing>
- [2] E. N. Bruce, “Biomedical signal processing and signal modelling”, John Wiley and Sons, 2001.
- [3] E. Niedermeyer, F.L. da Silva. “Electroencephalography: Basic Principles, Clinical Applications, and Related Fields”, Lippincot Williams & Wilkins, 2004.
- [4] G. H. Klem, H. O. Lüders, H. H. Jasper, C. Elger, “The ten twenty electrode system of the international federation”, *Electroencephalography and Clinical Neurophysiology*, vol. 10, pp. 214–215, 1958.
- [5] K. E. Misulis, T. Fakhoury, ”Spehlmann’s Evoked Potential Primer”, Butterworth-Heinemann., 2001.
- [6] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, T. M. Vaughan, “Brain–computer interfaces for communication and control”, *Clinical Neurophysiology*, vol. 113, pp. 767–791, 2002.
- [7] D. J. Creel, “Visually Evoked Potentials”, Webvision, 2012.
- [8] A. Rakotomamonjy, V. Guigue, “BCI Competition III: Dataset II – Ensemble of SVMs for BCI P300 Speller”, *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 3, pp. 1147–1154, 2008.
- [9] A. Kübler, K. Müller, “An Introduction to Brain-Computer Interfacing”, *Toward Brain-Computer Interfacing*, MIT Press, Cambridge, Massachusetts, pp. 1-25, 2007.
- [10] X. Gao, D. Xu, M. Cheng, S. Gao, “A BCI-based environmental controller for the motion-disabled”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 11, pp. 137–140, 2003.
- [11] K.D. Nielsen, A.F. Cabrera, O.F. do Nascimento, “EEG based BCI-towards a better control. Brain-computer interface research at Aalborg University”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, pp. 202–204, 2006.

- [12] K.C. Squires, K.C.E. Donchin, R.I. Herning, G. McCarthy, “On the influence of task relevance and stimulus probability on event-related-potential components, *Electroencephalography and Clinical Neurophysiology*, vol. 42, pp.1–14, 1977.
- [13] L.A. Farwell, E. Donchin, “Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials”, *Electroencephalography and Clinical Neurophysiology*, vol. 70, pp. 510–523, 1988.
- [14] R. Corralejo, “Extracción de Características de Electroencefalogramas. Aplicación a Brain Computer Interface”, Proyecto Fin de Carrera, Departamento de Teoría de la Señal y Comunicaciones, Valladolid, Universidad de Valladolid, 2008.
- [15] L. Mayaud, M. Congedo, A. Van Laghenhove, D. Orlikowski, M. Figère, E. Azabou, F. Cheliout-Heraut, “A comparison of recording modalities of P300 event-related potentials (ERP) for brain-computer interface (BCI) paradigm”, *Neurophysiologie Clinique/Clinical Neurophysiology*, vol. 43, no. 4, pp 217–227, 2013.
- [16] F. Akram, M.K. Metwally, Hee-Sok Han, Hyun-Jae Jeon, Tae-Seong Kim, “A Novel P300-Based BCI System for Words Typing”, *International Winter Workshop on Brain-Computer Interface (BCI)*, pp. 24–25, 2013.
- [17] S.T. Ahi, H. Kambara, Y. Koike, “A Dictionary-Driven P300 Speller with a Modified Interface”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 1, pp. 6–14, 2011.
- [18] D. J. McFarland, W.A. Sarnacki, G. Townsend, T. Vaughan, J. R. Wolpaw, “The P300-based brain–computer interface (BCI): Effects of stimulus rate”, *Clinical Neurophysiology*, vol. 122, no. 4, pp. 731–737, 2011.
- [19] J. Guo, S. Gao, B. Hong “An Auditory Brain–Computer Interface Using Active Mental Response”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 3, pp. 230–235, 2010.
- [20] D. Aminaka, K. Mori, T. Matsui, S. Makino, T.M. Rutkowski, “Bone-conduction-based brain Computer Interface Paradigm-EEG Signal Processing, Feature Extraction and Classification”, *International Conference on Signal-Image Technology & Internet-Based Systems*, pp. 818–824, 2013.
- [21] Z. Amini, V. Abootalebu, M. T. Sadeghi, “Comparison of Performance of Different Feature Extraction Methods in Detection of P300”, *Biocybernetics and Biomedical Engineering*, vol. 33, no. 1, pp. 3–20, 2013.
- [22] Y. Shahriari, A. Erfanian, “Improving the performance of P300-based brain–computer interface through subspace-based filtering”, *Neurocomputing*, vol. 121, pp. 434–441, 2013.
- [23] A. Combaz, N. Chumerin, N.V. Manyakov, A. Robben, J.A.K. Suykens, M.M. Van Hulle, “Towards the detection of error-related potentials and its integration in the context of a P300 speller brain–computer interface”, *Neurocomputing*, vol. 80, pp. 73–82, 2012.

- [24] H. Cecotti, A. Graser, “Convolutional Neural Networks for P300 Detection with Application to Brain-Computer Interfaces”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 433–445, 2011.
- [25] A. Turnip, A. Widyotriatmo and Suprijanto, “Classification of EEG-P300 signals using Fisher’s linear discriminant analysis”, *3rd International Conference on Instrumentation Control and Automation (ICA)*, pp. 98–103, 2013.
- [26] X. Li, F. Wang, X. Chen, R.K. Ward, “A P300-based BCI classification algorithm using median filtering and Bayesian feature extraction”, *Multimedia Signal Processing (MMSP)*, IEEE 14th International Workshop, pp. 305–308, 2012.
- [27] W. Speier, C. Arnold, J. Lu, A. Deshpande, N. Pouratian, “Integrating Language Information With a Hidden Markov Model to Improve Communication Rate in the P300 Speller”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, por publicar, 2014.
- [28] A. González, I. Nambu, H. Hokari, M. Iwahashi, Y. Wada, “Towards the Classification of Single-trial Event-Related Potentials using Adapted Wavelets and Particle Swarm Optimization”, *IEEE International Conference on Systems, Man and Cybernetics*, pp. 3089–3094, 2013.
- [29] Y. Shahriari, A. Erfanian, “A Mutual Information Based Channel Selection Scheme for P300-based Brain Computer Interface”, *5th International IEEE EMBS Conference on Neural Engineering*, pp. 434–437, 2011.
- [30] J. P. Rosenfeld, X. Hu, E. Labkovsky, J. Meixner, M. R. Winograd, “Review of recent studies and issues regarding the P300-based complex trial protocol for detection of concealed information”, *International Journal of Psychophysiology*, vol. 90, no. 2, pp. 118–134, 2013.
- [31] B. Rebsamen, C. Guan, H. Zhang, C. Wang, C. Teo, M. H. Ang, Jr., E. Burdet, “A Brain Controlled Wheelchair to Navigate in Familiar Environments”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 6, pp. 590–598, 2010.
- [32] H. Riechmann, N. Hachmeister, H. Ritter, A. Finke, “Asynchronous, parallel on-line classification of P300 and ERD for an efficient hybrid BCI”, *5th International IEEE EMBS Conference on Neural Engineering*, pp. 412–415, 2011.
- [33] Y. Li, J. Long, T. Yu, Z. Yu, C. Wang, H. Zhang, C. Guan, “An EEG-Based BCI System for 2-D Cursor Control by Combining Mu/Beta Rhythm and P300 Potential”, *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 10, pp. 2495–2505, 2010.
- [34] D. Krusienski, G. Schalk, “BCI Competition III Challenge 2004”, 2004: [http://www.bbci.de/competition/iii/desc\\_II.pdf](http://www.bbci.de/competition/iii/desc_II.pdf)

- [35] L. Martín, “Desarrollo de un sistema Brain Computer Interface basado en potenciales evocados P300 para el control de dispositivos domóticos”, Trabajo Fin de Máster, Departamento de Teoría de la Señal y las Comunicaciones, Valladolid, Universidad de Valladolid, 2012.
- [36] H. Mirghasemi, M.B. Shamsollahi, R. Fazel-Rezai, “Assessment of Preprocessing on Classifiers Used in the P300 Speller Paradigm”, *Proceedings of the 28th IEEE EMBS Annual International Conference*, pp. 1319–1322, 2006.
- [37] M. Salvaris, F. Sepúlveda, “Wavelets and Ensemble of FLDs for P300 Classification”, *Proceedings of the 4th International IEEE EMBS Conference on Neural Engineering*, pp. 339–342, 2009.
- [38] Y. Sakamoto, M. Aono, “Supervised Adaptive Downsampling for P300-based Brain Computer Interface”, *31st Annual International Conference of the IEEE EMBS*, pp. 567–570, 2008.
- [39] A. Onishi, A. H. Phan, K. Matsuoka, A. Cichocki, “Tensor Classification for P300-based Brain Computer Interface”, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 581–584, 2012.
- [40] Umut Çelik, Sami Anca, “Classification of the Visual Evoked EEG Using Multi-resolution Approximation Based on Excitatory Post-Synaptic Potential Waveform”, *IEEE 26th Convention of Electrical and Electronics Engineers in Israel (IEEEI)*, pp. 929–933, 2010.
- [41] H. El Dabbagh, W. Fakhir, “Multiple classification algorithms for the BCI P300 speller diagram using ensemble of SVMs”, *IEEE GCC Conference and Exhibition (GCC)*, pp. 393–396, 2011.
- [42] R. Tomioka, S. Haufe, “Combined classification and channel/basis selection with L1-L2 regularization with application to P300 speller system”, *4th International Brain-Computer Interface Workshop and Training Course*, 2008.
- [43] C.M. Bishop, “Pattern Recognition and Machine Learning”, Springer, New York, 2006.
- [44] S. Haykin, “Neural Networks. A Comprehensive Foundation”, Prentice Hall International, 1999.
- [45] L.F Nicolás-Alonso, R. Corralejo, J. Gómez-Pilar, D. Álvarez, R. Hornero, “Adaptive semi-supervised classification to reduce intersession non-stationarity in multi-class motor imagery-based brain computer interfaces”, *Journal of Neural Engineering*, 2014. In press.
- [46] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, K. Mullers, “Fisher discriminant analysis with kernels”, *Proceedings of IEEE Signal Processing Society Workshop*, (Madison, WI, Aug. 23–25) pp 41–48, 1999.

- 
- [47] D. Cai, X. He, J. Han, “Speed up kernel discriminant analysis”, *The VLDB Journal* no. 20 pp 21–33, 2011.
- [48] M. García, “Procesado de retinografías basado en redes neuronales para la detección automática de lesiones asociadas a la retinopatía diabética”, M. S. Thesis, Departamento de Teoría de la Señal y las Comunicaciones, Valladolid, Universidad de Valladolid, 2008.
- [49] K. R. Müller, C. W. Anderson, G. E. Birch, “Linear and Nonlinear Methods for Brain Computer Interfaces,” *IEEE Transaction on Rehabilitation Engineering*, vol. 11, no. 2, pp. 165–169, 2003.
- [50] BCI Competition III, Final Results, <http://www.bbci.de/competition/iii/results/>
- [51] L. Bottou, C.-J. Lin, “Support vector machine solvers”, in *Large-Scale Kernel Machines*, L. Bottou, O. Chapelle, D. DeCoste and J. Weston (Eds.), The MIT Press, pp. 1–27, 2007.



# Anexo A

## Listado del código desarrollado en Matlab

### Fichero cargarDatos.m

```
function [ Data ] = cargarDatos( nombreFichero )  
  
%Extractor de Datos en estructura  
  
    Data=load(nombreFichero);  
  
end
```

### Fichero clasificacion.m

```
function [p,clase] = clasificacion( trainFeatures, testFeatures, trainClass, metodo)  
  
%Funcion que llama a los distintos clasificadores  
  
%Clasificador SRKDA  
  
if (strcmp(metodo,'SRKDA'))  
    %Opciones del clasificador  
    options = [];  
    options.KernelType = 'Gaussian';  
    options.ReguType = 'Ridge';  
    options.ReguAlpha = 0.05;  
    options.t = 900;  
  
    [clase,p]=clasificacionSRKDA(trainFeatures,testFeatures,trainClass,options);  
end  
  
% Clasificador LDA  
  
if (strcmp(metodo,'LDA'))  
    %Opciones del clasificador  
    [clase,err,p]=classify(testFeatures,trainFeatures,trainClass,'linear');  
    p=p(:,2);  
end  
  
% Clasificador LDA2  
  
if (strcmp(metodo,'LDA2'))  
    %Opciones del clasificador  
    options=[];  
    options.Regu = 1;  
    options.ReguAlpha = 13001;  
    [clase,p]=clasificacionLDA(trainFeatures, trainClass, testFeatures,options);  
    p=p(:,1);  
end
```

```

% Clasificador SVM

if (strcmp (metodo,'SVM'))
    %Numero de divisiones k
    addpath('SVM\libsvm-3.18\matlab')
    k=31;
    longitud=ceil(length(trainClass)/k);
    [longitudTest,nada]=size(testFeatures);
    claseaux=NaN(longitudTest,k);
    p=NaN(longitudTest,1);
    clase=p;
    N = size(trainFeatures,1);
    parfor ii=1:k
        featuresTrain=(trainFeatures(longitud*(ii-1)+1:min(longitud*ii,N),:));
        claseTrain=(trainClass(longitud*(ii-1)+1:min(longitud*ii,N)));
        claseaux(:,ii)=clasificacionSVM(featuresTrain, claseTrain, testFeatures);
    end
    for ii=1:longitudTest
        cont=0;
        for a=1:k
            if(claseaux(ii,a)==1)
                cont=cont+1;
            end
        end
        p(ii)=cont/k;
        if(mean(claseaux(ii,:))>=0)
            clase(ii)=1;
        else
            clase(ii)=-1;
        end
    end
end
end

```

## Fichero clasificacionCV.m

```

% Procedimiento de variacion cruzada

% Descomentar y comentar en funcion del clasificador

% Parametros
dir='..\..\02mat';
metodoExt='ganador';
metodoClas='SVM';
sujetos={'Sujeto A', 'Sujeto B'};
tipoCurva='ROC';

Nsujetos=size(sujetos,2);

for it=1:Nsujetos
    fprintf('%s\n',sujetos{it})
    suj=sprintf('%s SVM',sujetos{it});

    train=sprintf('%s\\%s\\train\\%s.mat',dir,metodoExt,sujetos{it});
    test=sprintf('%s\\%s\\test\\%s.mat',dir,metodoExt,sujetos{it});
    trainData=cargarDatos(train);
    testData=cargarDatos(test);

    %% Obtencion parametros ideales
    trainData.features=reshape(permute(trainData.features,[2 1 3]),15300,896);
    trainData.class=reshape(permute(trainData.class,[2 1]),15300,1);

    %features = trainData.features(1:2000,:);
    %class = trainData.class(1:2000,:);
    features = trainData.features;
    class = trainData.class;

    %a = 0.05:0.01:0.1; %SRKDA
    %t = 800:100:2200; %SRKDA

```

```

%regu=1:1000:50000; %LDA2
%matriz = crossvalmatrix( a,t ); %SRKDA
%N = length(matriz); %SRKDA
%N=length(regu); %LDA
%cparam=[0.01 0.05 0.1 0.5 1]; %kernel lineal
cparam=[1 5 10 15 20 25 30];
%gammainv=10000:1000000:100000000;

matriz=crossvalmatrix( gammainv,cparam ); %SVM
N=length(matriz); %SVM
cvMCRplot=NaN(1,N);
for i = 1:N
    %% Param
    %a=matriz(i,1); %SRKDA
    %t=matriz(i,2); %SRKDA
    gamma=1/matriz(i,1); %SVM
    cparam=matriz(i,2); %SVM
    %% CV
    s = RandStream('mt19937ar','Seed',2014);
    RandStream.setGlobalStream(s);
    cp = cvpartition(class,'k',10);
    string=sprintf('-g %f -c %f -w1 11 -w-1 1 -q', gamma, cparam);
    %perf = @(xtrain, ytrain,xtest, ytest) curvas(ytest, ...
        clasificacionParams(xtrain,xtest,ytrain,metodoClas,a(1,i)),tipoCurva);
    %perf = @(xtrain, ytrain,xtest, ytest) curvas(ytest, ...
        clasificacionParams(xtrain,xtest,ytrain,metodoClas,[a t]),tipoCurva);
    %perf = @(xtrain, ytrain,xtest, ytest) curvas(ytest, ...
        clasificacionParams(xtrain,xtest,ytrain,metodoClas,regu(i)),tipoCurva);
    perf = @(xtrain, ytrain,xtest, ytest) curvas(ytest, ...
        clasificacionParams(xtrain,xtest,ytrain,metodoClas,string),tipoCurva);
    cvMCR = crossval(perf,features,class,'partition',cp);
    cvMCR=mean(cvMCR);
    %fprintf('Par1: %.02f Par2: %.02f\n', a, t); %SRKDA
    %fprintf('Parametro regularizacion: %.02f\n', regu(i)); %LDA2
    fprintf('Gamma: %d c: %.02f\n', gamma, cparam); % SVM
    fprintf('Performance: %.02f\n\n', cvMCR);
    cvMCRplot(i)=cvMCR;
end
figure
%plot(regu,cvMCRplot);
plot(cvMCRplot);
[maximo,posicion]=max(cvMCRplot);
%aopt=matriz(posicion,1) %SRKDA
%topt=matriz(posicion,2) %SRKDA
%reguOpt=regu(posicion); %LDA2
gopt=1/matriz(posicion,1);
%gopt=1/(matriz(posicion,2));
copt=matriz(posicion,2);

%% Clasificacion

testData.features=reshape(permute(testData.features,[2 1 3]),18000,896);
testData.class=reshape(permute(testData.class,[2 1]),18000,1);

%p,clase] = clasificacionParams( trainData.features, testData.features, ...
    trainData.class,metodoClas,[aopt topt]);
%p,clase] = clasificacionParams( trainData.features, testData.features, ...
    trainData.class,metodoClas,reguOpt);
string=sprintf('-g %f -c %f -w1 11 -w-1 1 -q', gopt, copt);
[p,clase] = clasificacionParams( trainData.features, testData.features, ...
    trainData.class,metodoClas,string);

%% Letras

testData.code=reshape(permute(testData.code,[2 1]),18000,1);
[letras, caract]=determinarLetra(p,testData.code,testData.matrix);

%% Evaluacion

aciertos=posicionLetras(letras,testData.target,caract);

```

```

%save(suj,'cvMCRplot','reguOpt','aciertos','regu') %LDA2
%save(suj,'cvMCRplot','aopt','topt','aciertos','matriz') %SRKDA
save(suj,'cvMCRplot','gopt','copt','aciertos','matriz') %SVM

```

```
end
```

### Fichero clasificacionLDA.m

```

function [clase,p]=clasificacionLDA(trainFeatures, trainClass, testFeatures,options)

% Clasificador LDA

    addpath('LDA')

    s = RandStream('mt19937ar','Seed',0);
    RandStream.setGlobalStream(s);
    N=size(trainClass,1);
    w=NaN(N,1);
    num1=length(find(trainClass==1));
    num2=length(find(trainClass==-1));
    p1=N/num1;
    p2=N/num2;
    w(trainClass==-1,1)=(p2)/(p1+p2);
    w(trainClass==1,1)=(p1)/(p1+p2);
    y = randsample(1:N,100000,true,w);

    % trainFeatures=trainFeatures(y,:);
    % trainClass=trainClass(y,:);
    model = LDAttrain(trainFeatures, trainClass, options);
    [accuracy, clase, p] = LDApredict(trainFeatures, trainClass, model);
    if mean(p(trainClass == -1,1)) > mean(p(trainClass ~= -1,1))
        model.eigvector = -model.eigvector;
    end
    N = size(testFeatures,1);

    [accuracy, clase, p] = LDApredict(testFeatures, ones(N,1), model);
end

```

### Fichero clasificacionParams.m

```

function [p,clase] =clasificacionParams(trainFeatures,testFeatures,trainClass,metodo,params)

% Funcion que llama a los distintos clasificadores

% Clasificador SRKDA

if (strcmp(metodo,'SRKDA'))
    %Opciones del clasificador
    options = [];
    options.KernelType = 'Gaussian';
    options.ReguParam = 'Ridge';
    options.ReguParam = params(1);
    options.t = params(2);

    [clase,p]=clasificacionSRKDA(trainFeatures,testFeatures,trainClass,options);
end

% Clasificador LDA

if (strcmp(metodo,'LDA'))
    %Opciones del clasificador

    [clase,err,p]=classify(testFeatures,trainFeatures,trainClass,'linear');
end

% Clasificador LDA2

if (strcmp(metodo,'LDA2'))

```

```

%Opciones del clasificador
options=[];
options.Regu = 1;
options.ReguAlpha = params;
[clase,p]=clasificacionLDA(trainFeatures, trainClass, testFeatures,options);
end

% Calsificador SVM
if (strcmp (metodo,'SVM'))
%Numero de divisiones k
addpath('SVM\libsvm-3.18\matlab')
k=17;
longitud=ceil(length(trainClass)/k);
[longitudTest,nada]=size(testFeatures);
claseaux=NaN(longitudTest,k);
p=NaN(longitudTest,1);
clase=p;
string=params;
N = size(trainFeatures,1);
parfor ii=1:k
    featuresTrain=(trainFeatures(longitud*(ii-1)+1:min(longitud*ii,N),:));
    claseTrain=(trainClass(longitud*(ii-1)+1:min(longitud*ii,N)));
    claseaux(:,ii)=clasificacionSVMparams(featuresTrain,claseTrain,testFeatures,string);
end

for ii=1:longitudTest
    cont=0;
    for a=1:k
        if(claseaux(ii,a)==1)
            cont=cont+1;
        end
    end
    p(ii)=cont/k;
    if(mean(claseaux(ii,:))>=0)
        clase(ii)=1;
    else
        clase(ii)=-1;
    end
end
end
end

```

### Fichero clasificacionSRKDA.m

```

function [clase,p]=clasificacionSRKDA(trainFeatures,testFeatures,trainClass,options)

% Clasificacion SRKDA

addpath('SRKDA')
%Entrenamiento
model = SRKDAtrain(trainFeatures, trainClass, options);
%Clasificacion
N = size(testFeatures,1);
[~, clase, p] = SRKDApredict(testFeatures, ones(N,1), model);

end

```

### Fichero clasificacionSVM.m

```

function [clase,p] = clasificacionSVM( trainFeatures, trainClass, testFeatures )

% Clasificacion SVM

model = svmtrain(trainClass,trainFeatures,'-g 1e-6 -c 30 -w1 11 -w-1 1 -q');
N = size(testFeatures,1);
clase = svmpredict(ones(N,1), testFeatures, model, '-q');

end

```

### Fichero clasificacionSVMParams.m

```
function [clase,p] = clasificacionSVMparams(trainFeatures,trainClass,testFeatures,string)

% Clasificacion SVM con parametros de entrada

    model = svmtrain(trainClass, trainFeatures, string);
    N = size(testFeatures,1);
    clase = svmpredict(ones(N,1), testFeatures, model, '-q');

end
```

### Fichero clasificacionTest.m

```
% Script de clasificacion principal

% Parametros

dir='../..\..\02mat';
metodoExt='ganador';
metodoClas='SVM';
sujetos={'Sujeto A', 'Sujeto B'};
tipoCurva='F';

Nsubjectos=size(sujetos,2);

for it=1:Nsubjectos

    fprintf('%s \n',sujetos{it})

    train=sprintf('%s\\%s\\train\\%s.mat',dir,metodoExt,sujetos{it});
    test=sprintf('%s\\%s\\test\\%s.mat',dir,metodoExt,sujetos{it});
    trainData=cargarDatos(train);
    testData=cargarDatos(test);

    %% Clasificacion

    noIllum=15; %Seleccionar numero de iluminaciones
    noIllum=((180)/15)*noIllum;
    noIllumtot=noIllum*100;
    testData.features=testData.features(:,1:noIllum,:);
    testData.class=testData.class(:,1:noIllum);
    testData.code=testData.code(:,1:noIllum);
    trainData.features=reshape(permute(trainData.features,[2 1 3]),15300,896);
    testData.features=reshape(permute(testData.features,[2 1 3]),noIllumtot,896);
    trainData.class=reshape(permute(trainData.class,[2 1]),15300,1);
    testData.class=reshape(permute(testData.class,[2 1]),noIllumtot,1);

    [p,clase] =clasificacion(trainData.features,testData.features,trainData.class,metodoClas);

    %% Letras

    testData.code=reshape(permute(testData.code,[2 1]),noIllumtot,1);
    [letras, caract]=determinarLetra(p,testData.code,testData.matrix);

    %% Evaluacion

    aciertos=posicionLetras(letras,testData.target,caract);

    %% Curva ROC

    [X,Y,T,AUC] = perfcurve(testData.class,p,1);
    figure(it)
    plot(X,Y)
    title('Curva ROC')
    drawnow

end
```

## Fichero crossvalmatrix.m

```
function matriz = crossvalmatrix( par1,par2,varargin )

% Funcion de creacion de matriz para variacion cruzada

    if nargin==2
        [n1,n2] = ndgrid(par2,par1);
        matriz=[n2(:) n1(:)];
    end
    if nargin==3
        [n1,n2,n3] = ndgrid(varargin{1},par2,par1);
        matriz=[n3(:) n2(:) n1(:)];
    end

end
```

## Fichero curvas.m

```
function [out_perf,X,Y] = curvas(class,p,type)

% Funcion para dibujar las curvas de distintos tipos

    if (strcmp(type,'ROC'))
        [X,Y,T,out_perf] = perfcurve(class,p,1);
        figure
        plot(X,Y)
        title('Curva ROC')
        drawnow
    end

    if(strcmp(type,'F'))
        [X,Y1,T1,AUC(1)] = perfcurve(class,p,1,'YCrit','TPR');
        figure
        plot(T1,Y1)
        title('Sensibilidad')
        [X,Y2,T2,AUC(2)] = perfcurve(class,p,1,'YCrit','PPV');
        figure
        plot(T2,Y2)
        title('Precision')
        F=(Y2.*Y1)./(Y2+Y1);
        figure
        plot(T2,F)
        title('F-Score')
        drawnow
        X = T2;
        Y = F;
        out_perf=max(F);
    end

end
```

## Fichero determinarLetra.m

```
function [letras,caract]=determinarLetra(p,code,matrix)

%Determinar letras

    %Seleccionar numero de iluminaciones

    noIllum=15;
    noIllum=(180)/15)*noIllum;
    p = reshape(p,noIllum,[]);
    [nada,caract]=size(p);
    for n=1:caract

        v=NaN(1,12);
        for i= 1:12 % Cada fila-columna (12*15 rep= 180 ilum)
```

```

        v(1,i)=mean(p(code(1+(n-1)*noIllum:noIllum+(n-1)*noIllum,1)==i,n));
        %En cada elemento de v guardamos la media de las probabilidades de
        % cada fila/columna
    end

    v1= repmat(v(1,1:6),6,1);
    v2= repmat(v(1,7:12)',1,6);

    V=(v1+v2)/2;

    valor=max(V(:));
    [x,y]=find(V==valor,1);

    letras(n)=matrix(x,y);

end

clear nada
end

```

### Fichero posicionLetras.m

```

function aciertos=posicionLetras(letras,target,caract)

% Funcion que indica las letras elegidas y porcentaje de aciertos

cont=0;

for n=1:caract

    %fprintf('\nLa letra elegida por el clasificador para el caracter es ...
    %c \nLa letra correcta era %c ',letras(n),target(n));

    if letras(n)==target(n)
        cont=cont+1;
    end

end

aciertos=((cont)/caract)*100;
fprintf('\n\nEl porcentaje de aciertos para 15 secuencias ha ...
        sido del %d%% \n\n',aciertos);

end

```

# Anexo B

## Pliego de condiciones

Este documento contiene las condiciones legales que guiarán la realización del proyecto “Diseño y aplicación de técnicas de análisis de electroencefalogramas para sistemas Brain Computer Interface basados en potenciales evocados”.

En lo que sigue se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de utilizar el sistema BCI. Dicha empresa ha debido desarrollar una línea de investigación con el objeto de llevar a cabo el proyecto. Esta línea de investigación, junto con el posterior desarrollo del programa, está amparada por las condiciones particulares del siguiente pliego. Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes condiciones:

- Condición 1** La modalidad de contratación será el concurso. La adjudicación se hará por tanto a la proposición más favorable, sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho de declararlo desierto.
- Condición 2** El montaje y la mecanización completa de los equipos que intervengan, será realizado totalmente por la empresa licitadora.
- Condición 3** En la oferta se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.
- Condición 4** La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos que sea preciso para el desarrollo de la misma.
- Condición 5** Aparte del Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no está obligado a aceptarla.
- Condición 6** El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autori-

zará con su firma las copias solicitadas por el contratista después de confrontarlas.

- Condición 7** Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le haya comunicado por escrito el Ingeniero Director de obras siempre que dicha obra se haya ejecutado a los preceptos de los pliegos de condiciones, de acuerdo a los cuales se harán modificaciones y la valoración de las diversas unidades, sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto no podrán servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.
- Condición 8** Tanto en las certificaciones de la obra como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.
- Condición 9** Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata, pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero considere justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.
- Condición 10** Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere, y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento se sujetarán siempre al establecido en el punto anterior.
- Condición 11** Cuando el contratista, con la autorización del Ingeniero Director de obras, emplee material de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio, ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sino a lo que le correspondería si se hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.
- Condición 12** Las cantidades calculadas para obras accesorias, aunque figuren por partidaalzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ella se formen, o en su defecto, por lo que resulte de su medición final.

- 
- Condición 13** El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras, así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.
- Condición 14** Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.
- Condición 15** La garantía definitiva será del 4 % del presupuesto y la provisional del 2 %.
- Condición 16** La forma de pago será por certificaciones mensuales de la obra ejecutada de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.
- Condición 17** La fecha de comienzo de las obras será a partir de los quince días naturales del replanteo oficial de las mismas, y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna a la reclamación de la fianza.
- Condición 18** Si el contratista, al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.
- Condición 19** El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras o con el delegado que este designe, para todo lo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.
- Condición 20** Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista la conservación de la obra ejecutada hasta la recepción de la misma, por lo que el deterioro total o parcial de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o construido por su cuenta.
- Condición 21** El contratista deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa por retraso en la ejecución siempre que este no sea debido a causas de fuerza mayor. A la terminación de la obra se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.
- Condición 22** Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las

mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas establecidas.

**Condición 23** Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad “Presupuesto de Ejecución por contrata”, y anteriormente llamado “Presupuesto de Ejecución Material” que hoy designa otro concepto.

La empresa constructora, que ha desarrollado este proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

**Condición particular 1** La propiedad intelectual de los procesos descritos y analizados en el presente trabajo pertenece por entero a la empresa constructora representada por el Ingeniero Director del Proyecto.

**Condición particular 2** La empresa constructora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos posteriores, para la misma empresa cliente o para otra.

**Condición particular 3** Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con la autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

**Condición particular 4** En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

**Condición particular 5** En todas las reproducciones se indicará su procedencia, explicando el nombre del proyecto, nombre del Ingeniero Director y empresa consultora.

**Condición particular 6** Si el proyecto pasa a la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de este, la empresa consultora decidirá aceptar o no la modificación propuesta.

**Condición particular 7** Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta al añadirla.

**Condición particular 8** Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

**Condición particular 9** Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

**Condición particular 10** La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

**Condición particular 11** La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia de este hecho. En este caso deberá autorizar expresamente los proyectos presentados por otros.

**Condición particular 12** El Ingeniero Director del proyecto será responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona asignada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.

Los requisitos que han sido necesarios para la elaboración del proyecto son:

### **Documentación**

- Servicio de Biblioteca de la Escuela Técnica Superior de Ingenieros de Telecomunicación de la Universidad de Valladolid.
- Acceso a Internet a través del laboratorio de Telecomunicaciones para la Sanidad de la Escuela Técnica Superior de Ingenieros de Telecomunicación de la Universidad de Valladolid.

### **Hardware**

- Ordenador:
  - Equipo Intel® Core™ i7-2600 CPU @ 3.40 GHz con 8 GB de RAM

### **Software**

- Señales EEG pertenecientes al conjunto de datos II de la III Competición BCI 2004.
- Matlab® 2012<sup>a</sup> (7.14.0.739)  
Copyright© 1984-2012, The MathWorks Inc.
- Microsoft® Windows 7 Professional Service Pack 1  
Copyright© 1981-2009, Microsoft Corporation
- Microsoft® Office Word 2010 (14.0.7116.5000)  
Copyright© 1983-2010, Microsoft Corporation
- Adobe® Acrobat® 8.1.0 Professional (7.1.0)  
Copyright© 1984-2006, Adobe Systems y sus concesionarios de licencia.
- Mozilla Firefox 29.0.1



# Anexo C

## Presupuesto

El presupuesto estimado de acuerdo con las condiciones expuestas en el pliego de condiciones es el siguiente:

### **EJECUCIÓN MATERIAL**

Equipo Intel® Core™ i7-2600 CPU @ 3.40 GHz con 8 GB de RAM .....	600 €
Software empleado .....	2.135 €
<hr/>	
TOTAL EJECUCIÓN MATERIAL .....	2.735 €

### **GASTOS GENERALES**

16 % sobre la ejecución material .....	438 €
--	-------

### **BENEFICIO INDUSTRIAL**

6 % sobre la ejecución material .....	164 €
---------------------------------------	-------

### **MATERIAL FUNGIBLE**

Gastos de impresión (tinta y papel) .....	15 €
Encuadernación .....	18 €
CD .....	4 €
<hr/>	
TOTAL MATERIAL FUNGIBLE .....	37 €

### **HONORARIOS DEL PROYECTO**

150 horas a 30 €/hora .....	4.500 €
-----------------------------	---------

**SUBTOTAL DEL PRESUPUESTO**

Subtotal del presupuesto ..... 7.874 €

**I.V.A. APLICABLE**

21 % sobre el subtotal del presupuesto ..... 1.653,54 €

**PRESUPUESTO TOTAL ..... 9.527,54 €**

El total del presupuesto, en euros, asciende a:

NUEVE MIL QUINIENTOS VEINTISIETE CON CINCUENTA Y CUATRO CÉNTIMOS DE EURO
---

En Valladolid, Julio de 2014.

Fdo. Pablo Núñez Novo.