

Trabajo Fin de Máster

Máster Universitario en Ingeniería de Telecomunicación

Detección automática de emociones mediante el análisis del electroencefalograma

Autor: Francisco Javier Ortiz Bonilla

Tutores: Rubén Martín Clemente, Sergio Antonio Cruces Álvarez

Departamento de Teoría de la Señal y Comunicaciones
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2019



Trabajo Fin de Máster
Máster Universitario en Ingeniería de Telecomunicación

Detección automática de emociones mediante el análisis del electroencefalograma

Autor:

Francisco Javier Ortiz Bonilla

Tutores:

Rubén Martín Clemente

Sergio Antonio Cruces Álvarez

Departamento de Teoría de la Señal y Comunicaciones

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2019

Autor: Francisco Javier Ortiz Bonilla

Tutores: Rubén Martín Clemente
Sergio Antonio Álvarez Cruces

El tribunal nombrado para juzgar el Trabajo arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2019

El Secretario del Tribunal

A mi familia

A mis amigos

A mis maestros

Agradecimientos

Con este trabajo concluyo mis estudios como Ingeniero de Telecomunicaciones, y si he llegado hasta aquí es sin duda gracias no sólo a mi esfuerzo si no también a la ayuda y esfuerzo de muchas más personas.

En primer lugar, tengo que agradecer a mis padres toda la dedicación volcada en mí durante estos años. Es por vosotros que estoy donde estoy. Sabed que el cariño y lo que me queréis es recíproco. Y ya que estamos y hay espacio, gracias también a mi abuelo, alias “El Manco” y “El cónsul de Triana”.

Gracias a también a mis amigos. Si me llevo algo bueno del Máster es sin duda el grupo de amigos que hemos formado, en el que siempre hemos estado cada uno para los demás y con los que he podido contar en todo momento durante el curso. Me alegro mucho de que nos hayamos conocido y de la amistad que hemos entablado.

También me gustaría agradecer a los profesores del Máster su labor. Aunque no esté de acuerdo con cómo se enfocan algunas asignaturas o las metodologías aplicadas, en general los profesores han sido correctos y de cada uno de ellos hemos aprendido algo. En particular, me gustaría agradecer al coordinador Juan José Murillo que siempre haya estado ahí para nosotros y su preocupación por que recibamos contenido útil durante el Máster.

Por último, agradecer a mis tutores Sergio y Rubén toda la atención prestada y que me hayan guiado durante el proyecto. He aprendido mucho.

Muchas gracias a todos.

Francisco Javier Ortiz Bonilla

Sevilla, 2019

Resumen

Las emociones son un importante factor en la vida humana y conforman un aspecto de suma relevancia en la comunicación e interacción interpersonal, y hoy día, en un mundo hiper conectado en el que la sociedad es cada vez más dependiente de la tecnología, la mayoría de los sistemas de interacción persona-ordenador son aún deficientes a la hora de identificar los estados emocionales de las personas.

La computación afectiva es una rama del estudio y desarrollo de la inteligencia artificial que hace referencia al diseño de sistemas y dispositivos que pueden reconocer, interpretar y procesar emociones humanas. Que los ordenadores y otros dispositivos sean capaces de detectar e identificar cómo se sienten las personas abre la puerta a una cantidad incontable de aplicaciones y supondría una mejora de la calidad de vida para la sociedad.

En el presente trabajo se ha estudiado el reconocimiento de emociones mediante el electroencefalograma. Se ha hecho una revisión del estado del arte en esta materia y de técnicas de aprendizaje máquina para posteriormente abordar los problemas de la extracción de características de las señales de encefalograma y la clasificación de emociones. Esta última se ha dividido en dos problemas de clasificación binaria: por un lado se clasifica si la emoción es positiva o negativa (clasificación de la valencia) y por otro si la emoción supone una excitación alta o baja (clasificación del grado de excitación).

Abstract

Emotions play a big role in human life and constitute an extremely important aspect in interpersonal communication and interaction, and nowadays, in a hyper-connected world in which society is increasingly dependent on technology, most of human-computer interaction systems are still deficient when it comes to identifying the emotional states of people.

Affective computing is the study and development of systems and devices that can recognize, interpret, process, and simulate human affects. Computers and other devices being able to detect and identify how people feel would open the door to an uncountable number of applications and would imply an improvement in the quality of life for society.

In this project the emotion recognition through the electroencephalogram has been studied. Reviews of the state of the art in this subject and machine learning techniques have been carried out to later address the problems of the extraction of characteristics from encephalogram signals and the classification of emotions. The latter has been divided into two problems of binary classification: on the one hand it is classified if the emotion is positive or negative (valence classification) and on the other hand if the emotion implies high or low excitation (arousal classification).

Índice

Agradecimientos	ix
Resumen	xi
Abstract	xiii
Índice	xv
Índice de Tablas	xvii
Índice de Figuras	xix
1 Introducción	1
1.1. <i>Motivación</i>	1
1.2. <i>Electroencefalografía</i>	2
1.2.1. <i>Bandas cerebrales</i>	3
1.3. <i>Estructura de la memoria</i>	4
2 Las emociones y el cerebro	5
2.1 <i>Qué son las emociones</i>	5
2.2 <i>Clasificación de las emociones</i>	6
2.2.1. <i>Modelos discretos</i>	6
2.2.2. <i>Modelos dimensionales</i>	6
2.3. <i>Las emociones en el cerebro</i>	8
2.3.1 <i>Teoría del sistema límbico</i>	8
2.4 <i>Conclusiones</i>	8
3 Estado del arte en el reconocimiento de emociones	9
3.1 <i>Modo de reconocimiento: en vivo o en diferido</i>	9
3.2 <i>Incitación de emociones</i>	9
3.3 <i>Reconocimiento de emociones mediante electroencefalograma</i>	9
3.3.1 <i>Obtención de señales EEG</i>	9
3.3.2 <i>Preprocesado de las señales EEG</i>	11
3.3.3 <i>Extracción de características</i>	11
3.3.4 <i>Entrenamiento y validación de un clasificador</i>	14
3.4 <i>Resultados previos en la literatura para la clasificación de emociones</i>	14
3.5 <i>Conclusiones</i>	15
4 Base de datos utilizada: DEAP Dataset	17
4.1 <i>Elección de los estímulos</i>	17
4.2 <i>Experimentos</i>	18
4.3 <i>Uso de la base de datos</i>	19
4.4 <i>Conclusiones</i>	20
5 Técnicas utilizadas	21
5.1 <i>Elección de características</i>	21
5.1.1 <i>Conjuntos de características 1 y 4</i>	22
5.1.2 <i>Conjuntos de características 3 y 6</i>	23

5.1.3	Conjuntos de características 2 y 5	25
5.2	<i>Técnicas para la extracción de características</i>	27
5.2.1	Filtrado de las señales EEG en cada banda	27
5.2.2	Cálculo de las potencias por banda y potencia total en cada electrodo	30
5.2.3	Cálculo de las características a partir de las potencias	30
5.2.4	Exportación de las características	31
5.3	<i>Implementación de la extracción de las características elegidas</i>	31
5.3.1	Script <i>características.m</i>	31
5.4	<i>Técnicas utilizadas para la clasificación</i>	36
5.4.1	Análisis Discriminante Lineal (LDA)	36
5.4.2	Árbol de decisión y Bosques aleatorios (Random Forest)	39
5.4.3	K vecinos más cercanos (KNN)	40
5.4.4	Máquinas de soporte vectorial (SVM)	41
5.4.5	Redes neuronales artificiales (ANN)	42
5.4.6	Análisis de Componentes Principales (PCA)	44
5.5	<i>Conclusiones</i>	45
6	Simulaciones	47
6.1	<i>Clasificación mediante KNN</i>	48
6.1.1	Clasificación de la valencia	48
6.1.2	Clasificación del grado de excitación	50
6.2	<i>Clasificación mediante Random Forest</i>	52
6.2.1	Clasificación de la valencia	52
6.2.2	Clasificación del grado de excitación	54
6.3	<i>Clasificación mediante SVM</i>	56
6.3.1	Clasificación de la valencia	57
6.3.2	Clasificación del grado de excitación	61
6.4	<i>Otros clasificadores</i>	65
6.5	<i>Conclusiones</i>	68
7	Conclusiones y líneas futuras	71
	Referencias	73
	Glosario	77

ÍNDICE DE TABLAS

Tabla 1: Dispositivos de electroencefalografía disponibles en el mercado.	10
Tabla 2: Electrodo en cada banda que muestran una mayor correlación con el valor de la valencia.	13
Tabla 3: Electrodo en cada banda que muestran una mayor correlación con el valor del grado de excitación.	14
Tabla 4: Ficheros disponibles en la base de datos DEAP dataset.	19
Tabla 5: Conjuntos de características extraídos para la clasificación de la valencia y grado de excitación.	21
Tabla 6: Pares de electrodo simétricos en las muestras del DEAP dataset.	22
Tabla 7: Características del conjunto 3.	23
Tabla 8: Características del conjunto 6.	24
Tabla 9: Características del conjunto 2.	25
Tabla 10: Características del conjunto 5.	26
Tabla 11: Parámetros configurables del script características.m.	32
Tabla 12: Inventariados utilizados en las simulaciones.	47
Tabla 13: Capas de la red neuronal implementada.	66

ÍNDICE DE FIGURAS

Figura 1: Mujer durante un electroencefalograma. Fuente: [8].	2
Figura 2: Electroencefalograma con diez canales. Fuente: [9].	3
Figura 3: Estructura en árbol propuesta por Parrot para la clasificación de emociones. Fuente [19].	6
Figura 4: Rueda de Plutchik. Fuente: [16].	7
Figura 5: Modelo de Russell.	7
Figura 6: Electroodos utilizados en el sistema 10-20. Fuente: [23].	11
Figura 7: Correlación entre las diferentes bandas y electrodos y el valor de la valencia. Fuente: [40].	12
Figura 8: Correlación entre las diferentes bandas y electrodos y el valor del grado de excitación. Fuente: [40].	13
Figura 9: Captura de pantalla de la interfaz web usada para la autoevaluación online. Fuente: [40].	18
Figura 10: Electroodos (amarillo) utilizados en los experimentos del DEAP dataset.	19
Figura 11: Respuesta en frecuencia del filtro para la banda <i>theta</i> .	27
Figura 12: Respuesta en frecuencia del filtro para la banda <i>slow alpha</i> .	28
Figura 13: Respuesta en frecuencia del filtro para la banda <i>alpha</i> .	28
Figura 14: Respuesta en frecuencia del filtro para la banda <i>beta</i> .	28
Figura 15: Respuesta en frecuencia del filtro para la banda <i>gamma</i> .	29
Figura 16: Señal EEG y la misma señal EEG filtrada en la beta, ambas en el tiempo.	29
Figura 17: Señal EEG y la misma señal EEG filtrada en la beta, ambas en frecuencia.	30
Figura 18: Separación del conjunto de datos usando LDA. Fuente: [44].	38
Figura 19: Pseudocódigo para la implementación de LDA. Fuente: [46].	39
Figura 20: Ejemplo de árbol de decisión. Fuente: [47].	39
Figura 21: División del espacio de características en subespacios correspondientes a cada clase con KNN. Fuente: [50].	41
Figura 22: Máquinas de soporte vectorial: Vectores de soporte, hiperplano óptimo y margen máximo. Fuente: [52].	42
Figura 23: Transformación del conjunto de datos mediante un kernel. Fuente: [53].	42
Figura 24: Partes de una neurona. Fuente [54].	43
Figura 25: Red neuronal con 4 capas. Fuente: [56].	43
Figura 26: Modelo computacional de una neurona. Fuente: [57].	44
Figura 27: Análisis de las diferentes ventanas para la clasificación de la valencia usando KNN.	48
Figura 28: Análisis del parámetro K en la clasificación de la valencia con KNN.	49
Figura 29: Análisis de la reducción del espacio de características con PCA en la clasificación de la valencia con KNN.	49
Figura 30: Análisis de outliers en la clasificación de la valencia con KNN.	50

Figura 31: Análisis de las diferentes ventanas para la clasificación del grado de excitación usando KNN.	50
Figura 32: Análisis del parámetro K en la clasificación del grado de excitación con KNN.	51
Figura 33: Análisis de la reducción del espacio de características con PCA en la clasificación del grado de excitación con KNN.	51
Figura 34: Análisis de outliers en la clasificación del grado de excitación con KNN.	52
Figura 35: Análisis de las diferentes ventanas para la clasificación de la valencia usando Random Forest.	52
Figura 36: Análisis del parámetro K en la clasificación de la valencia con Random Forest.	53
Figura 37: Análisis de la reducción del espacio de características con PCA en la clasificación de la valencia con Random Forest.	53
Figura 38: Análisis de outliers en la clasificación de la valencia con Random Forest.	54
Figura 39: Análisis de las diferentes ventanas para la clasificación del grado de excitación usando Random Forest.	54
Figura 40: Análisis del parámetro K en la clasificación del grado de excitación con Random Forest.	55
Figura 41: Análisis de la reducción del espacio de características con PCA en la clasificación del grado de excitación con Random Forest.	55
Figura 42: Análisis de outliers en la clasificación del grado de excitación con Random Forest.	56
Figura 43: Kernels para SVM utilizados en las simulaciones.	56
Figura 44: Análisis de las diferentes ventanas para la clasificación de la valencia con el conjunto de características 1 usando SVM.	57
Figura 45: Análisis de las diferentes ventanas para la clasificación de la valencia con el conjunto de características 2 usando SVM.	57
Figura 46: Análisis de las diferentes ventanas para la clasificación de la valencia con el conjunto de características 3 usando SVM.	58
Figura 47: Análisis del parámetro C en la clasificación de la valencia con SVM con kernel lineal.	58
Figura 48: Análisis del parámetro C en la clasificación de la valencia con SVM con kernel polinómico.	59
Figura 49: Análisis del parámetro C en la clasificación de la valencia con SVM con kernel gaussiano.	59
Figura 50: Análisis del parámetro C en la clasificación de la valencia con SVM con kernel sigmoide.	59
Figura 51: Resultados obtenidos para la clasificación de valencia con los diferentes kernels para SVM.	60
Figura 52: Análisis de la reducción del espacio de características con PCA en la clasificación de la valencia con SVM.	60
Figura 53: Análisis de outliers en la clasificación de la valencia con SVM.	61
Figura 54: Análisis de las diferentes ventanas para la clasificación del grado de excitación con el conjunto de características 4 usando SVM.	61
Figura 55: Análisis de las diferentes ventanas para la clasificación del grado de excitación con el conjunto de características 5 usando SVM.	62
Figura 56: Análisis de las diferentes ventanas para la clasificación del grado de excitación con el conjunto de características 6 usando SVM.	62
Figura 57: Análisis del parámetro C en la clasificación del grado de excitación con SVM con kernel lineal.	63
Figura 58: Análisis del parámetro C en la clasificación del grado de excitación con SVM con kernel polinómico.	63
Figura 59: Análisis del parámetro C en la clasificación del grado de excitación con SVM con kernel gaussiano.	63

Figura 60: Análisis del parámetro C en la clasificación del grado de excitación con SVM con kernel sigmoide.	64
Figura 61: Resultados obtenidos para la clasificación del grado de excitación con los diferentes kernels para SVM.	64
Figura 62: Análisis de la reducción del espacio de características con PCA en la clasificación del grado de excitación con SVM.	64
Figura 63: Análisis de outliers en la clasificación del grado de excitación con SVM.	65
Figura 64: Pseudocódigo para la implementación de LDA. Fuente: [46].	66
Figura 65: Tasa de acierto en función del número de épocas para la clasificación de la valencia con ANN.	67
Figura 66: Tasa de acierto en función del número de épocas para la clasificación del grado de exictación con ANN.	67
Figura 67: Tasas de acierto obtenidas con los diferentes clasificadores.	68
Figura 68: Modelo de Russell.	71

1 INTRODUCCIÓN

Las emociones desempeñan un papel importante en la comunicación humana y pueden expresarse verbalmente a través de vocabulario explícito, o de forma no verbal mediante la entonación de la voz, expresiones faciales y gestos.

Sin embargo, la mayoría de los sistemas actuales de interacción persona-ordenador (HCI, Human Computer Interaction) son deficientes a la hora de interpretar esta información, pues carecen de inteligencia emocional. En otras palabras, no pueden identificar los estados afectivos de las personas y utilizar esta información para decidir qué acciones son las más apropiadas para ejecutar. El objetivo de la computación afectiva es suplir estas carencias mediante la detección de señales que indiquen los estados emocionales de las personas durante la interacción persona-ordenador y la síntesis de respuestas emocionales.

En teoría, todo lo que alguien hace o dice tiene su origen en su cerebro. La actividad cerebral produce señales que son detectables sobre el cuero cabelludo, en la superficie del cerebro y dentro del mismo. Al usar esa actividad para la interacción persona-ordenador se dispondría de una gran cantidad de información muy valiosa, y las posibilidades parecen infinitas.

Las interfaces cerebro-ordenador (BCI, Brain Computer Interface) procesan estas señales producidas por la actividad cerebral y permiten una interacción persona-ordenador basada en la actividad cerebral. Se han construido interfaces cerebro-ordenador para jugar a videojuegos [1,2], para dispositivos de escritura [2,3] o incluso para controlar un ordenador con Windows [4], pero la lista de posibles aplicaciones que podrían ser controladas por una interfaz BCI incluye en realidad a todas las que se conocen en la actualidad.

Usar una interfaz BCI para conocer el estado emocional podría mejorar la calidad y la eficacia de las interacciones persona-ordenador, además de que abre nuevas formas para que los ordenadores entiendan el comportamiento o significado humano.

El objetivo de este trabajo es conseguir conocer el estado emocional de una persona a partir de una serie de señales de encefalograma y mediante la utilización de técnicas de machine learning. En concreto, se buscará determinar por una parte si la persona está experimentando una emoción positiva o negativa (valencia), y por otra parte el grado de activación o excitación (alta o baja).

1.1. Motivación

Hoy día, las nuevas formas de interacción entre las personas y el contenido digital tienen el potencial de revolucionar el entretenimiento, el aprendizaje y muchas otras áreas de nuestra vida. Dado que las emociones desempeñan un papel importante en la vida diaria de los seres humanos, la necesidad y la importancia del reconocimiento automático de emociones se ha incrementado con el creciente papel de las aplicaciones basadas en la interacción persona-ordenador.

Se han hecho muchos esfuerzos para reconocer emociones mediante otras modalidades como el habla, el texto o las expresiones faciales. Se espera que el reconocimiento de emociones a partir de las señales del EEG sea superior a estas, ya que la actividad cerebral tiene información directa sobre la emoción mientras que las otras modalidades no son más que un reflejo indirecto de la emoción. Además, las señales de EEG se pueden medir en cualquier momento y no dependen de que alguien hable o genere una expresión facial.

Tradicionalmente, la tecnología basada en EEG se ha utilizado en aplicaciones médicas. En la actualidad, están llegando al mercado nuevos dispositivos de electroencefalografía inalámbricos que cumplen con los criterios del consumidor en cuanto a usabilidad, precio, portabilidad y facilidad de uso, posibilitando la difusión de la tecnología a aplicaciones en el reconocimiento de emociones en diferentes áreas como entretenimiento,

aprendizaje electrónico, mundos virtuales, mundos cibernéticos, etc. El reconocimiento automático de emociones de las señales del EEG está recibiendo más atención con el desarrollo de nuevas formas de interacción entre las personas y el contenido digital.

Vemos que las posibles aplicaciones de esta técnica son muy variadas. En primer lugar, la simple tarea de reconocer la emoción de alguien automáticamente podría ayudar a los terapeutas y psicólogos a hacer su trabajo. Otras aplicaciones se pueden encontrar en el campo de la interacción persona-ordenador, la comunicación entre personas a través de ordenadores y para ayudar a personas discapacitadas a comunicarse.

En el campo de la interacción persona-ordenador, el ordenador podría ajustar su comportamiento en función del estado emocional de la persona: cuando un juego se vuelve aburrido o demasiado excitante, se podría cambiar su nivel para ajustar estas emociones. Otro ejemplo sería que el ordenador cambiase la música o el fondo de las ventanas dependiendo del estado de humor de la persona sentada frente a él. Estos tipos de interacción persona-ordenador existen ya en el mundo comercial [1, 5, 6].

La comunicación a través de mensajería instantánea es muy común hoy día. Desafortunadamente, esta comunicación se realiza únicamente a través de palabras, aunque las notas de voz y el vídeo son grandes añadidos. Así mismo, la enorme cantidad de emoticonos que se usa en las conversaciones muestra el deseo que tenemos de compartir y expresar nuestras emociones al hablar con los demás. En este aspecto, una interfaz BCI podría ser una herramienta útil para incluir información emocional de forma automática durante la conversación.

Otra aplicación sería hacer posible expresar sus emociones a personas que padecen enfermedades musculares graves como la esclerosis lateral amiotrófica (ALS). Es posible que estas personas no puedan mover sus músculos o hablar. Se han hecho avances en la creación de una interfaz cerebro-ordenador para permitir a estos pacientes comunicarse nuevamente, y añadir una forma para que puedan expresar sus emociones sería una gran mejora para dicho sistema [7].

1.2. Electroencefalografía

La electroencefalografía consiste en el registro de la actividad eléctrica que tiene lugar en el cerebro a través de electrodos adheridos sobre el cuero cabelludo.

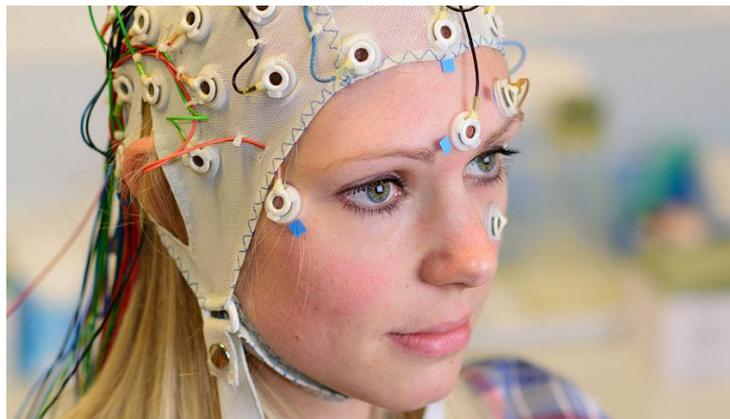


Figura 1: Mujer durante un electroencefalograma. Fuente: [8].

Por cada electrodo usado durante el registro del electroencefalograma, tendremos una señal EEG distinta pero correspondiente al mismo electroencefalograma. Comúnmente se habla de canales a la hora de referirse a los electrodos: cada electrodo es un canal.

La figura de a continuación ilustra un electroencefalograma recogido usando diez canales. Cada una de las señales corresponde a un electrodo diferente.

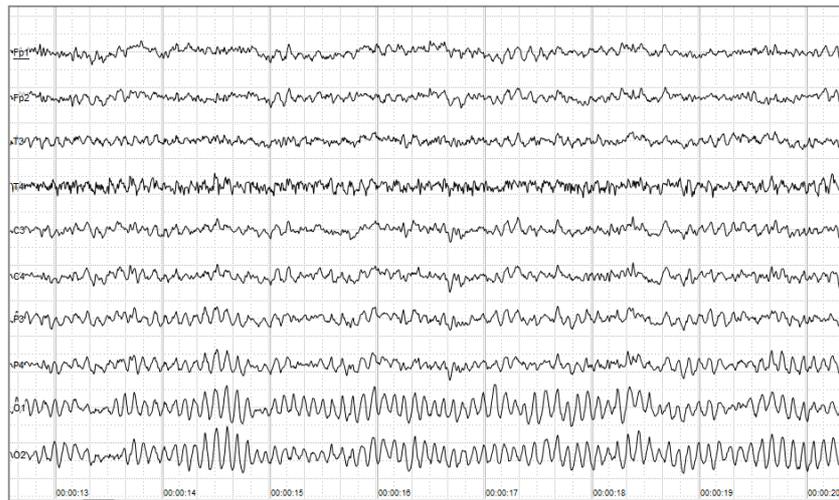


Figura 2: Electroencefalograma con diez canales. Fuente: [9].

Sin embargo, este conjunto de señales EEG representadas en el tiempo no parece decirnos mucho a priori. Veremos más adelante que la información de mayor relevancia la obtendremos observando en qué zonas del cerebro la actividad es mayor o menor, la diferencia de actividad entre zonas simétricas del cerebro (en cada hemisferio) y la actividad en las diferentes bandas cerebrales.

1.2.1. Bandas cerebrales

Los impulsos eléctricos producidos por el cerebro y los cuales registramos en señales EEG pueden corresponder a diferentes ritmos cerebrales, también conocidos como ondas cerebrales. Existen principalmente cinco tipos de ondas y bandas cerebrales distinguidas por sus rangos de frecuencia [10]:

- **Banda *delta*: 0.5 – 4 Hz.**
Las ondas *delta* son las de mayor amplitud y menor frecuencia. Se asocian a un estado de sueño profundo, y al ser lentas pueden ser confundidas con artefactos debidos a movimiento, aunque analizando las señales es relativamente fácil distinguir si se tratan de artefactos o de ondas *delta*.
- **Banda *theta*: 4 – 8 Hz.**
Las ondas *theta* tienen una amplitud grande aunque menor que la de las ondas *delta*. Se asocian a estados de meditación, lucidez y creatividad, y también a procesos en los que interviene la memoria [11].
- **Banda *alpha*: 8 – 12 Hz.**
Las ondas *alpha* están asociadas a estados de relajación, calma y pensamiento abstracto, y la ausencia de estas también se asocia con estados de concentración o ansiedad. Su origen fisiológico es desconocido, aunque son las más investigadas.
- **Banda *beta*: 12 – 30 Hz.**
Las ondas *beta* son las que predominan en un estado normal y despierto, en el que el sujeto está alerta y activo mentalmente. Se asocian a estados de actividad mental intensa, concentración relajada, agitación y ansiedad.
- **Banda *gamma*: 30 – 45 Hz.**
Aunque no se ha llegado a resultados concluyentes en cuanto al significado de las ondas *gamma*, se ha sugerido que podrían estar implicadas en el proceso de percepción consciente [12].

1.3. Estructura de la memoria

La memoria del presente trabajo se ha organizado en los siguientes capítulos:

- **Capítulo 1: Introducción**
En este capítulo se ha introducido el tema del trabajo y se han descrito las motivaciones que dan pie a su realización.
- **Capítulo 2: Las emociones y el cerebro**
Este capítulo trata sobre las emociones: qué son, qué modelos se han propuesto y existen para clasificarlas y cómo estas se manifiestan en el cerebro.
- **Capítulo 3: Estado del arte**
En este capítulo se analizan las investigaciones, resultados y avances con respecto al reconocimiento de emociones mediante el análisis del electroencefalograma.
- **Capítulo 4: Base de datos utilizada**
En este capítulo se habla sobre la base de datos que se ha utilizado para llevar a cabo este trabajo, llamada DEAP dataset.
- **Capítulo 5: Técnicas utilizadas**
En este capítulo se exponen las técnicas utilizadas para la extracción de características a partir de señales EEG y las técnicas de machine learning empleadas para la posterior clasificación.
- **Capítulo 6: Simulaciones**
En este capítulo se describen las simulaciones que se han llevado a cabo y los resultados obtenidos.
- **Capítulo 7: Conclusiones**
En este capítulo exponen las conclusiones obtenidas tras la realización del trabajo y las líneas para la continuación del mismo.

2 LAS EMOCIONES Y EL CEREBRO

En este capítulo se va a hablar acerca de las emociones y la relevancia que tienen en la vida humana y en la sociedad. Además, se hablará de los diferentes modelos propuestos para clasificar las emociones y de las diferentes teorías acerca de cómo se manifiestan o tienen lugar en el cerebro, entre las cuales destaca la teoría del sistema límbico.

2.1 Qué son las emociones

Las emociones son estados afectivos e indican estados internos personales: sentimientos, motivaciones, deseos e incluso objetivos. Apenas tenemos unos meses de vida, adquirimos emociones básicas como el miedo, el enfado o la alegría, las cuales compartimos con algunos animales. No obstante, a diferencia de estos, estas emociones básicas en los humanos se van haciendo más complejas debido en gran parte al lenguaje: usamos símbolos, signos y significados.

Si bien estas son intuitivamente conocidas por todos, definir qué es una emoción no es una tarea trivial. El filósofo griego Aristóteles hablaba de las emociones como un estímulo que evalúa experiencias en base al potencial para sentir dolor o placer. Muchos años más tarde, en el siglo diecisiete, Descartes consideró a las emociones algo intermedio entre estímulo y respuesta [14].

A día de hoy aún existe poco consenso sobre la definición de emoción. Paul R. Kleinginna y Anne M. Kleinginna recogieron y analizaron 92 definiciones de emoción a partir de la literatura presente por entonces. Concluyeron que había poca consistencia entre las diferentes definiciones y sugirieron la siguiente [15]:

Una emoción es un conjunto de interacciones complejo entre factores subjetivos y objetivos, mediados por sistemas neuronales y hormonales, que puede:

1. dar lugar a experiencias afectivas como sentimientos de excitación, placer o desagrado;
2. generar procesos cognitivos como efectos perceptivos emocionalmente relevantes, opiniones, procesos de etiquetado;
3. activar ajustamientos psicológicos a las condiciones excitantes; y
4. conducir a un comportamiento que es, frecuentemente pero no siempre, expresivo, orientado a una meta y adaptable.

Esta definición muestra las dos caras de lo que es una emoción: por un lado, una emoción genera sentimientos específicos en una persona e influencia su comportamiento; por otro lado, una emoción también ajusta el estado cerebral de una persona, y directa o indirectamente influencia distintos procesos.

Las emociones son un importante factor en la vida humana y conforman un aspecto de suma relevancia en la comunicación e interacción interpersonal. El estado de humor de una persona influencia enormemente su manera de comunicarse, pero también su forma de actuar y su productividad. Imagine dos conductores de coche, uno feliz y otro realmente enrabiado: la forma en la que conducirán será totalmente diferente. Las emociones también juegan un papel crucial en la comunicación cotidiana. Una persona puede decir una palabra como “Claro” de forma agradable pero también con decepción o sarcasmo. En la mayoría de las comunicaciones, interpretamos estos significados mediante el tono de voz o a partir de comunicación no

verbal.

2.2 Clasificación de las emociones

Existen varios modelos para clasificar y representar las emociones. Podemos agruparlos en dos tipos: modelos discretos y modelos dimensionales.

2.2.1. Modelos discretos

Estos modelos proponen categorizaciones discretas de las emociones. Aquí podemos encontrar:

- **Modelo de Ekman y Friesen**

Afirma que existen seis emociones básicas: ira, miedo, tristeza, felicidad, disgusto y sorpresa [16].

- **Modelo de Parrot**

Propone la siguiente estructura en árbol [16].

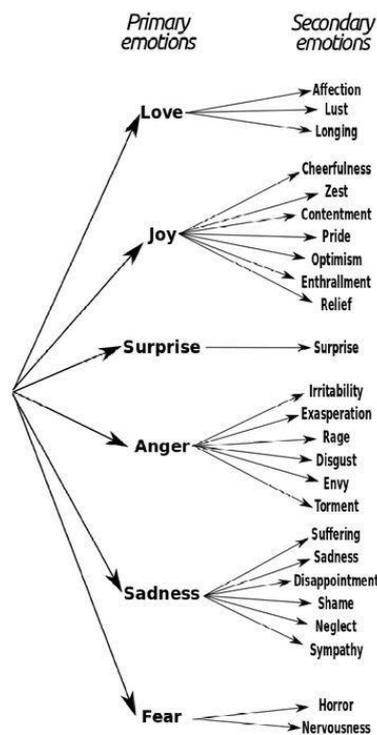


Figura 3: Estructura en árbol propuesta por Parrot para la clasificación de emociones. Fuente [19].

2.2.2. Modelos dimensionales

Estos modelos se componen de múltiples dimensiones, y sitúan cada emoción en una escala multidimensional. Destacan los siguientes:

- **Modelo de Plutchik**

Agrupar las emociones en ocho categorías primarias, que tienen funciones específicas para ayudar a la supervivencia: temor, sorpresa, tristeza, disgusto, ira, esperanza, alegría y aceptación. Las demás emociones serían combinaciones de las ocho anteriores [17].

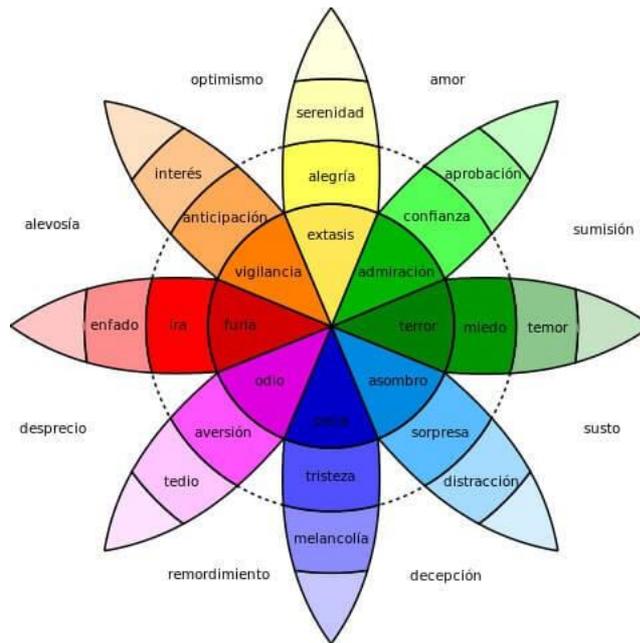


Figura 4: Rueda de Plutchik. Fuente: [16].

- **Modelo de Russell**

Sugiere que las emociones están distribuidas en un espacio circular de dos dimensiones: excitación y valencia. La excitación representa el eje vertical y la valencia el eje horizontal. El centro del círculo representa una valencia neutral y un nivel de excitación medio: se habla de valencia negativa o positiva y de excitación baja o alta [13].

- Valencia: Varía de emociones negativas (como tristeza, enfado) a emociones positivas (como felicidad, euforia).
- Excitación: Varía de emociones de excitación baja (calma, depresión) a emociones de excitación alta (ira, euforia).

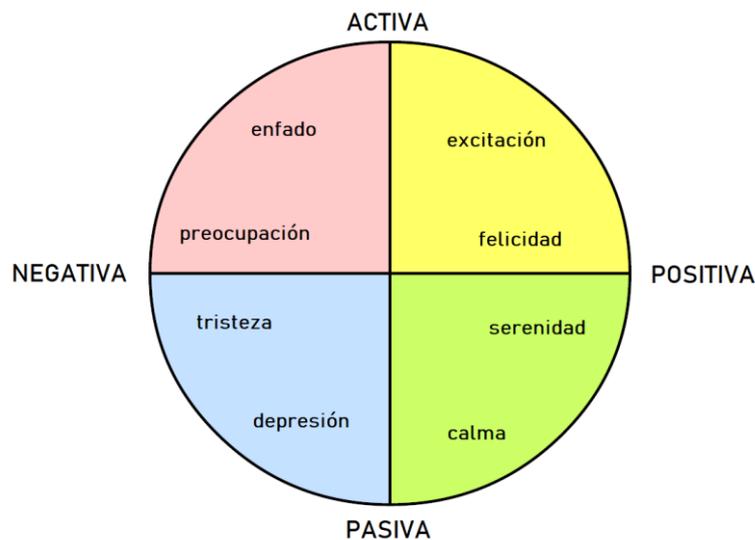


Figura 5: Modelo de Russell.

El modelo de Russell es el más extendido y usado en la mayoría de estudios por dos razones:

1. Simplicidad: Es sencillo expresar una emoción en términos de valencia y excitación.
2. Universalidad: Valencia y excitación son términos comprendidos universalmente y no hay controversia a la hora de usarlos como escala para clasificar emociones.

Además, es el que se ha utilizado en este trabajo.

2.3. Las emociones en el cerebro

Durante años, científicos han razonado e investigado sobre las emociones. Una de las primeras teorías propuestas fue la de James-Lange, a finales del siglo diecinueve. Esta teoría afirmaba que una emoción era la experiencia de cambios en el cuerpo. Por ejemplo: cuando alguien ve un oso, saldrá corriendo, y el hecho de salir corriendo le producirá un sentimiento de miedo.

En la década de 1920, Cannon y Bard cuestionaron esta teoría ya que los cambios en el cuerpo son demasiado lentos como para ser responsables de las emociones, y cambios artificiales en el cuerpo son insuficientes para generar emociones. Afirmaron que el hipotálamo era la región del cerebro que traducía de estímulos a emociones.

Años después, Papez partió de esta idea para desarrollar un esquema de cómo las emociones se procesan en el cerebro. Este esquema consta de dos flujos, correspondiente a la acción de sentir y otro a la de pensar. Según él, las experiencias afectivas son una función de la actividad de ambos flujos, que se calcula en el giro cingulado. Gran parte de esta teoría, en concreto las conexiones que propuso, se ha probado que existen, aunque no todas las regiones que menciona son importantes en el procesamiento de las emociones [13].

2.3.1 Teoría del sistema límbico

En 1940, Maclean introdujo otro modelo más preciso, e inventó el término sistema límbico. Él sostenía que la percepción de lo que nos rodea, combinada con la información de los cambios corporales, generaba experiencias afectivas, y esta integración de conocimiento ocurría en el sistema límbico.

Esta teoría del sistema límbico ha reinado desde entonces. El hipotálamo es responsable de procesar las señales entrantes y activar los efectos fisiológicos correspondientes, como un aumento en las pulsaciones del corazón [18].

A partir del hipotálamo, la información del estímulo se pasa al cuerpo amigdalino, que es importante para aprender a conectar estímulos a reacciones emocionales y para evaluar nuevos estímulos comparándolos con experiencias pasadas [18].

El cuerpo amigdalino se considera fundamental para el procesamiento de las emociones. Sin embargo, no puede detectarse directamente a partir de grabaciones desde la corteza cerebral. El cuerpo amigdalino se conecta a los córtex temporal y prefrontal [13].

El lóbulo temporal es fundamental para escuchar, el lenguaje y las emociones, y también juega un papel importante en la memoria. El lóbulo prefrontal es responsable de procesos cognitivos, emocionales y motivacionales, además forma parte del córtex frontal, el cual se cree que es el centro del control emocional y lo que determina la personalidad.

2.4 Conclusiones

En este capítulo se ha hablado acerca de qué son las emociones y la importancia que tienen, de los diferentes modelos propuestos para clasificarlas, de los cuales en el presente trabajo se utilizará el de Russell; y de las diferentes teorías acerca de cómo se manifiestan o tienen lugar en el cerebro, entre las cuales destaca la teoría del sistema límbico.

3 ESTADO DEL ARTE EN EL RECONOCIMIENTO DE EMOCIONES

Los avances en las tecnologías BCI (Interfaz Cerebro Ordenador) han impulsado el estudio e investigación en el reconocimiento y clasificación de emociones. Existe un gran número de estudios en los que se investiga la detección y reconocimiento de emociones y los ámbitos en los que se aplican son muy variados: comunicación, educación, entretenimiento, marketing, medicina...

Con el fin de entender las tendencias en la investigación de la detección de emociones mediante encefalograma, Al-Nafjan et al. revisaron en [20] 285 artículos relativos a este tema.

3.1 Modo de reconocimiento: en vivo o en diferido

El análisis y clasificación de las señales EEG puede ser llevado a cabo en vivo o en diferido, aunque la forma más extendida es hacerlo en diferido. En [24], los autores construyeron un sistema que detectaba el estado afectivo del paciente actual y obtuvieron una tasa de acierto de 65%.

3.2 Incitación de emociones

Se han desarrollado y reportado diferentes técnicas para la incitación de emociones: presentación de estímulos (imágenes, vídeo y audios), técnicas de imaginación (imaginación guiada y evocación de recuerdos), a través de interacción social presencial (por ejemplo, con juegos) y mediante acciones faciales. En [20] se concluye que el uso de múltiples estímulos es más efectivo a la hora de inducir emociones que el uso de un único estímulo.

A la hora de emplear estas técnicas para inducir ciertas emociones, se usan estímulos extraídos de fuentes reconocidas como la base de datos de imágenes IAPS (International Affective Picture System) o la base de datos de sonidos IADS (International Affective Digitized Sounds). También han sido usadas por parte de investigadores otras técnicas como la evocación de recuerdos o la interacción con una persona facilitadora que ayuda a la incitación de determinadas emociones.

3.3 Reconocimiento de emociones mediante electroencefalograma

El reconocimiento de emociones mediante encefalograma es una labor que se compone de diferentes tareas o etapas. En primer lugar, es necesario realizar un pre-procesado sobre las señales EEG disponibles; en segundo lugar, a partir de dichas señales EEG se deberá extraer una serie de características que se crean útiles para, por último, llevar a cabo el entrenamiento de un clasificador.

3.3.1 Obtención de señales EEG

Como se ha explicado en la sección 1.2 del capítulo 1, mediante la electroencefalografía se registra la actividad eléctrica que tiene lugar en el cerebro a través de electrodos adheridos sobre el cuero cabelludo.

3.3.1.1 Dispositivos de electroencefalografía

Para ello, existen diferentes dispositivos en el mercado. En [20] encontraron hasta 48 dispositivos de diferentes compañías en todo el mundo para la grabación de señales EEG. La siguiente tabla muestra los más extendidos y disponibles de forma comercial que se hayan usado en al menos cinco artículos.

Tabla 1: Dispositivos de electroencefalografía disponibles en el mercado.

Dispositivo EEG	Número de artículos
Quik-cap, NuAmps (Compumedics NeuroScan Inc., El Paso, TX, USA)	33
Active-electrodes (BioSemi Inc., Amsterdam, Netherlands)	28
EPOC (Emotiv Inc., San Francisco, CA, USA)	24
Geodesic Sensor Net (Electrical Geodesics Inc., Eugene, OR, USA)	22
actiCAP, EASYCAP, BrainCap (Brain Products Inc., Munich, Germany)	22
EasyCap (FMS, Herrsching-Breitbrunn, Germany)	15
Electro-Cap (Electro-Cap International Inc., Eaton, OH, USA)	9
g.MOBILab, g.EEGcap (g.tec Guger Technologies Inc., Graz, Austria)	7

3.3.1.2 Electroodos

En cuanto al número de electrodos, los aspectos a considerar son el tiempo necesario para montar el dispositivo EEG, el grado de confort de los participantes, la usabilidad y el número de características a extraer. Teniendo estos aspectos en cuenta, lo que se recomienda es no usar una gran cantidad de electrodos, aunque la mayoría de dispositivos EEG sí que requieren un número de electrodos elevado [20].

El sistema 10-20 es un método reconocido internacionalmente para describir y aplicar la ubicación de los electrodos en el cuero cabelludo en el contexto de un examen EEG, un estudio de sueño polisomnográfico o una investigación de laboratorio voluntaria. El "10" y el "20" se refieren al hecho de que las distancias reales entre los electrodos adyacentes son el 10% o el 20% de la distancia total entre las partes delantera y trasera del cráneo o entre las partes derecha e izquierda del cráneo [23]. En total, describe la posición de 21 electrodos.

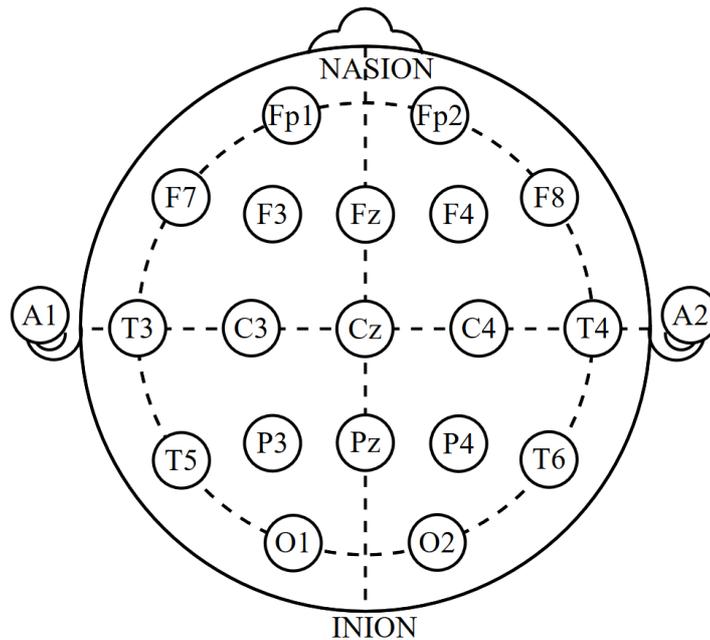


Figura 6: Electrodo utilizados en el sistema 10-20. Fuente: [23].

Un sistema de mayor resolución se consigue agregando electrodos adicionales mediante la división del 10%, que llena los sitios intermedios a la mitad entre el sistema 10-20. En [21], el artículo correspondiente a la base de datos utilizada en este trabajo, se utilizan 32 canales y se obtienen unas tasas de acierto para la clasificación binaria de valencia y excitación de 57.6% y 62% respectivamente.

En [22], se usan 16 canales y se obtiene una tasa de acierto de un 98% para la clasificación de 6 categorías musicales.

3.3.2 Preprocesado de las señales EEG

Como se comentó en el capítulo 1, la actividad cerebral genera impulsos eléctricos que corresponden a diferentes ritmos u ondas cerebrales. La frecuencia de estas ondas varía entre 0.5 y 45Hz, por lo que se antoja necesario filtrar las señales en este rango de frecuencias.

Además, las señales EEG suelen ser ruidosas y contienen artefactos de magnitud grande, siendo los más comunes los ocasionados por movimientos oculares y por el parpadeo. Es por esto que para poder trabajar con señales EEG será necesario detectar y eliminar estos artefactos previamente.

La base de datos utilizada en el presente trabajo pone a nuestra disposición las señales EEG ya pre-procesadas y listas para trabajar con ellas, por lo que en nuestro caso no será necesario realizar esta tarea.

3.3.3 Extracción de características

Se pueden considerar una gran multitud de características diferentes a extraer a partir de señales EEG. Las características más comunes en la literatura son las siguientes:

- 1) Coeficientes de la transformada de Fourier de la señal EEG.
- 2) Coeficientes de la densidad espectral de potencia de la señal EEG.
- 3) Diferencia entre densidades espectrales de potencia de señales EEG en electrodos simétricos.
- 4) Potencia total en cada banda cerebral de la señal EEG.
- 5) Diferente entre potencias totales en bandas cerebrales de la señal EEG en electrodos simétricos.

El procedimiento para la extracción de estas características es bastante intuitivo. Para los 3 primeros grupos bastaría con calcular la FFT de las señales EEG y posteriormente la densidad espectral de potencia a partir de tal transformada. Para los grupos 4 y 5, sería necesario filtrar la señal EEG con filtros paso banda correspondientes a cada banda cerebral y calcular cada una de las potencias.

En los capítulos 4 y 6 se detallarán las técnicas y características utilizadas en el presente trabajo, las cuales entrarían en los grupos 4 y 5 arriba mencionados.

3.3.3.1 Revisión en la literatura de características para la detección de emociones

Uno de los resultados más evidentes es el importante papel de la banda *alpha*. Kostyunina y Kulikov encontraron que diferentes estados emocionales muestran diferentes frecuencias pico en la banda *alpha* [34]. Shemyakina y Danko observaron una sincronización espacial (comportamiento parecido de las potencias de señales EEG en diferentes zonas del cerebro) para diferentes emociones, especialmente en el lóbulo temporal [35].

3.3.3.1.1 Valencia

Una de las cosas más claras en cuanto a las emociones en mediciones de señales EEG es la importancia de la diferencia entre los hemisferios izquierdo y derecho. Se ha sugerido que en esta diferencia se puede encontrar información sobre la dimensión de la valencia, y recibe el nombre de valencia emocional hemisférica (hev) [13]. Hay dos teorías sobre la valencia emocional hemisférica:

- 1) **Hipótesis del hemisferio derecho:** esta hipótesis afirma que el hemisferio derecho está involucrado en el procesamiento de las emociones. Además, esta lateralización es más evidente con emociones negativas.
- 2) **Hipótesis de asimetría de Valencia:** esta hipótesis establece que el grado de participación de cada hemisferio depende de la valencia de la emoción. El hemisferio derecho es el dominante con las emociones negativas, mientras que el izquierdo presenta una mayor activación con las emociones positivas.

Aunque ambas hipótesis han sido apoyadas por multitud de investigaciones, lo cierto es que artículos recientes que han analizado gran parte de estas investigaciones no han encontrado evidencia empírica para ninguna de las teorías. Schiffer et al. sugieren que la dirección de la valencia emocional hemisférica es una cuestión personal y varía en cada individuo, tal y como ocurre con el género o la forma de escribir [36].

En [40], el artículo correspondiente a la base de datos utilizada en este trabajo, se calcularon las correlaciones entre la valencia y las potencias en cada banda medidas en los diferentes electrodos utilizados.

La figura de a continuación muestra la correlación entre las diferentes zonas del cerebro y el valor de la valencia: un color más cálido indica una correlación positiva fuerte y un color frío una correlación negativa grande.

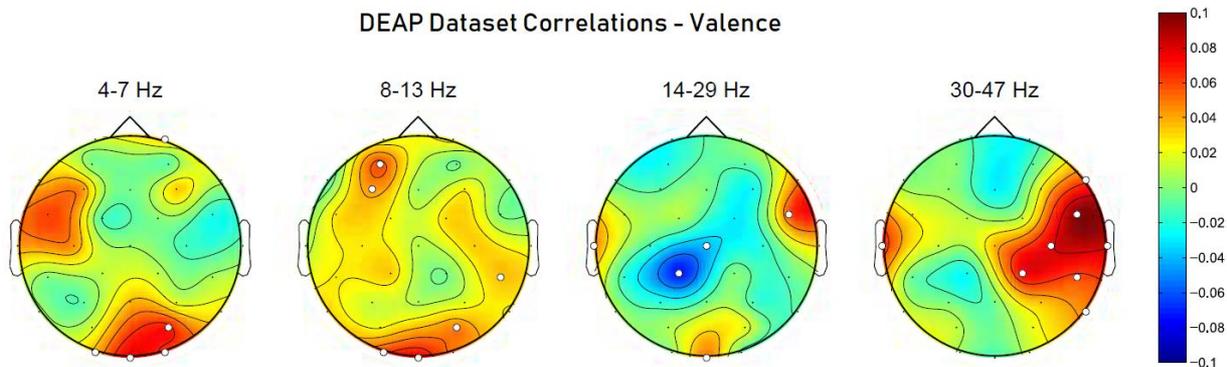


Figura 7: Correlación entre las diferentes bandas y electrodos y el valor de la valencia. Fuente: [40].

En la siguiente tabla se señalan los electrodos que correlan significativamente con los valores de valencia.

Tabla 2: Electrodo en cada banda que muestran una mayor correlación con el valor de la valencia.

Banda	Electrodos
Theta (4 - 7 Hz)	O1, OZ , O2 , PO4 , Fp2
Alpha (8 - 13 Hz)	F3, AF3 , O1 , OZ , PO4, CP6
Beta (14 - 29 Hz)	T7, CP1 , CZ, OZ, FC6
Gamma (30 - 47 Hz)	T7 , CP2 , C4 , CP6, FC6 , P8, F8 , T8

3.3.3.1.2 Excitación

Mientras que para medir la valencia a partir de señales EEG hay mucha información y varias teorías, para el caso de la excitación se conoce bastante menos. Heller, Nitschke, y Lindsay ofrecen en [37] una visión general del resultado de múltiples estudios, y la principal conclusión que obtienen es que el hemisferio posterior derecho algo tiene que ver con la excitación.

Aftanas et al. declaran que no sólo el hemisferio posterior derecho está relacionado con la excitación, sino también el hemisferio anterior izquierdo [38]. Muestran que los estados emocionales con un valor de activación alto o bajo, en comparación con estados neutros, muestran incrementos asimétricos de la actividad en las áreas posteriores del hemisferio y las áreas anteriores del hemisferio izquierdo.

Krause et al. adoptan otro enfoque estudiando las diferencias entre las bandas de las señales EEG en lugar de las zonas del cerebro [39]. Sus investigaciones muestran que la banda 4-6 Hz presenta una mayor sincronización al ver una película agresiva que al ver una película neutral o triste.

En [40], el artículo correspondiente a la base de datos utilizada en este trabajo, se calcularon las correlaciones entre la excitación y las potencias en cada banda medidas en los diferentes electrodos utilizados.

La siguiente figura muestra la correlación entre las diferentes zonas del cerebro y el valor de la excitación: un color más cálido indica una correlación positiva grande y un color frío una correlación negativa fuerte.

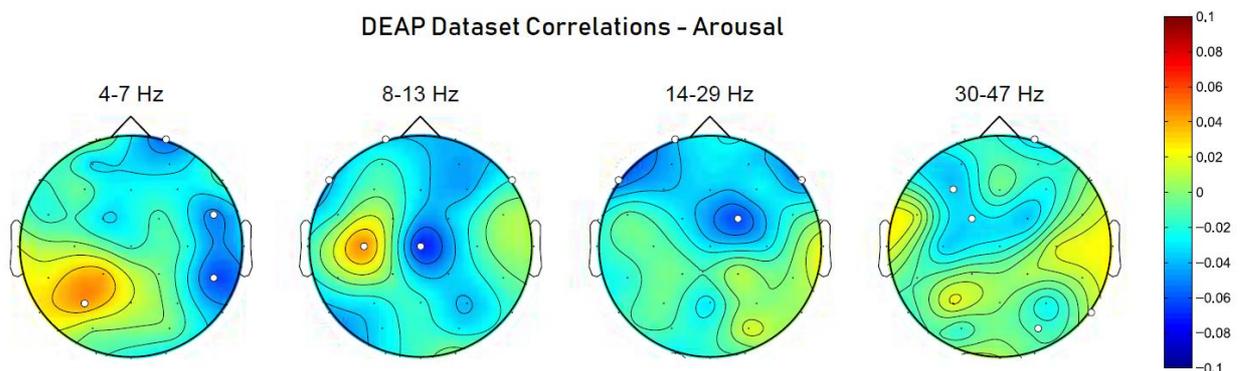


Figura 8: Correlación entre las diferentes bandas y electrodos y el valor del grado de excitación. Fuente: [40].

En la siguiente tabla se señalan los electrodos que correlacionan significativamente con los valores de excitación.

Tabla 3: Electrodo en cada banda que muestran una mayor correlación con el valor del grado de excitación.

Banda	Electrodos
<i>Theta</i> (4 - 7 Hz)	P3, Fp2 , FC6, CP6
<i>Alpha</i> (8 - 13 Hz)	F7, C3, Fp1, Cz
<i>Beta</i> (14 - 29 Hz)	F7 , Fp1, FC2 , F8
<i>Gamma</i> (30 - 47 Hz)	F3, FC1, Fp2, PO4, P8

3.3.4 Entrenamiento y validación de un clasificador

Para entrenar y validar un clasificador es preciso disponer de un conjunto de datos etiquetado. Este conjunto de datos no es más que una serie de muestras, cada una con una etiqueta indicando la clase a la que pertenece dicha muestra.

El objetivo del entrenamiento es que el clasificador sea capaz de predecir la clase de una muestra sin etiquetar que recibe como entrada. Para llevar a cabo el entrenamiento y la posterior validación, es conveniente separar el conjunto de datos en los siguientes conjuntos:

- 1) Conjunto de datos de entrenamiento: Este conjunto de datos es el que se utilizará durante la fase de entrenamiento del clasificador, y las clases de las muestras serán conocidas.
- 2) Conjunto de datos de test: Este conjunto de datos se utilizará como entrada al clasificador para comprobar su desempeño. El clasificador ante cada muestra de entrada predecirá la clase a la que esta pertenece, y se comparará posteriormente las clases predichas con las reales.

Existen multitud de algoritmos y clasificadores. En el capítulo 4 se hablará sobre aquellos utilizados en este trabajo.

3.4 Resultados previos en la literatura para la clasificación de emociones

En el año 2000, Choppin propuso el uso de señales EEG para clasificar seis emociones en base a los niveles de valencia y excitación [7]. Asoció las emociones positivas con una coherencia alta (potencias similares en ambos hemisferios) en la banda alpha en el lóbulo frontal y con una potencia alta en la banda beta en el lóbulo parietal del hemisferio derecho. Niveles mayores de excitación fueron asociados con una coherencia y potencia alta en la banda beta en el lóbulo parietal y con una actividad menor en la banda alpha.

En 2003, Ishino y Hagiwara propusieron un sistema que estimaba el estado afectivo a través del uso de redes neuronales para la clasificación de estados emocionales basados en características extraídas de señales EEG [27]. Reportaron unos resultados de entre 54.5% y 67.7% tasa de acierto para cada uno de los cuatro estados emocionales.

En 2004, Takahashi usó una cinta para la cabeza con tres electrodos para la clasificación de cinco emociones (alegría, ira, tristeza, miedo y relajación) en base a múltiples señales (EEG, pulso y conductancia de la piel) [28]. Entrenó clasificadores usando máquinas de soporte vectorial (SVM) y obtuvo las tasas de acierto usando por un lado el conjunto de todas las señales y por otro únicamente las señales EEG. Obtuvo una tasa de acierto de 41.7% para el reconocimiento de cinco emociones usando máquinas de soporte vectorial.

En 2006, Oude describió un método para el reconocimiento de emociones a partir de señales EEG grabadas con el dispositivo BraInquiry PET [29]. Utiliza los electrodos Fpz, F3 y F4 y un clasificador lineal basado en el análisis discriminante lineal de Fisher, obteniendo tasas de acierto para la clasificación binaria de valencia y excitación superiores a 90%.

En 2007, Heraz, Razaki y Frasson utilizaron el dispositivo Pendant EEG para la grabación de señales EEG y

entrenaron un clasificador basado en el método de los k vecinos más cercanos para el reconocimiento de ocho emociones (ira, aburrimiento, confusión, odio, curiosidad, disgusto, euforia y frustración) [10]. Obtuvieron una tasa de acierto del 82.27%.

En 2009, Chanel, Kierkels, Soleymani y Pun entrenaron un clasificador basado en SVM para la detección de tres estados emocionales (negativo, positivo y neutral) a partir de información en tiempo y frecuencia de señales EEG, obteniendo una tasa de acierto de 63% [30].

También en 2009, Zhand y Lee propusieron un sistema para la clasificación del estado emocional en dos clases (positivo y negativo) utilizando imágenes como estímulo y un clasificador basado en SVM entrenado con características relativas a la asimetría en el lóbulo frontal, obteniendo una tasa de acierto de 73% [31].

En 2010, Lin, Wang, Wu, Jeng y Chen utilizaron diferentes modelos de clasificación basados en SVM para el reconocimiento de cuatro emociones (alegría, placer, ira y tristeza) y la clasificación binaria de valencia y excitación, obteniendo tasas de acierto de 92.57%, 94.86% y 94.43% respectivamente [32].

En 2012, Ramirez y Zacharias propusieron un método para detectar emociones a partir de señales EEG obtenidas mediante el dispositivo Emotiv EPOC [16]. En sus experimentos, los participantes escucharon sonidos extraídos de la base de datos IADS (International Affective Digitized Sounds). Utilizaron el modelo de Russel basado en las dos dimensiones valencia y excitación, y seleccionaron 12 estímulos situados en los extremos del plano (3 en cada cuadrante). Posteriormente midieron las señales EEG en los electrodos AF3, AF4 y F3. Para determinar la excitación utilizaron el ratio entre las potencias en las bandas beta y alpha, y para determinar la valencia compararon el nivel de activación de ambos hemisferios. Utilizando clasificadores basados en SVM para la clasificación binaria de los niveles de excitación y valencia, obtuvieron tasas de acierto de 77.82% y 80.11% respectivamente.

3.5 Conclusiones

En este capítulo se ha hecho una revisión del estado del arte en el reconocimiento de emociones:

- Se ha explicado las diferentes tareas en las que se divide el proceso de clasificación de emociones mediante el electroencefalograma: obtención de señales EEG, preprocesado de las señales EEG, extracción de características y entrenamiento y validación de un clasificador.
- Se han comentado los resultados obtenidos en investigaciones y experimentos previos para la clasificación de emociones mediante el electroencefalograma, en las que se observa que el método más utilizado para clasificar es SVM. Las características utilizadas suelen ser los coeficientes de la transformada de Fourier de las señales EEG o potencias en diferentes zonas del cerebro y bandas cerebrales.

4 BASE DE DATOS UTILIZADA: DEAP DATASET

En este capítulo se va a hablar sobre la base de datos utilizada, la cual recibe el nombre de DEAP dataset.

DEAP dataset es una base de datos para el análisis de emociones en personas. Consiste en una serie de señales de encefalograma recogidas para 32 participantes: para estos, los electroencefalogramas en diferentes electrodos y otras señales fisiológicas fueron registrados mientras veían fragmentos de vídeos musicales. Los participantes puntuaron cada vídeo en términos de:

- Nivel de excitación
- Nivel de valencia.
- Nivel de agrado (cuánto les gustó).
- Dominancia.
- Familiaridad.

Todas las puntuaciones se dieron en una escala continua entre 1 y 9 excepto la familiaridad, que se dio en una escala discreta de 1 a 5.

De los 32 participantes, la mitad fueron mujeres y la otra mitad hombres. Sus edades estaban comprendidas entre 19 y 37 años (26.9 años de media) [40].

4.1 Elección de los estímulos

Para la elección de los vídeos que se usarían en el experimento, se realizó previamente una autoevaluación online en la que los participantes puntuaban vídeos de un conjunto de 120 vídeos musicales en términos de valencia, activación y dominancia.

A partir de esta autoevaluación online, se eligieron finalmente los 40 vídeos que se usarían en los experimentos. La siguiente figura ilustra cómo era la interfaz web utilizada para realizar la autoevaluación online [40].

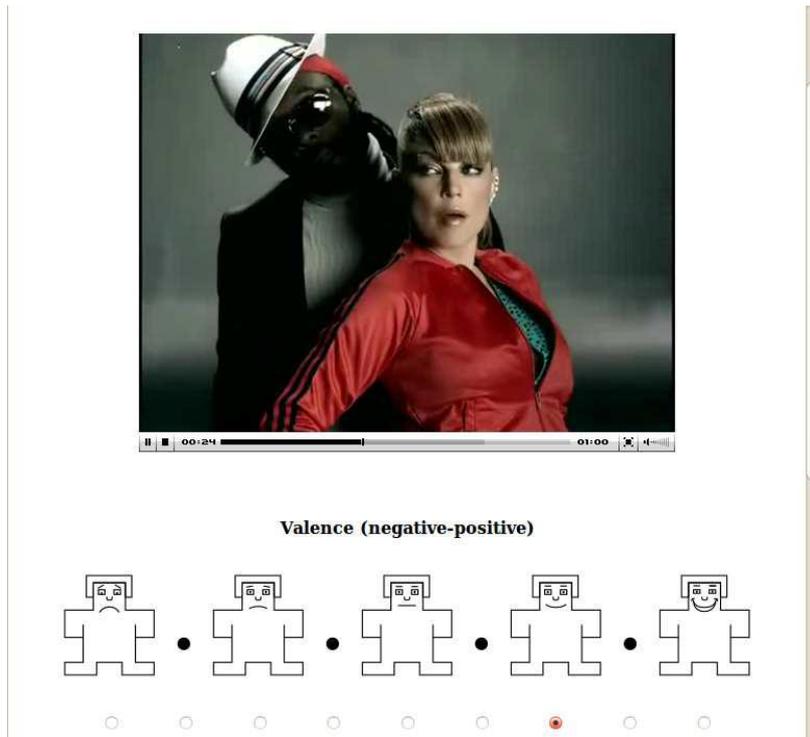


Figura 9: Captura de pantalla de la interfaz web usada para la autoevaluación online. Fuente: [40].

4.2 Experimentos

En cada experimento, un participante mira un vídeo de música durante un minuto (se dan 3 segundos de reposo al principio de cada experimento), y durante él se registra el encefalograma en cada uno de los 32 electrodos. Por lo que por experimento por paciente, se tienen 32 señales de encefalograma de 63 segundos de duración cada una.

En total, se dispone de 1280 muestras (señales y puntuaciones de 32 participantes y 40 vídeos por participante) para la clasificación de los niveles de activación y valencia a partir del análisis del encefalograma [40].

Los 32 electrodos usados en el experimento, y para cada cual se registra un encefalograma por muestra, son los que se marcan en amarillo en la siguiente figura.

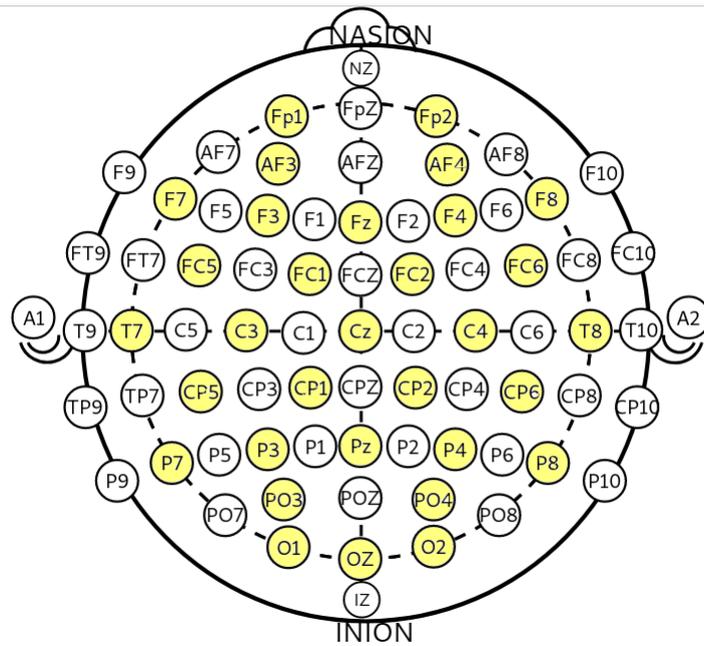


Figura 10: Electrodo (amarillo) utilizados en los experimentos del DEAP dataset.

4.3 Uso de la base de datos

La base de datos es pública, aunque para poder tener acceso a ella, es necesario imprimir, firmar y escanear un EULA (End User License Agreement) y enviarlo por email.

La siguiente tabla muestra los ficheros disponibles una vez se tiene acceso a la base de datos.

Tabla 4: Ficheros disponibles en la base de datos DEAP dataset.

Fichero	Formato	Contenido
online_ratings	xls, csv, ods	Puntuaciones individuales recogidas durante la autoevaluación online.
video_list	xls, csv, ods	Lista con todos los vídeos utilizados en la autoevaluación online y los experimentos.
participant_ratings	xls, csv, ods	Puntuaciones de los participantes para cada vídeo recogidas durante los experimentos.
participant_questionnaire	xls, csv, ods	Respuestas de los participantes al cuestionario rellenado antes de los experimentos.
face_video.zip	Archivo .zip	Videos de las caras de los 22 primeros participantes grabados durante los experimentos.
data_original.zip	Archivo .zip	Datos recogidos originales. Hay 32 ficheros .bdf, cada uno con 48 canales a 512 Hz (32 canales EEG, 12 canales periféricos, 3 canales sin usar y 1 canal de estado).

data_preprocessed_matlab.zip	Archivo .zip	Datos recogidos preprocesados (Señales EEG filtradas entre 3 y 47 Hz y submuestreadas a 128 Hz). Contiene 32 ficheros .mat (matlab).
data_preprocessed_python.zip	Archivo .zip	Datos recogidos preprocesados (Señales EEG filtradas entre 3 y 47 Hz y submuestreadas a 128 Hz). Contiene 32 ficheros .dat.

Archivo data_preprocessed_matlab.zip

Para llevar a cabo este trabajo, se ha hecho uso únicamente del archivo comprimido data_preprocessed_matlab.zip. Este archivo contiene 32 ficheros .mat para cargar con matlab, y cada uno de ellos recoge los datos relativos a uno de los participantes.

Al cargar uno de estos ficheros .mat con matlab, dispondremos de las siguientes matrices:

- **data**

Matriz tridimensional de 40x40x8064. La primera dimensión se corresponde con el nº de muestra o experimento (cada vídeo), la segunda dimensión con uno de los electrodos (1-32) o una señal fisiológica (33-40) y la tercera con un instante de tiempo de la señal (63 segundos a 128 Hz).

- **labels**

Matriz de 40x4 con las etiquetas (valencia, activación, dominancia y agrado) correspondientes a cada muestra o experimento. Las etiquetas son las puntuaciones que el participante ha dado a cada vídeo en términos de valencia, activación, dominancia y agrado en una escala continua de 1 a 9.

4.4 Conclusiones

En este capítulo se ha hablado acerca de la base de datos utilizada en el presente trabajo, que se llama DEAP dataset. Se ha explicado sin entrar en gran detalle cómo se elaboró (los experimentos que se llevaron a cabo) y los datos que contiene. Se ha visto que de todos los ficheros disponibles en la base de datos, se ha utilizado únicamente el archivo **data_preprocessed_matlab.zip**, que contiene una serie de señales EEG y etiquetas para cada una de ellas en términos de valencia, activación, dominancia y agrado.

5 TÉCNICAS UTILIZADAS

Como se comentó en la más atrás del capítulo 3 correspondiente al estado del arte en el reconocimiento de emociones, esta labor conlleva una serie de tareas: pre-procesado de las señales EEG, elección y extracción de características de las señales EEG e implementación y entrenamiento de un clasificador.

La base de datos utilizada en el presente trabajo pone a nuestra disposición las señales EEG ya pre-procesadas y listas para trabajar con ellas, por lo que en nuestro caso no será necesario realizar esta tarea.

En este capítulo se detallarán las características elegidas a extraer de las señales EEG, las técnicas utilizadas para la extracción de dichas características y se explicarán los diferentes clasificadores que se han implementado y probado en este trabajo para clasificar emociones a partir del encefalograma.

5.1 Elección de características

Se ha decidido extraer 3 conjuntos de características diferentes para la clasificación de valencia y otros 3 conjuntos de características para la clasificación del grado de excitación. A lo largo de la memoria se hará referencia a estos conjuntos de características indicando simplemente el número del conjunto en cuestión.

La siguiente tabla describe dichos conjuntos de características.

Tabla 5: Conjuntos de características extraídos para la clasificación de la valencia y grado de excitación.

Nº conjunto	Tipo	Nº características	Descripción
1	Valencia	230	Características propuestas en [40]: potencias en cada banda para todos los electrodos y asimetría de potencia en cada banda entre pares de electrodos simétricos.
2	Valencia	10	Características propuestas en [22] para clasificación de valencia.
3	Valencia	24	Potencias en las bandas y electrodos que muestran una mayor correlación con la valencia [40]. Ver Tabla 2.
4	Excitación	230	Características propuestas en [40]: potencias en cada banda para todos los electrodos y asimetría de potencia entre pares de electrodos simétricos.
5	Excitación	17	Características propuestas en [22] para clasificación del grado de excitación.
6	Excitación	17	Potencias en las bandas y electrodos que muestran una mayor correlación con el grado de excitación [40]. Ver Tabla 3.

5.1.1 Conjuntos de características 1 y 4

En [40], el artículo relativo a la base de datos utilizada, se proponen como características las potencias en cada banda (considerando las bandas theta, slow alpha, alpha, beta y gamma) y en cada electrodo (32 electrodos * 5 bandas = 160 características) y la asimetría de potencia entre pares de electrodos simétricos en cada banda (14 pares * 5 bandas = 70 características), haciendo un total de 230 características.

Los conjuntos 1 y 4 contienen ambos las mismas características.

Para el cálculo de las potencias en cada banda y electrodo, por cada electrodo se filtra la señal EEG correspondiente y posteriormente se calculan las potencias en cada banda.

Para el cálculo de la asimetría de potencia entre pares de electrodos simétricos, se usa la siguiente fórmula propuesta en [41].

$$\frac{R - L}{R + L} \quad 5.1$$

Donde R y L representan un par de electrodos simétricos: R representa la potencia del electrodo situado en el hemisferio derecho y L la potencia del situado en el hemisferio izquierdo. Se aplicaría dicha fórmula para las potencias en cada banda de cada par de electrodos simétricos.

Los pares de electrodos simétricos presentes en la base de datos utilizada se muestran en la siguiente tabla.

Tabla 6: Pares de electrodos simétricos en las muestras del DEAP dataset.

Par	Hemisferio derecho	Hemisferio Izquierdo
1	Fp2	Fp1
2	AF4	AF3
3	F4	F3
4	F8	F7
5	FC2	FC1
6	FC6	FC5
7	C4	C3
8	T8	T7
9	CP2	CP5
10	CP6	CP5
11	P4	P3
12	P8	P7

13	PO4	PO3
14	O2	O1

5.1.2 Conjuntos de características 3 y 6

Los conjuntos 3 y 6 son subconjuntos de los conjuntos 1 y 4, y cada uno de ellos contiene las potencias de las parejas banda-electrodo que presentan una mayor correlación con la valencia y el grado de excitación, respectivamente, tal y como se describe en la sección 3.2.2.1 del capítulo 3.

La siguiente tabla ilustra las características del conjunto 3.

Tabla 7: Características del conjunto 3.

$\theta O1$
θOZ
$\theta O2$
$\theta P04$
$\theta Fp2$
$\alpha F3$
$\alpha AF3$
$\alpha O1$
αOZ
$\alpha P04$
$\alpha CP6$
$\beta T7$
$\beta CP1$
βCZ
βOZ
$\beta FC6$
$\gamma T7$
$\gamma CP2$
$\gamma C4$

γ_{CP6}
γ_{FC6}
γ_{P8}
γ_{F8}
γ_{T8}

La tabla de continuación muestra las características del conjunto 6.

Tabla 8: Características del conjunto 6.

θ_{P3}
θ_{Fp2}
θ_{FC6}
θ_{CP6}
α_{F7}
α_{C3}
α_{Fp1}
α_{FC2}
α_{F8}
β_{T7}
β_{Fp1}
β_{FC2}
β_{F8}
γ_{T7}
γ_{F3}
γ_{FC1}
γ_{Fp2}
γ_{P04}
γ_{P8}

Cada característica representa la potencia en una banda y electrodo concretos, por ejemplo, $\beta Fp1$ correspondería a la potencia en la banda *beta* y electrodo Fp1.

5.1.3 Conjuntos de características 2 y 5

Los conjuntos de características 2 y 5 contienen aquellas características propuestas en [22] para la clasificación de la valencia y el grado de excitación, respectivamente. Cada característica es una combinación de potencias de diferentes parejas banda-electrodo, y el cálculo de cada una de ellas es trivial una vez se tiene la potencia de cada pareja banda-electrodo.

En estos conjuntos de características se usan también potencias totales en electrodos, es decir, la potencia de la señal EEG de un electrodo completa y no en una banda determinada.

La siguiente tabla ilustra las características del conjunto 2.

Tabla 9: Características del conjunto 2.

$(\beta AF4/\alpha AF4) - (\beta AF3/\alpha AF3)$
$[(\beta AF4 + \beta F4)/(\alpha AF4 + \alpha F4)] -$ $[(\beta AF3 + \beta F3)/(\alpha AF3 + \alpha F3)]$
$[(\beta AF4 + \beta F4 + \beta F8 + \beta FC6 + \beta T8 + \beta P4 + \beta P8 + \beta O2) /$ $(\alpha AF4 + \alpha F4 + \alpha F8 + \alpha FC6 + \alpha T8 + \alpha P4 + \alpha P8 + \alpha O2)] -$ $[(\beta AF3 + \beta F3 + \beta F7 + \beta FC5 + \beta T7 + \beta P3 + \beta P7 + \beta O1) /$ $(\alpha AF3 + \alpha F3 + \alpha F7 + \alpha FC5 + \alpha T7 + \alpha P3 + \alpha P7 + \alpha O1)]$
AF4 – AF3
F4 – F3
F8 – F7
FC6 – FC5
T8 – T7
P8 – P7
O2 – O1
$\beta FC2$
$\beta F8$
$\gamma T7$
$\gamma F3$
$\gamma FC1$
$\gamma Fp2$

$\gamma P04$
$\gamma P8$

La tabla de acontinuación muestra las características del conjunto 5.

Tabla 10: Características del conjunto 5.

$(\beta AF4 + \beta AF3)/(\alpha AF4 + \alpha AF3)$
$(\beta AF4 + \beta AF3 + \beta F4 + \beta F3) /$ $(\alpha AF4 + \alpha AF3 + \alpha F4 + \alpha F3)$
$(\beta AF4 + \beta F4 + \beta F8 + \beta FC6 + \beta T8 + \beta P4 + \beta P8 + \beta O2 +$ $\beta AF3 + \beta F3 + \beta F7 + \beta FC5 + \beta T7 + \beta P3 + \beta P7 + \beta O1) /$ $(\alpha AF4 + \alpha F4 + \alpha F8 + \alpha FC6 + \alpha T8 + \alpha P4 + \alpha P8 + \alpha O2 +$ $\alpha AF3 + \alpha F3 + \alpha F7 + \alpha FC5 + \alpha T7 + \alpha P3 + \alpha P7 + \alpha O1)$
$\beta AF3/\alpha AF3$
$\beta AF4/\alpha AF4$
$\beta F3/\alpha F3$
$\beta F4/\alpha F4$
$\beta F7/\alpha F7$
$\beta F8/\alpha F8$
$\beta FC5/\alpha FC5$
$\beta FC6/\alpha FC6$
$\beta T7/\alpha T7$
$\beta T8/\alpha T8$
$\beta P7/\alpha P7$
$\beta P8/\alpha P8$
$\beta O1/\alpha O1$
$\beta O2/\alpha O2$

5.2 Técnicas para la extracción de características

Atendiendo a la sección anterior, observamos que las características que nos van a ser útiles están todas relacionadas con la potencia en alguna determinada banda y zona del cerebro o con la relación entre varias de estas potencias. Ejemplos de características serían las siguientes:

- Potencia en la banda *theta* en el electrodo P8: $\theta P8$.
- Ratio de potencia entre las banda *beta* y *alpha* en el electrodo AF3: $\beta AF3 / \alpha AF3$.
- Diferencia entre la potencia total en el electrodo FC6 y la potencia total en el electrodo FC5: $FC6 - FC5$.

Como se ha comentado en el capítulo 4, los datos con los que se han trabajado son señales EEG filtradas entre 3 y 47 Hz y muestreadas a 128 Hz. Todas las características extraídas y utilizadas están relacionadas con potencias en diferentes bandas y en diferentes electrodos, con lo que podemos dividir el proceso de extracción de características en los siguientes pasos:

- Filtrado de las señales EEG en cada banda.
- Cálculo de las potencias por banda y potencia total en cada electrodo.
- Cálculo de las características a partir de las potencias.
- Exportación de las características.

Todo el proceso de extracción de características se ha hecho utilizando Matlab.

5.2.1 Filtrado de las señales EEG en cada banda

Para filtrar las señales EEG en cada banda, se han diseñado en Matlab los siguientes filtros paso banda IIR:

- Filtro para la banda *theta* (4 – 8 Hz).

```
designfilt('bandpassiir', 'FilterOrder', 20, 'PassbandFrequency1', 4, 'PassbandFrequency2', 8, 'PassbandRipple', 1, 'SampleRate', 128)
```

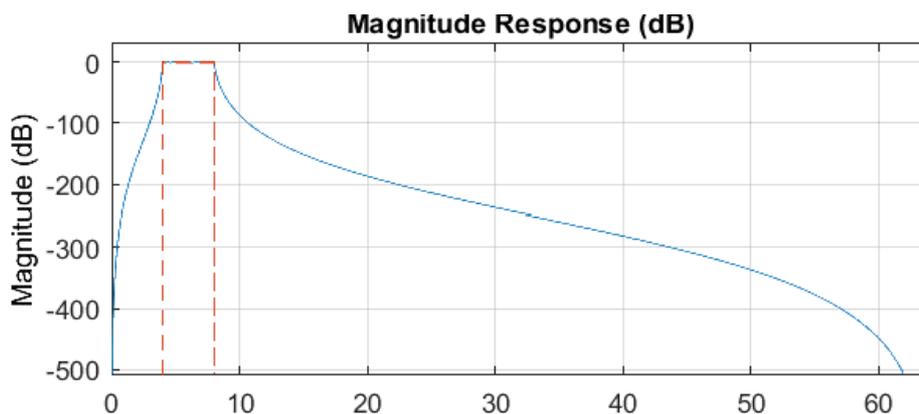


Figura 11: Respuesta en frecuencia del filtro para la banda *theta*.

- Filtro para la banda *slow alpha* (8 – 10 Hz).

```
designfilt('bandpassiir', 'FilterOrder', 20, 'PassbandFrequency1', 8, 'PassbandFrequency2', 10, 'PassbandRipple', 1, 'SampleRate', 128)
```

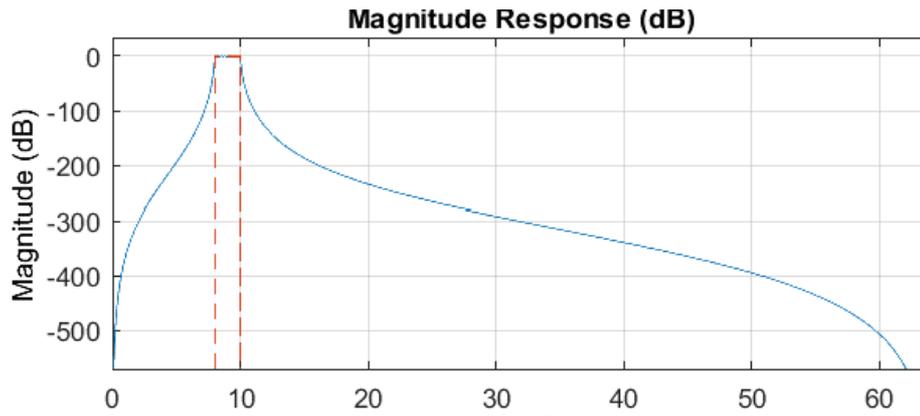


Figura 12: Respuesta en frecuencia del filtro para la banda *slow alpha*.

- Filtro para la banda *alpha* (8 – 12 Hz).

```
designfilt('bandpassiir', 'FilterOrder', 20, 'PassbandFrequency1', 8,
'PassbandFrequency2', 12, 'PassbandRipple', 1, 'SampleRate', 128)
```

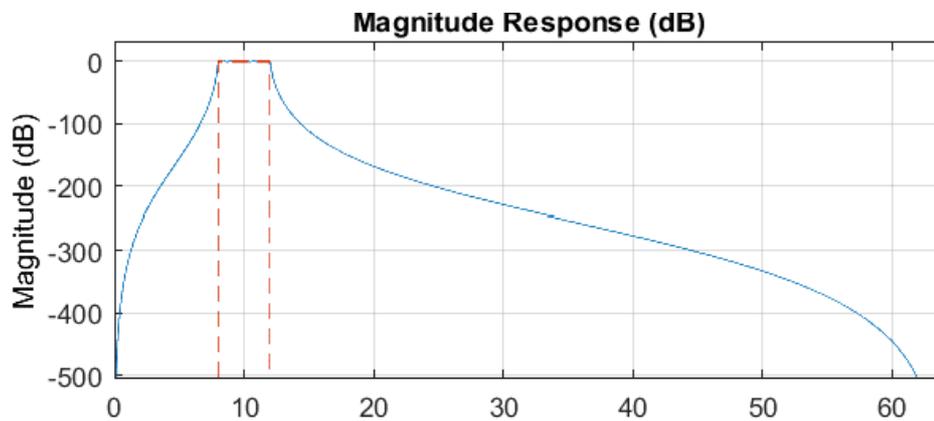


Figura 13: Respuesta en frecuencia del filtro para la banda *alpha*.

- Filtro para la banda *beta* (12 – 30 Hz).

```
designfilt('bandpassiir', 'FilterOrder', 20, 'PassbandFrequency1', 12,
'PassbandFrequency2', 30, 'PassbandRipple', 1, 'SampleRate', 128)
```

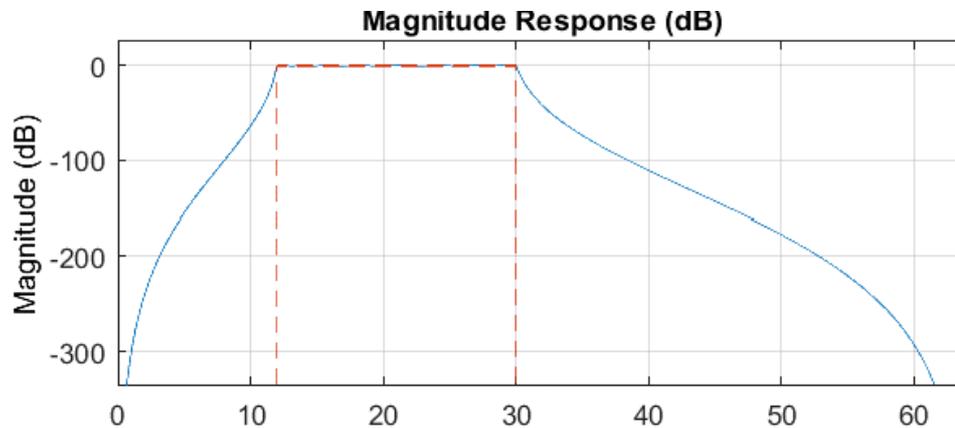


Figura 14: Respuesta en frecuencia del filtro para la banda *beta*.

- Filtro para la banda *gamma* (30 – 45 Hz).

```
designfilt('bandpassiir', 'FilterOrder', 20, 'PassbandFrequency1', 30,  
'PassbandFrequency2', 45, 'PassbandRipple', 1, 'SampleRate', 128)
```

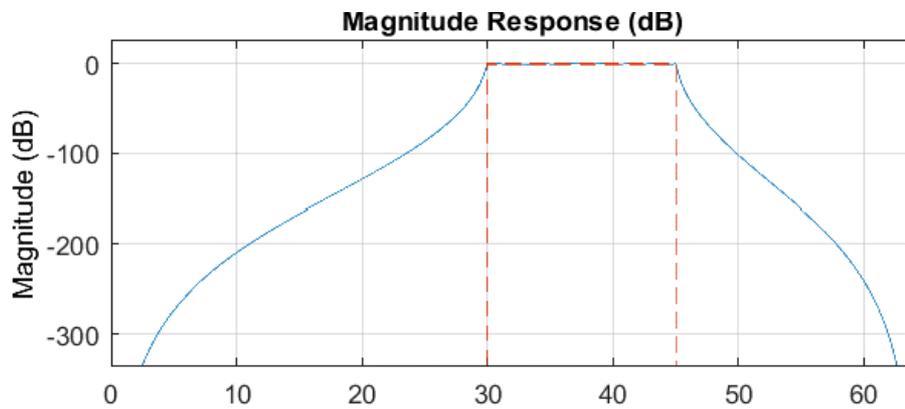


Figura 15: Respuesta en frecuencia del filtro para la banda *gamma*.

Una vez se tienen los filtros de cada banda, obtener la señal filtrada es inmediato aplicando la función `filter` de matlab:

```
s_eeg_beta = filter(filtro_beta, s_eeg)
```

Donde `s_eeg` es un conjunto de señales (matriz en la que cada columna representa una señal diferente y cada fila un instante de tiempo), `filtro_beta` el filtro devuelto por la función `designfilt` y `s_eeg_beta` las señales EEG filtradas en la banda *beta*.

Las figuras siguientes muestran el resultado de filtrar una señal EEG en la banda *beta*.

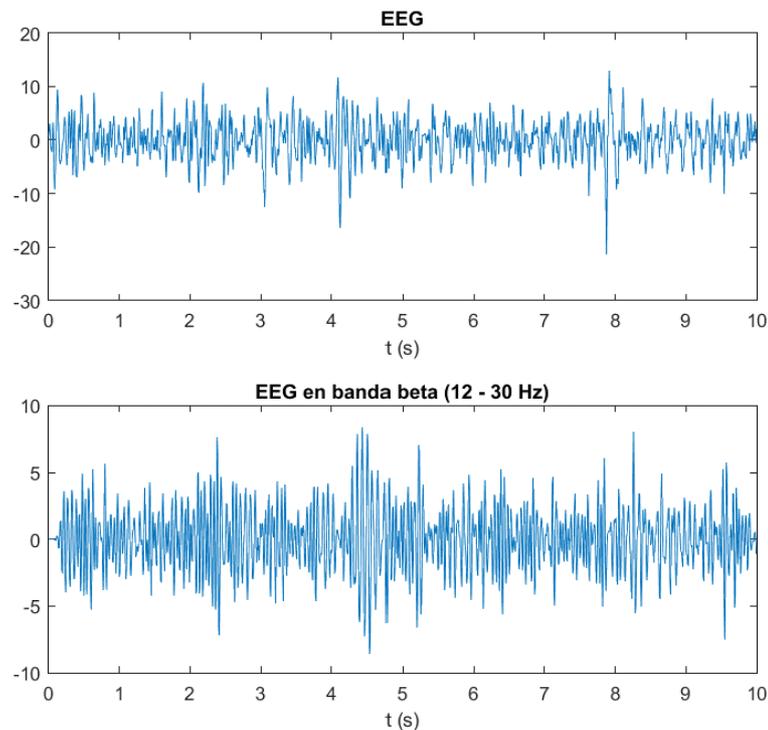


Figura 16: Señal EEG y la misma señal EEG filtrada en la beta, ambas en el tiempo.

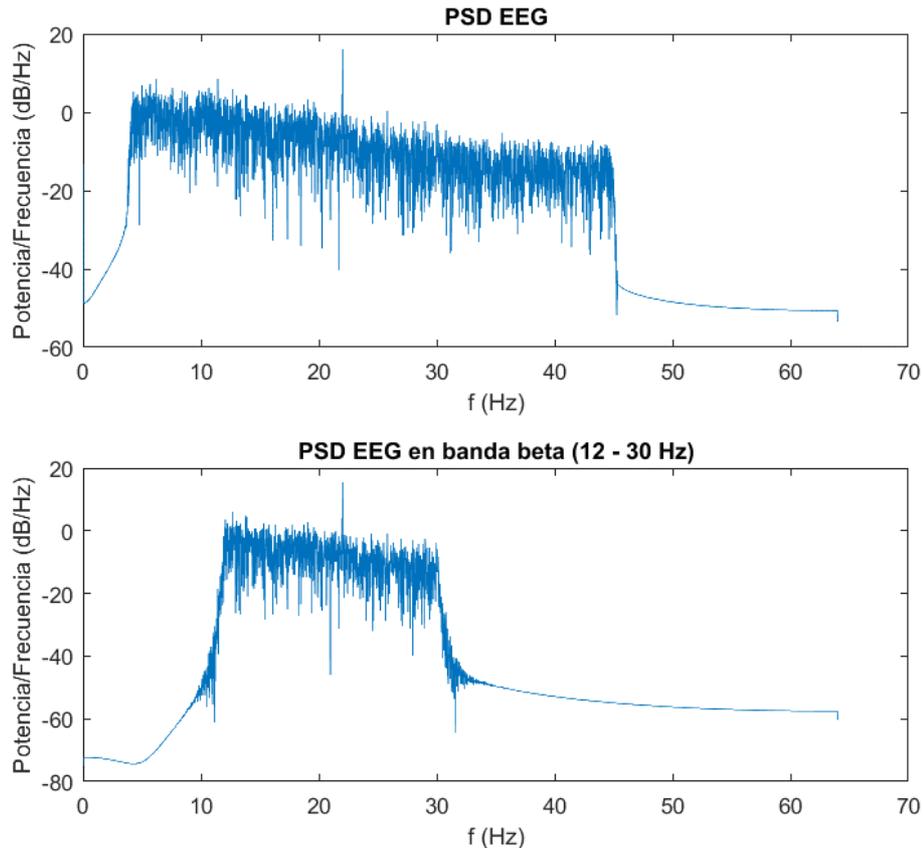


Figura 17: Estima de las densidades espectrales de potencia de la señal EEG y de la misma señal EEG filtrada en la banda beta.

5.2.2 Cálculo de las potencias por banda y potencia total en cada electrodo

El cálculo de las potencias por banda y la potencia total es trivial una vez se tienen las señales correspondientes a cada banda de la señal EEG. Si estamos trabajando con las bandas *theta*, *slow alpha*, *alpha* y *beta*, el cálculo de tales potencias se haría de la siguiente forma para un conjunto de señales s_eeg , donde cada columna representa una señal diferente y cada fila un instante de tiempo:

```
s_eeg_beta = filter(filtro_beta, s_eeg)

% Densidades de potencia
pd_theta    = s_eeg_theta.^2;
pd_salpha  = s_eeg_salpha.^2;
pd_alpha    = s_eeg_alpha.^2;
pd_beta    = s_eeg_beta.^2;

% 1) Potencias absolutas en las diferentes bandas
p_theta     = mean(pd_theta);
p_salpha    = mean(pd_salpha);
p_alpha     = mean(pd_alpha);
p_beta     = mean(pd_beta);

% 2) Potencia total
p_total    = p_theta + p_alpha + p_beta;
```

5.2.3 Cálculo de las características a partir de las potencias

Una vez se tienen las potencias en cada banda y electrodo, se procede a calcular las características que se

necesiten aplicando las fórmulas de cada una de ellas.

Por ejemplo, para calcular las características a, b y c mencionadas al principio del punto 5.2, hacemos lo siguiente:

```
a = p_beta(30);           % 30 es el índice del electrodo P8.
b = p_beta(2)/p_alpha(2); % 2 es el índice del electrodo AF3.
c = p_total(22)-p_total(5);
caracteristicas = [a, b, c];
```

5.2.4 Exportación de las características

Para guardar las características en un fichero CSV y evitar así tener que calcularlas continuamente, utilizamos la función `csvwrite` de Matlab. Antes de exportarlas, añadimos una última columna con las etiquetas.

Las etiquetas corresponden a las puntuaciones de los usuarios en términos de nivel de activación y valencia, tomando valores entre 1 y 9. Con el objetivo de obtener mejores resultados en clasificación binaria, separamos en dos clases aquellas muestras cuyas puntuaciones estén más a los extremos (entre 1 y 3, y entre 7 y 9 por ejemplo), obviando el resto de muestras.

Una vez seleccionadas las muestras, añadimos las etiquetas y exportamos los datos en un fichero CSV que posteriormente se podrá cargar con Matlab o Python.

```
dataset = [caracteristicas etiquetas];
csvwrite('fichero.csv', dataset);
```

5.3 Implementación de la extracción de las características elegidas

La base de datos utilizada consiste en una serie de ficheros `.mat` que pueden cargarse con matlab. Cada fichero `.mat` corresponde a un participante diferente –32 en total– e incluye 32 señales EEG, cada una correspondiente a un electrodo diferente, para cada uno de los 40 trials. Es decir, cada fichero `.mat` contiene $32 * 40 = 1280$ señales EEG.

La duración de estas señales EEG es de 63 segundos y están muestreadas a 128 Hz, por lo que cada señal se compondrá de $63 * 128 = 8064$ muestras. Al cargar el fichero `.mat` en matlab, dispondremos de una matriz tridimensional de $40 \times 40 \times 8064$ en la variable `data`:

- La primera dimensión se corresponde con el n° de muestra o experimento (cada vídeo).
- La segunda dimensión corresponde a uno de los electrodos (1-32) o a una señal fisiológica (33-40).
- La tercera dimensión corresponde a un instante de tiempo de la señal (8064 muestras).

5.3.1 Script `caracteristicas.m`

Para la extracción de los conjuntos de características mencionados en el punto anterior, se ha hecho el script en matlab `caracteristicas.m`. Este script a su vez utiliza otras dos funciones, contenidas en los ficheros `filtrarEEG.m` y `calcularPotencias.m`

La siguiente tabla muestra los diferentes parámetros configurables en este script.

Tabla 11: Parámetros configurables del script características.m.

Parámetro	Valor	Descripción
CONJUNTO_CARACTERISTICAS	Entero [1, 6].	Indica el conjunto de características a extraer de los mencionados en el punto 6.1.
DIR_GUARDAR	Cadena de caracteres.	Directorio donde se va a guardar el fichero .csv con las características y etiquetas.
tipo	“valence” o “arousal”	Indica la etiqueta a utilizar.
TSTART	Vector de enteros entre 0 y 63.	Estas variables indican las ventanas –y por lo tanto, el nº de muestras a extraer– que se van a utilizar en cada señal EEG. Por defecto, se toma una única ventana correspondiente a los últimos 30 segundos de cada EEG.
TSTOP	Vector de enteros entre 0 y 63. Se debe cumplir $TSTOP[i] > TSTART[i] \forall i$	
PACIENTES	Vector de enteros entre 1 y 32.	Indica los participantes a utilizar para la extracción de características. Por defecto, todos.
ELECTRODOS	Vector de enteros entre 1 y 32.	Indica los electrodos a utilizar para la extracción de características. Su valor será diferente en función del conjunto de características elegido.
PARES	Matriz 2xN de enteros entre 1 y 32	Son dos vectores que indican las parejas de electrodos simétricos a utilizar. Su valor será diferente en función del conjunto de características elegido.
TRIALS	Vector de enteros entre 1 y 40.	Indica los trials a utilizar para la extracción de características. Por defecto, todos.
PUNTUACION_CLASE1	Decimal [1, 9).	Estas variables indican las puntuaciones máximas y mínimas para que una muestra se considere de la clase 1 (etiqueta en [0, PUNTUACION_CLASE1] o clase 2 (etiqueta en [PUNTUACION_CLASE1, 9], respectivamente. Los trials que no pertenezcan a ninguna de estas clases, se desecharán. Estas variables nos sirven para quedarnos con los trials más significativos.
PUNTUACION_CLASE2	Decimal (1, 9]. Debe ser mayor que PUNTUACION_CLASE1	

A continuación se va a comentar un poco el código del script sin entrar en gran detalle, con el objetivo de que el lector obtenga una mayor comprensión sobre cómo se lleva a cabo la extracción de características.

- 1) En función del valor de CONJUNTO_CARACTERISTICAS, se asignan los valores correspondientes a las variables tipo, PARES y ELECTRODOS.

```

switch CONJUNTO_CARACTERISTICAS
case 1
    tipo          = "valence";
    % Hemisferio derecho e izquierdo, respectivamente.
    PARES(1,:) = [17 18 20 21 23 22 25 26 28 27 29 30 31 32];
    PARES(2,:) = [1 2 3 4 6 5 7 8 10 9 11 12 13 14];
    ELECTRODOS   = 1:N_ELECTRODOS;
case 2
    tipo          = "valence";
    PARES         = [];
    ELECTRODOS    = [2 3 4 5 8 12 14 18 20 21 22 26 30 32];
    % 2      AF3      1
    % 3      F3       2
    % 4      F7       3
    % 5      FC5      4
    % 8      T7       5
    % 12     P7       6
    % 14     O1       7
    % 18     AF4      8
    % 20     F4       9
    % 21     F8       10
    % 22     FC6      11
    % 26     T8       12
    % 30     P8       13
    % 32     O2       14
case 3
    tipo          = "valence";
    pares         = [];
    ELECTRODOS    = [2 3 8 10 14 15 17 21 22 24 25 26 27 28 30 31 32];
case 4
    tipo          = "arousal";
    PARES(1,:) = [17 18 20 21 23 22 25 26 28 27 29 30 31 32];
    PARES(2,:) = [1 2 3 4 6 5 7 8 10 9 11 12 13 14];
    ELECTRODOS    = 1:N_ELECTRODOS;
case 5
    tipo          = "arousal";
    pares         = [];
    ELECTRODOS    = [2 3 4 5 8 12 14 18 20 21 22 26 30 32];
case 6
    tipo          = "arousal";
    pares         = [];
    ELECTRODOS    = [1 3 4 6 7 11 17 18 21 22 23 24 27 30 31];
otherwise
    % Valores que se hayan configurado a mano.
    ;;
end

```

- 2) Se itera por cada paciente, calculando las potencias en las bandas y electrodos para todos sus trials y para cada trozo de señal EEG definido por las variables TSTART y TSTOP.

```

% Por cada paciente.
for p = 1:N_PACIENTES

    paciente = PACIENTES(p);

    % Cargamos sus datos de EEG y etiquetas.
    fileName = sprintf('s%02d.mat', paciente);
    datos_paciente = fullfile(DIR_DATASET, fileName);
    load(datos_paciente); % Carga variables data y labels.

    % Inicializamos la matriz con las señales EEG por canal:
    Fs*(TSTOP-TSTART)x(N_ELECTRODOS * N_TRIALS).

```

```

for v = 1:N_TRIALS
    video = TRIALS(v);
    s_eeg_ = [];
    % Dividimos cada señal EEG en múltiples señales EEG para tener más
    % muestras.
    for m = 1:multiplicidad
        s_eeg(:,1+(v-1)*N_ELECTRODOS*multiplicidad:v*N_ELECTRODOS*multiplicidad) = ...
            s_eeg_;
    end
s_eeg = zeros(Fs*(TSTOP(1)-TSTART(1)), multiplicidad*N_ELECTRODOS*N_TRIALS);
s_eeg_ = [s_eeg_,
          reshape(...
            data(video,ELECTRODOS,1+Fs*TSTART(m):Fs*TSTOP(m)),...
            N_ELECTRODOS,...
            Fs*(TSTOP(m)-TSTART(m)))'...
          ];
end

% Queda una matriz de la siguiente forma:
% - Cada fila es un instante de tiempo de una señal EEG
% - Cada columna indica una señal EEG para un par electrodo,vÃ-deo
% - Orden: [ Video 1: e1 e2 e3 ... eN, Video 2: e1 e2 e3 eN, ... Video M: e1
%           e2 ... eN]

% Filtrado de las señales.
[s_eeg_theta, s_eeg_salpha, s_eeg_alpha, s_eeg_beta, s_eeg_gamma] = ...
    filtrarEEG(s_eeg, Fs);

% Cálculo de potencias.
[p_theta, p_salpha, p_alpha, p_beta, p_gamma, p_total] = ...
    calcularPotencias(N_ELECTRODOS, N_TRIALS_M, PARES, s_eeg_theta, ...
    s_eeg_salpha, s_eeg_alpha, s_eeg_beta, s_eeg_gamma);
end

```

- 3) Se seleccionan los índices de los trials cuyas puntuaciones son inferiores a PUNTUACION_CLASE1 y se etiquetan como muestras de la clase 1, y se seleccionan los índices de los trials cuyas puntuaciones son superiores a PUNTUACION_CLASE2 y se etiquetan como muestras de la clase 2.

```

if (tipo == "valence")
    etiquetas = etiquetas_valence;
else
    etiquetas = etiquetas_arousal;
end
CLASE1 = 0; CLASE2 = 1;

% Las muestras estarán ordenadas por paciente. Se buscan aquellas con ratings
% de acuerdo a los valores asignados a PUNTUACION_CLASE1 y PUNTUACION_CLASE2:
% Se guardan los índices en muestras_i.
muestras_i = [];
tags = [];
n_muestras_paciente = zeros(N_PACIENTES, 1);
for p = 1:N_PACIENTES
    paciente = PACIENTES(p);
    c1_i = (p-1)*N_TRIALS_M + ...
        find(etiquetas(1+(p-1)*N_TRIALS_M:p*N_TRIALS_M)<=PUNTUACION_CLASE1);
    if (PUNTUACION_CLASE1 == PUNTUACION_CLASE2)
        c2_i = (p-1)*N_TRIALS_M + ...
            find(etiquetas(1+(p-1)*N_TRIALS_M:p*N_TRIALS_M)>PUNTUACION_CLASE2);
    else
        c2_i = (p-1)*N_TRIALS_M + ...
            find(etiquetas(1+(p-1)*N_TRIALS_M:p*N_TRIALS_M)>=PUNTUACION_CLASE2);
    end
    n_muestras_paciente(p) = length(c1_i) + length(c2_i);
    muestras_i = [muestras_i; c1_i; c2_i];
    tags = [tags; repmat(CLASE1, length(c1_i), 1); ...
           repmat(CLASE2, length(c2_i), 1)];
end

```

4) En función del valor de CONJUNTO_CARACTERISTICAS, se extraen unas características diferentes.

```

switch CONJUNTO_CARACTERISTICAS
case 1
    features = [potencias_bandas(muestras_i,:), ...
                potencias_pares_bandas(muestras_i,:)];
case 2
    features = zeros(n_muestras, 10);

    features(:,1) = (potencias_beta(muestras_i,8) - ...
                    potencias_alpha(muestras_i,8)) ./ (potencias_beta(muestras_i, 1) - ...
                    potencias_alpha(muestras_i,1));

    features(:,2) = ((potencias_beta(muestras_i,8) + ...
                    potencias_beta(muestras_i, 9)) ./ (potencias_alpha(muestras_i,8) + ...
                    potencias_alpha(muestras_i,9))) - ((potencias_beta(muestras_i,1) + ...
                    potencias_beta(muestras_i, 2)) ./ (potencias_alpha(muestras_i,1) + ...
                    potencias_alpha(muestras_i,2)));

    features(:,3) = (sum(potencias_beta(muestras_i, 8:14), 2) ./ ...
                    sum(potencias_alpha(muestras_i, 8:14), 2)) - ...
                    sum(potencias_beta(muestras_i, 1:7), 2) ./ ...
                    sum(potencias_alpha(muestras_i, 1:7), 2));

    features(:,4:10) = potencias_total(muestras_i,8:14) - ...
                    potencias_total(muestras_i,1:7);
case 3
    features = [potencias_theta(muestras_i, [5 6 17 16 7]), ...
                potencias_alpha(muestras_i, [2 1 5 6 16 13]), ...
                potencias_beta(muestras_i, [3 4 10 6 9]), ...
                potencias_gamma(muestras_i, [3 14 11 13 9 15 8 12])];
case 4
    features = [potencias_bandas(muestras_i,:), ...
                potencias_pares_bandas(muestras_i,:)];
case 5
    features = zeros(n_muestras, 17);

    features(:,1) = (potencias_beta(muestras_i,8) + ...
                    potencias_beta(muestras_i,1)) ./ (potencias_alpha(muestras_i,8) + ...
                    potencias_alpha(muestras_i,1));

    features(:,2) = sum(potencias_beta(muestras_i,[8 1 9 2]), 2) ./ ...
                    sum(potencias_alpha(muestras_i,[8 1 9 2]), 2);

    features(:,3) = sum(potencias_beta(muestras_i,8:14), 2) ./ ...
                    sum(potencias_alpha(muestras_i,8:14), 2);

    elec = [1 8 2 9 3 10 4 11 5 12 6 13 7 14];
    features(:,4:17) = potencias_beta(muestras_i, elec) - ...
                    potencias_alpha(muestras_i, elec);
case 6
    features = [potencias_theta(muestras_i, [11 7 9 12]), ...
                potencias_alpha(muestras_i, [3 5 1 11]), ...
                potencias_beta(muestras_i, [3 1 10 8]), ...
                potencias_gamma(muestras_i, [2 4 7 14 13])];
otherwise
    features = [potencias_bandas(muestras_i,:), ...
                potencias_pares_bandas(muestras_i,:)];
end

```

5) Se guardan las características y etiquetas en un fichero .csv para poder posteriormente cargar el dataset generado con Python.

```
file_dataset = DIR_GUARDAR +"dataset"+ CONJUNTO_CARACTERISTICAS +"_m"+ ...
                multiplicidad +"_" + PUNTUACION_CLASE1 +"_" + PUNTUACION_CLASE2 + ".csv";

csvwrite(file_dataset, [features, tags]);
```

5.4 Técnicas utilizadas para la clasificación

A la hora de clasificar, se nos presenta un amplio abanico de opciones y algoritmos con los que realizar esta tarea.

A continuación se van a explicar brevemente los diferentes métodos y herramientas que se han utilizado en este trabajo para llevar a cabo la clasificación binaria entre valencia positiva y negativa y excitación baja y alta, que han sido los siguientes:

- Análisis Discriminante Lineal (LDA).
- Árbol de decisión y Bosques aleatorios (Random Forest).
- K vecinos más cercanos (KNN).
- Máquinas de soporte vectorial (SVM).
- Redes neuronales artificiales (ANN).
- Análisis de Componentes Principales (PCA).

5.4.1 Análisis Discriminante Lineal (LDA)

LDA es una generalización del discriminante línea de Fisher, un método utilizado en estadística, machine learning y reconocimiento de patrones para encontrar una combinación lineal de características que separen el conjunto de datos en dos clases. Esta separación puede usarse como un clasificador lineal. [42]

Un clasificador lineal basa su decisión en base al hiperplano que divide al espacio evaluado en dos, asignando a cada clase una parte. A este hiperplano se le conoce como hiperplano discriminante [45].

LDA realiza las siguientes asunciones sobre el discriminante lineal de Fisher [43]:

- 1) Las clases siguen una distribución normal.
- 2) Las covarianzas de las clases son idénticas (homocedasticidad).
- 3) La capacidad de predicción puede disminuir ante un aumento de la correlación entre las variables utilizadas para la predicción.
- 4) Los valores de una variable para un participante son independientes de los valores de esa misma variable en otros participantes.

En esencia, LDA supone que los datos que tenemos de las dos clases siguen una distribución normal, y para cada clase modela los parámetros media y varianza, con el objetivo de encontrar la distribución que mejor la describa. Una vez hecho esto, se utiliza el teorema de Bayes para calcular la probabilidad de pertenecer a cada clase y la frontera de decisión.

El teorema de Bayes nos dice lo siguiente respecto a la probabilidad de ocurrencia de los eventos A y B:

$$p(A|B) = \frac{p(B|A)P(A)}{p(B)} \quad (5.1)$$

Si queremos asignar una clase (c_i) entre un conjunto de clases a una observación (\mathbf{z}), y conocemos la probabilidad a priori de que se dé cada una de las clases ($p(C_i)$), entonces el teorema de Bayes define lo siguiente:

$$Posterior = \frac{Prior \cdot Likelihood}{Evidence} \rightarrow p(c_i|\mathbf{z}) = \frac{p(c_i)p(\mathbf{z}|c_i)}{\sum_{l=1}^c p(\mathbf{z}|c_l)} \quad (5.2)$$

Sabiendo esto, podemos tomar la decisión sobre qué clase asignar (\hat{c}) eligiendo aquella con mayor probabilidad a posteriori:

$$\hat{c} = c_i \Leftrightarrow p(c_i|\mathbf{z}) \geq p(c_l|\mathbf{z}), \forall (l \neq i) \quad (5.3)$$

Para el caso de clasificación binaria en el que tenemos dos clases, aplicando LDA tendremos:

$$Prior(c_1) = p(c_1); \text{ Prior}(c_2) = p(c_2) = 1 - p(c_1) \quad (5.4)$$

$$Likelihood = p(\mathbf{z}|c_l) \quad (5.5)$$

$$Evidence = \sum_{l=1}^2 p(c_l)p(\mathbf{z}|c_l) \quad (5.6)$$

$p(\mathbf{z}_n|c_l)$ indica la probabilidad de que una nueva observación pertenezca a la clase l .

Como se ha mencionado, LDA supone que las clases siguen una distribución normal, así que:

$$p(\mathbf{z}|c_l) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} e^{(-\frac{1}{2}(\mathbf{z}_n-\mu_l)^T\Sigma^{-1}(\mathbf{z}_n-\mu_l))} \quad (5.7)$$

Por lo que el criterio de decisión para determinar si una observación \mathbf{z}_n pertenece a la clase 1 será que se cumpla lo siguiente:

$$p(c_1)p(\mathbf{z}_n|c_1) \geq (1 - p(c_1))p(\mathbf{z}_n|c_2) \quad (5.8)$$

En caso contrario, se decidirá que la observación \mathbf{z}_n pertenece a la clase 2.

Lo más interesante es ahora encontrar la frontera que separa a las dos regiones de decisión, lo cual podemos hacer si en (4.8) evaluamos la igualdad entre ambas partes de la ecuación. Desarrollando dicha igualdad:

$$1 = \frac{(1 - p(c_1))p(\mathbf{z}_n|c_2)}{p(c_1)p(\mathbf{z}_n|c_1)} \quad (5.9)$$

$$1 = \frac{(1 - p(c_1))e^{j(-\frac{1}{2}(\mathbf{z}_n-\mu_2)^T\Sigma^{-1}(\mathbf{z}_n-\mu_2))}}{p(c_1)e^{j(-\frac{1}{2}(\mathbf{z}_n-\mu_1)^T\Sigma^{-1}(\mathbf{z}_n-\mu_1))}} \quad (5.10)$$

$$0 = \log\left(\frac{(1 - p(c_1))e^{j(-\frac{1}{2}(\mathbf{z}_n-\mu_2)^T\Sigma^{-1}(\mathbf{z}_n-\mu_2))}}{p(c_1)e^{j(-\frac{1}{2}(\mathbf{z}_n-\mu_1)^T\Sigma^{-1}(\mathbf{z}_n-\mu_1))}}\right) \quad (5.11)$$

$$0 = \log\left(\frac{1 - p(c_1)}{p(c_1)}\right) - \frac{1}{2}(\mathbf{z}_n - \mu_2)^T\Sigma^{-1}(\mathbf{z}_n - \mu_2) + \frac{1}{2}(\mathbf{z}_n - \mu_1)^T\Sigma^{-1}(\mathbf{z}_n - \mu_1) \quad (5.12)$$

$$0 = (\mu_2 - \mu_1)^T\Sigma^{-1}\mathbf{z}_n - \frac{1}{2}\mu_2^T\Sigma^{-1}\mu_2 + \frac{1}{2}\mu_1^T\Sigma^{-1}\mu_1 + \log\left(\frac{1 - p(c_1)}{p(c_1)}\right) \quad (5.13)$$

Llegamos a la ecuación (4.13) que se corresponde con el hiperplano que separa a las dos clases. La ecuación general de un hiperplano viene dada por:

$$0 = \mathbf{a}z_n + \mathbf{b} \quad (5.14)$$

donde \mathbf{a} son los coeficientes y \mathbf{b} el término independiente.

En este problema, tenemos que \mathbf{a} y \mathbf{b} son:

$$\mathbf{a} = (\mu_2 - \mu_1)^T \Sigma^{-1} \quad (5.15)$$

$$\mathbf{b} = -\frac{1}{2}\mu_2^T \Sigma^{-1} \mu_2 + \frac{1}{2}\mu_1^T \Sigma^{-1} \mu_1 + \log\left(\frac{1 - p(c_1)}{p(c_1)}\right) \rightarrow \quad (5.16)$$

$$\rightarrow \mathbf{b} = \frac{\mathbf{a}^T(\mu_1 + \mu_2)}{2} + \log\left(\frac{1 - p(c_1)}{p(c_1)}\right) \quad (5.17)$$

Y la estimación de la clase de una observación se puede realizar de la siguiente forma:

$$\hat{c} = \text{sgn}(\mathbf{a}^T z_n + \mathbf{b}) \quad (5.18)$$

A continuación se muestra la separación de un conjunto de datos en dos mediante el trazado del hiperplano discriminante, que sería en este caso la recta de color negro.

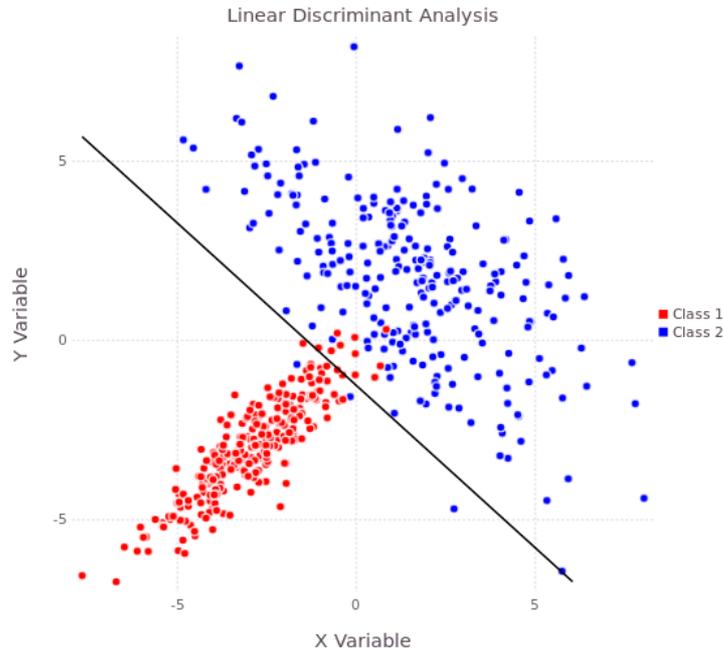


Figura 18: Separación del conjunto de datos usando LDA. Fuente: [44].

Es inmediato entrever que si nuestro conjunto de datos no es linealmente separable, utilizar LDA no proporcionará una clasificación totalmente precisa.

Para la implementación de LDA se ha utilizado Matlab tomando como referencia el siguiente pseudocódigo [46].

```

LEARNING THE LDA (BINARY) CLASSIFIER
% Using the training pairs (c(τ), f_τ)
μ_k = 1/N_{c_k} ∑_{τ:c(τ)=c_k} f_τ for k = 1, 2.
Σ_{f|c_k} = 1/N_{c_k} ∑_{τ:c(τ)=c_k} (f_τ - μ_k)(f_τ - μ_k)^T, k = 1, 2.
p(c_k) = N_{c_k}/(N_{c_1} + N_{c_2}), k = 1, 2.
Σ_f = p(c_1)Σ_{f|c_1} + p(c_2)Σ_{f|c_2}
α = Σ_f^{-1} (μ_1 - μ_2)
β = 1/2(μ_1 + μ_2) - (log p(c_1) - log p(c_2)) / ((μ_1 - μ_2)^T Σ_f^{-1} (μ_1 - μ_2)) (μ_1 - μ_2)

CLASSIFICATION OF TEST DATA
% Implementation using LDA with equal cov.
C_{Y_τ} = W^T C_{X_τ}^{(0)} W, τ ∈ Set of test trials.
Evaluate f_τ using the same formula as in (F1)
ĉ(τ) = c_k where k = 1.5 + 0.5 sign(α^T (f_τ - β)).

```

Figura 19: Pseudocódigo para la implementación de LDA. Fuente: [46].

5.4.2 Árbol de decisión y Bosques aleatorios (Random Forest)

El clasificador basado en árbol de decisión consiste en una división iterativa del conjunto de datos de entrenamiento, separando el conjunto en dos en cada división.

Es una técnica de clasificación supervisada y sirve para clasificación tanto binaria como multiclase. Aunque el entrenamiento pueda llegar a ser computacionalmente costoso cuando la dimensión de los datos es alta, las predicciones suelen ser rápidas.

Es necesario ajustar los siguientes hiperparámetros para evitar el sobreajuste:

- Profundidad máxima.
- Número mínimo de muestras.

La siguiente figura ilustra un ejemplo de clasificación binaria mediante árbol de decisión. En este caso se tiene un árbol de profundidad máxima (coincide con la dimensión de los datos: 3).

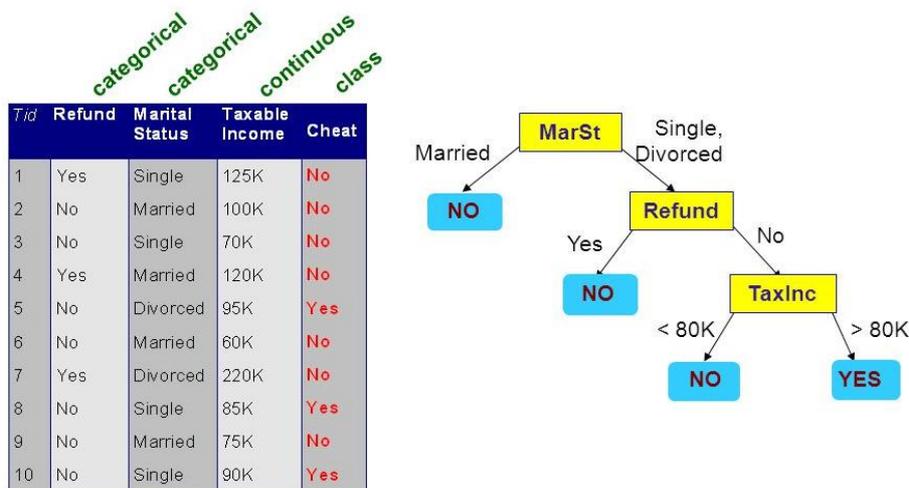


Figura 20: Ejemplo de árbol de decisión. Fuente: [47].

5.4.2.1 Bosques Aleatorios (Random Forest)

Random forest consiste en una combinación de árboles de decisión tal que cada árbol depende de los valores de un subconjunto del conjunto características, probado independientemente y con la misma distribución para cada uno de estos.

Es una modificación sustancial de bagging¹ que construye una larga colección de árboles no correlacionados y luego los promedia. Los árboles son candidatos ideales para el bagging, dado que pueden registrar estructuras de interacción compleja en los datos, y si alcanzan una profundidad suficiente, la parcialidad que tienen es relativamente baja. Puesto que los árboles son notoriamente ruidosos, se benefician enormemente al promediar.

Cada árbol de decisión que forma parte del bosque aleatorio se construye usando el siguiente algoritmo:

- 1) Sea N el número de casos de prueba (tamaño del conjunto de entrenamiento) y M el número de características.
- 2) Sea m el número de variables (características) de entrada a ser usado para determinar la decisión en un nodo dado. $m \ll M$.
- 3) Se elige un subconjunto de entrenamiento para este árbol y se usa el resto de los casos de prueba para estimar el error.
- 4) Para cada nodo del árbol, se eligen aleatoriamente m variables (características) en las cuales basar la decisión. Se calcula la mejor partición del conjunto de entrenamiento a partir de las m variables.

Para la predicción de la clase de una observación, el conjunto de características son empujadas hacia abajo por cada árbol y se le asigna la etiqueta del nodo terminal donde termina. Este proceso es iterado por todos los árboles en el ensamblado, y la etiqueta que obtenga la mayor cantidad de ocurrencias es la que se da como resultado de la predicción [48].

Para implementar Random Forest se ha utilizado el módulo `sklearn.ensemble` en Python.

5.4.3 K vecinos más cercanos (KNN)

El método de los K vecinos más cercanos es un método de clasificación supervisada que permite llevar a cabo una clasificación tanto binaria como multiclase.

El entrenamiento consiste únicamente en almacenar todas las muestras del conjunto de entrenamiento, y para clasificar se considera la distancia euclídea a los K vecinos (puntos del conjunto de entrenamiento) más cercanos, usando como predicción la etiqueta preponderante [49].

Dados dos puntos X e Y , ambos de dimensión N , la distancia euclídea entre ambos, $d(X, Y)$, se define como:

$$d(X, Y) = \sqrt{\sum_{i=1}^N (y_i - x_i)^2} \quad (5.19)$$

De forma que la estimación de la clase de una observación \mathbf{z} se puede realizar de la siguiente forma:

$$\hat{c}_{\mathbf{z}} = \operatorname{argmax}_{v \in V_k} \sum_{i=1}^k \delta(c_v, c_i) \quad (5.20)$$

¹ La idea esencial del bagging es promediar muchos modelos ruidosos pero aproximadamente imparciales, y por tanto reducir la variación.

Donde $\delta(c_v, c_i) = 1$ si $c_v = c_i$; y 0 en cualquier otro caso. V_k son los K vecinos más cercanos a la observación \mathbf{z} , c_v es la clase del vecino v y c_i la clase del vecino i -ésimo.

Aunque es un clasificador bastante sencillo, ofrece resultados muy buenos. K es un hiperparámetro y debe ser ajustado:

- Si es muy pequeño, no tendrá efecto sobre el clasificador del vecino más cercano (NN), el cual es sensible a datos ruidosos.
- Si es muy grande, tenderá a dar como solución la clase más probable.

La figura siguiente ilustra cómo clasificaría un clasificador KNN con $K = 1$ y a partir de un conjunto de datos de entrenamiento con tres clases.

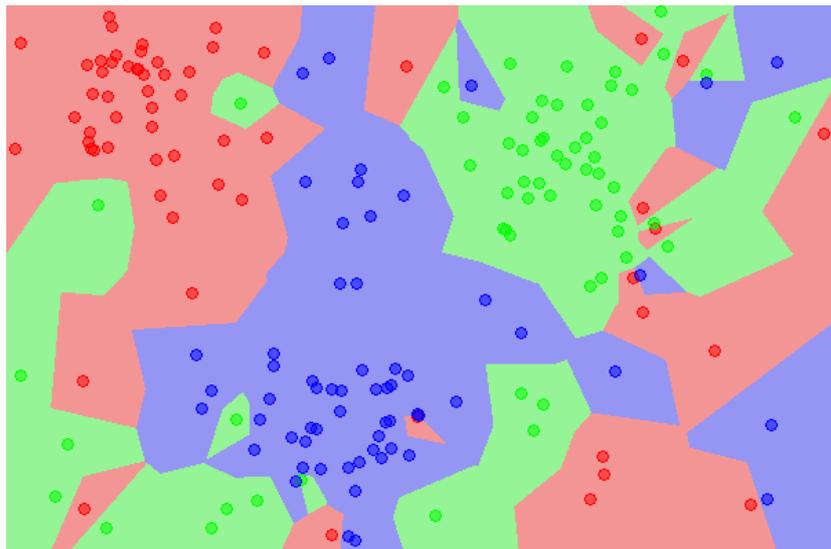


Figura 21: División del espacio de características en subespacios correspondientes a cada clase con KNN. Fuente: [50].

Para implementar este clasificador se ha utilizado el módulo `sklearn.neighbors` en Python.

5.4.4 Máquinas de soporte vectorial (SVM)

Las máquinas de soporte vectorial (Support Vector Machine, SVM) son sistemas de aprendizaje supervisado que pertenecen a la rama de los clasificadores lineales generalizados. Aunque a priori sólo permiten la clasificación binaria, pueden extenderse a clasificación multiclase mediante la agrupación de múltiples SVMs [51].

Las SVMs permiten definir hiperplanos que separan el espacio de características en dos, y pueden aplicarse a espacios no separables mediante el uso de kernels especiales (SVMs no lineales).

Hay ocasiones en las que existen múltiples fronteras que separan el espacio de datos en dos, y puede definirse lo que se llama margen: grosor máximo que puede tomar la frontera hasta tocar alguno de los puntos. El criterio que siguen las SVM es maximizar dicho margen, lo que implica que los ejemplos de interés para definir el clasificador son los que se encuentran cerca de este, y a estos ejemplos se les llama vectores de soporte [51].

La figura de a continuación ilustra estos conceptos.

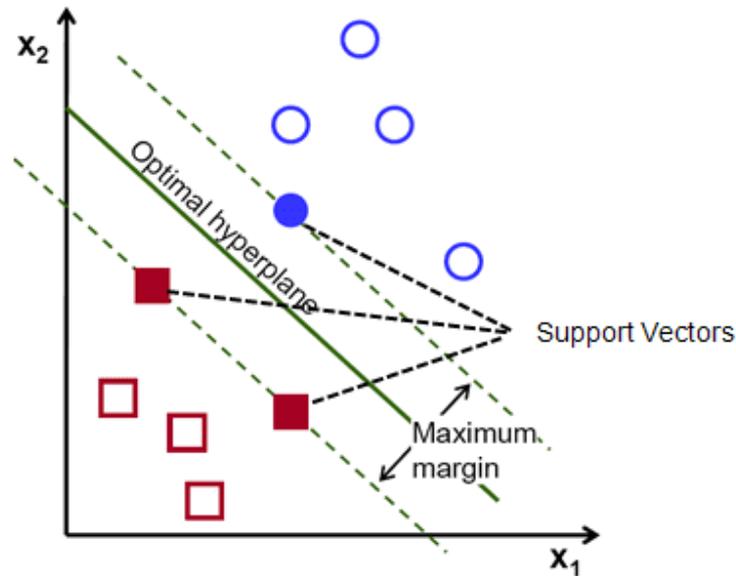


Figura 22: Máquinas de soporte vectorial: Vectores de soporte, hiperplano óptimo y margen máximo. Fuente: [52].

Cuando el conjunto de datos es separable, las SVM lineales (LSVM) funcionan muy bien. En el caso de que el conjunto de datos no sea linealmente separable, no podremos aplicar una LSVM.

Para el caso en el que el conjunto de datos no sea separable, se propuso un nuevo tipo de clasificador basado en SVM: una SVM no lineal (NLSVM). Una NLSVM consta de una transformación previa del conjunto de datos original a través de una función no lineal conocida como kernel. El objetivo de esta transformación es mapear el conjunto de datos a un espacio de dimensión mayor de modo que sí sea separable. La siguiente figura ilustra este procedimiento.

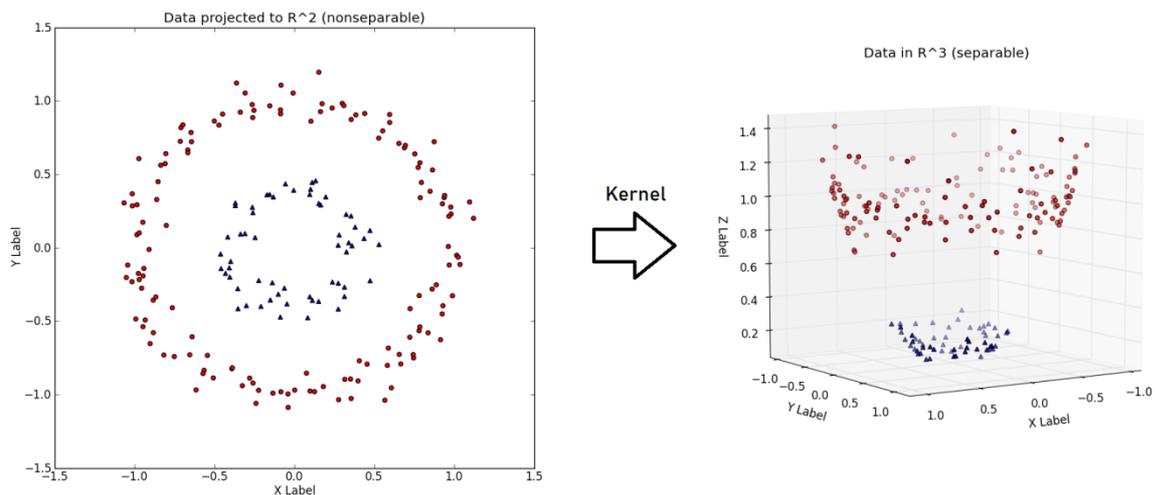


Figura 23: Transformación del conjunto de datos mediante un kernel. Fuente: [53].

Una vez se han transformado los datos mediante el kernel, se procede de igual forma que con una LSVM.

Para implementar este clasificador, se ha utilizado el módulo `sklearn.svm` en Python.

5.4.5 Redes neuronales artificiales (ANN)

Las redes neuronales artificiales modelan un tipo clasificador biológicamente inspirado. El elemento básico de computación del cerebro humano es la neurona. Cada neurona recibe entradas por sus dendritas y genera una

única salida a través de su axón, el cual puede dividirse múltiples veces y conectarse mediante sinapsis a las dendritas de otras neuronas.

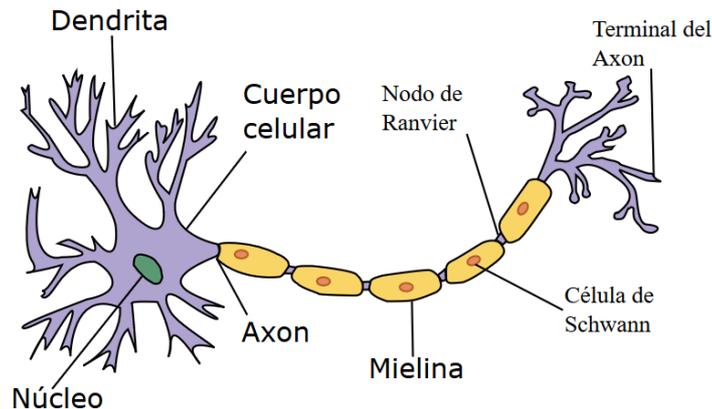


Figura 24: Partes de una neurona. Fuente [54].

Las redes neuronales artificiales unidireccionales se modelan como un conjunto de neuronas conectadas entre sí formando un grafo sin bucles, a diferencia del modelo cerebral de interconexión en el que las neuronas están conectadas de forma aparentemente aleatoria y amorfa.

Las redes neuronales se organizan en capas de neuronas, y la capa más común es la capa completamente conectada: cada neurona de dicha capa se conecta a todas las neuronas de la capa anterior y no se producen conexiones entre neuronas de la misma capa [55].

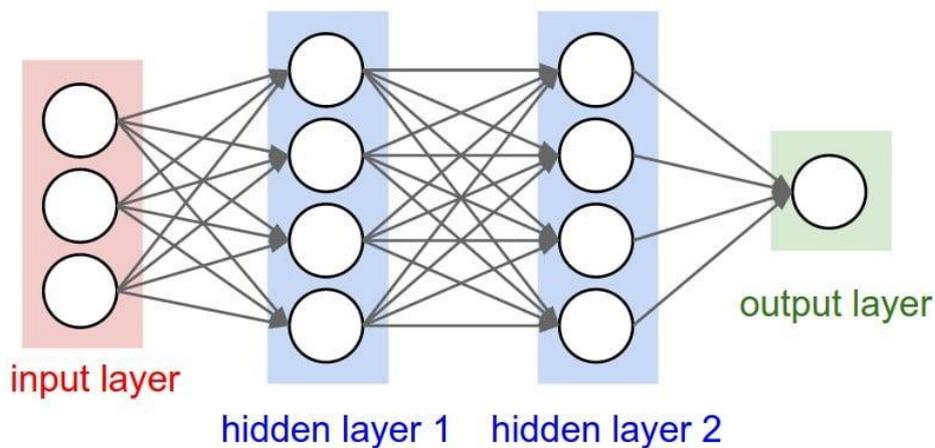


Figura 25: Red neuronal con 4 capas. Fuente: [56].

Toda red neuronal debe tener al menos las capas de entrada y salida, y el número de capas ocultas variará según el problema.

En el modelo computacional neuronal, las señales que produce un axón interaccionan en forma de multiplicación por un peso con las dendritas. Cada neurona k recibe una serie de entradas correspondientes a las salidas de las n neuronas previas que se conectan a ella (x_1, x_2, \dots, x_n) y que se ven moduladas cada una por un peso ($w_{k1}, w_{k2}, \dots, w_{kn}$), a las que posteriormente se le suma un sesgo (b_k). La salida de la neurona k se calcula mediante una función de activación a partir de la suma de las entradas moduladas por los pesos y la suma del sesgo:

$$Y_k = \varphi \left(b_k + \sum_{j=1}^n x_j w_{kj} \right) \quad (5.21)$$

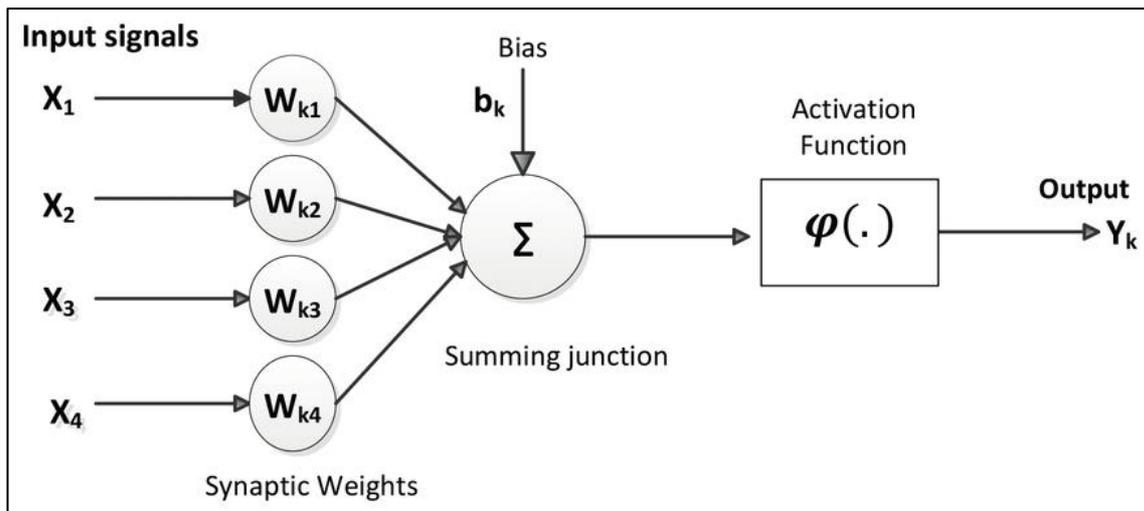


Figura 26: Modelo computacional de una neurona. Fuente: [57].

Existen diferentes funciones de activación para las redes neuronales, y es uno de los parámetros a la hora de crear nuestro modelo. Algunas de las más comunes son: sigmoid, relu, tanh, linear... El propósito de las funciones de activación es dotar al modelo de no linealidad, haciéndolo capaz de aprender y realizar tareas más complejas. Una red neuronal sin funciones de activación no sería más que un modelo de regresión.

El entrenamiento de una red neuronal consiste en el aprendizaje de los pesos y sesgos de cada una de las neuronas que conforman el modelo. Existen múltiples métodos para llevar a cabo este aprendizaje, y el más popular es el algoritmo de retroproyección filtrada, el cual se basa en la regla de la cadena e itera desde la salida hacia atrás, ajustando los pesos y sesgos de las neuronas de forma que se minimice el error a la salida.

Otros parámetros para el modelo de una red neuronal serían:

- **Número de épocas (“epochs”)**: Número de veces que se presenta el conjunto de datos de entrenamiento al modelo durante el aprendizaje. Un número muy alto produciría sobreajuste.
- **Tamaño del lote (“batch size”)**: Cantidad de datos del conjunto de entrenamiento que usa en cada iteración del aprendizaje.

Por ejemplo, si tenemos un conjunto de datos de entrenamiento de 12 muestras y fijamos un número de épocas de 5 y un tamaño de lote de 4, en total nuestro modelo realizaría $5 * (12/4) = 20$ iteraciones durante el aprendizaje.

Para la implementación de redes neuronales se ha utilizado el módulo `keras` en Python.

5.4.6 Análisis de Componentes Principales (PCA)

Muchos algoritmos de inteligencia artificial no son eficientes cuando los datos tienen una dimensión muy alta: son costosos de calcular o directamente no funcionan bien. Además, es habitual que cuando se dispone de un conjunto de características grande, la dimensión intrínseca de la información sea inferior: puede que sólo un subconjunto de estas sea realmente representativo.

PCA (Análisis de Componentes Principales) es una técnica para la reducción de dimensionalidad de un conjunto de datos. Consiste en una transformación lineal de los datos a una dimensión inferior, manteniendo la mayor información posible (varianza) de los datos.

El objetivo de PCA es resumir un conjunto de datos multidimensional con otro de dimensionalidad inferior derivado del original. La idea fundamental es encontrar un conjunto de ejes en torno al cual los datos muestren su mayor variación y proyectar los datos sobre él [58].

Grosso modo, PCA utiliza la descomposición en valores singulares y calcula una matriz de transformación de forma que cada columna de los datos transformados explique una cantidad mayor de información que la

siguiente, de forma que quedándonos con las m primeras columnas tengamos el conjunto de dimensión m que conserve la mayor cantidad de información de los datos originales (dimensión n , $m < n$).

Para aplicar PCA, se utiliza el módulo `sklearn.decomposition` en Python.

5.5 Conclusiones

En este capítulo se han explicado las técnicas utilizadas para la extracción de características a partir de las señales EEG y las técnicas utilizadas para la clasificación binaria de la valencia y grado de excitación a partir de las características anteriores.

Se han extraído un total de 6 conjuntos de características diferentes, 3 para la clasificación de la valencia y otros 3 para la clasificación del grado de excitación. A estos conjuntos se les irá haciendo referencia a lo largo de lo que sigue de memoria como conjuntos 1, 2, 3, 4, 5 y 6. Se ha visto que el proceso de extracción de características se ha llevado a cabo utilizando matlab, mediante la implementación de unos scripts.

Los diferentes clasificadores que se han utilizado para la clasificación binaria de la valencia y del grado de excitación son los siguientes:

- Análisis Discriminante Lineal (LDA)
- Bosques aleatorios (Random Forest)
- K vecinos más cercanos (KNN)
- Máquinas de soporte vectorial (SVM)
- Redes neuronales artificiales (ANN)

6 SIMULACIONES

En este capítulo se van a comentar los resultados obtenidos al realizar la clasificación binaria entre valencia positiva y negativa y la clasificación binaria entre grado de excitación alto y bajo. Para ello, se han utilizado los conjuntos de características mencionados en el punto 5.1 del capítulo anterior y los siguientes algoritmos de clasificación: K vecinos más cercanos (KNN), máquinas de soporte vectorial (SVM), bosques aleatorios (Random Forest).

Para estudiar el desempeño de los diferentes clasificadores utilizados, se han llevado a cabo los diferentes análisis:

1) Análisis de enventanado.

En este análisis se utiliza cada conjunto de características con un número y tamaño de ventanas diferente.

Tabla 12: Enventanados utilizados en las simulaciones.

Enventanado	Nº ventanas	Solape	Ventanas
Enventanado 1	1	-	1) Del segundo 33 al 63.
Enventanado 2	2	5 segundos	1) Del segundo 28 al 48.
			2) Del segundo 43 al 63.
Enventanado 3	3	2 segundos	1) Del segundo 25 al 39.
			2) Del segundo 37 al 51.
			3) Del segundo 49 al 63.

2) Análisis de los parámetros de los clasificadores.

Se analizan los resultados variando el valor de los parámetros de los clasificadores:

- En KNN se varía el valor de K.
- En Random Forest se varía el valor de la profundidad máxima del árbol.
- En SVM se prueban diferentes kernels y se varían los valores de C.

3) Análisis de reducción de dimensionalidad utilizando PCA.

Mediante este análisis, se reduce la dimensión del conjunto de características utilizando PCA y diferentes números de componentes. Se estudia si reducir la dimensión del conjunto de datos o simplemente aplicar PCA supone una mejora a la hora de clasificar.

4) Análisis de outliers (valores atípicos).

Los outliers o valores atípicos se introducen en el conjunto de entrenamiento variando la etiqueta de un porcentaje de los datos de dicho conjunto. Se busca que el clasificador entrene con datos cuya etiqueta está mal para ver cómo de robusto es ante la presencia de outliers.

Cada uno de estos análisis conlleva a su vez la realización de múltiples simulaciones variando la semilla aleatoria. A esto se le llama análisis de Montecarlo, y cada semilla aleatoria provoca los conjuntos de

entrenamiento y validación sean diferentes en cada simulación. En el caso de Random Forest, también provoca un estado aleatorio diferente para la inicialización del bosque de árboles.

Se realiza un análisis de Montecarlo cada vez que se varía el valor de un parámetro en los diferentes análisis (porcentaje de outliers, número de componentes en PCA, etc.) y los resultados obtenidos con cada semilla aleatoria posteriormente se promedian y es el resultado promedio el que se considera para dicho parámetro. De esta forma se consigue hacer una validación cruzada.

El capítulo se va a dividir en 4 puntos, uno correspondiente a cada uno de los clasificadores analizados (KNN, SVM y Random Forest) y otro en el que se comentan otros clasificadores que se han probado pero no de forma exhaustiva (LDA, redes neuronales).

6.1 Clasificación mediante KNN

El único parámetro del clasificador KNN es K, e indica el número de vecinos a considerar a la hora de clasificar. Para implementar este clasificador, se ha utilizado el módulo `sklearn.neighbors` en Python.

A continuación se comentan los resultados obtenidos utilizando KNN para la clasificación binaria de la valencia y del grado de excitación.

6.1.1 Clasificación de la valencia

En primer lugar, realizamos un análisis de enventanado en el que se prueban los conjuntos 1, 2 y 3 con los diferentes tipos de enventanado indicados al principio del capítulo. La siguiente figura ilustra los resultados obtenidos a partir de este análisis.

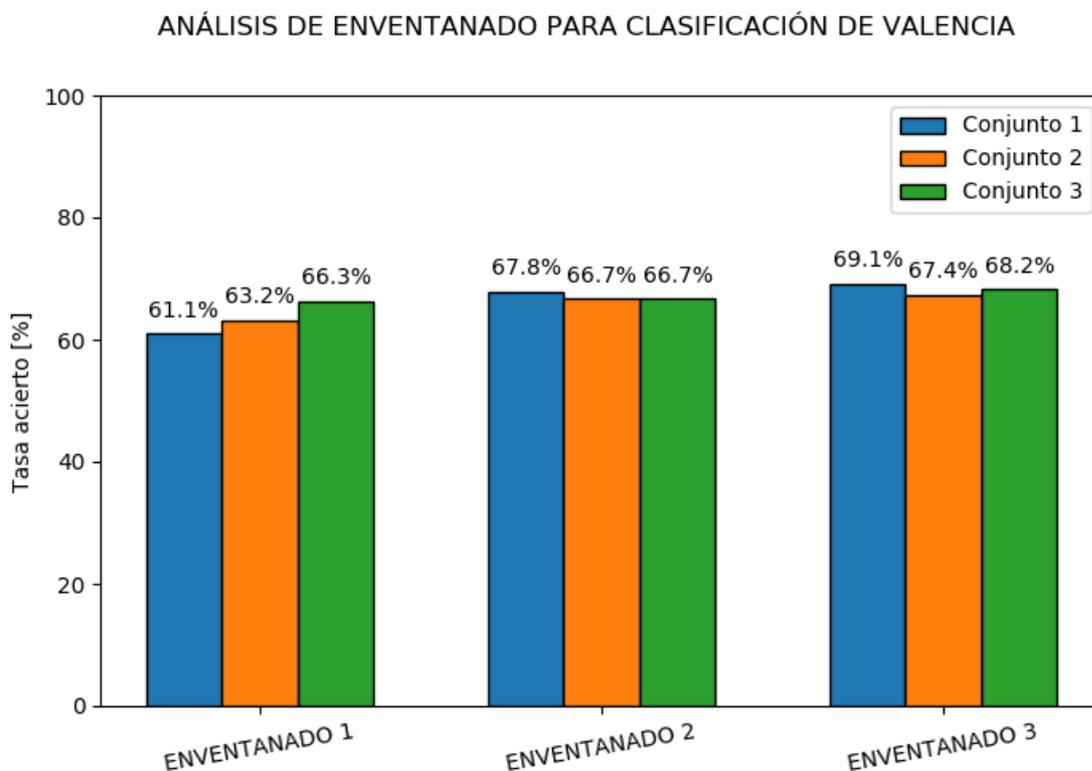


Figura 27: Análisis de las diferentes ventanas para la clasificación de la valencia usando KNN.

Se observa que la mejor tasa de acierto para la clasificación binaria de la valencia mediante KNN es de un 69.1% y se obtiene con el conjunto 1 y enventanado 3.

El enventanado 3 consistía en 3 ventanas de 14 segundos con solape de 2 segundos. Cada tipo de enventanado aumentaba el número de ventanas respecto al anterior y reducía el tamaño de estas, y donde se ve una mejora más significativa es en el conjunto 1.

Además, recordemos que el conjunto 3 (24 características) es un subconjunto del conjunto 1 (230 características) y ofrece una tasa de acierto sólo un 1% inferior a la del conjunto 1, por lo que podemos afirmar que es un resumen extremadamente bueno del conjunto 1.

En los análisis que siguen, se utilizará el conjunto 1 con inventariado 3, pues es la combinación que da el mejor resultado.

Análisis del parámetro K

En este análisis se varía el valor del parámetro K. La figura de a continuación ilustra los resultados obtenidos a partir de este análisis.

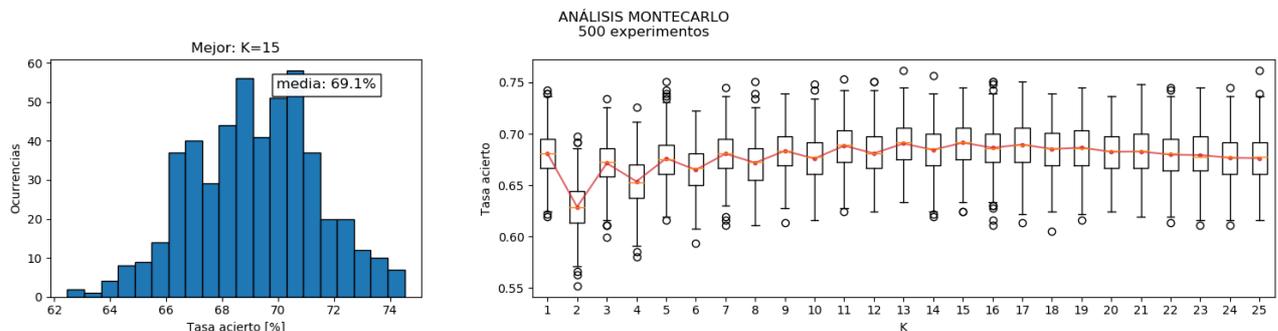


Figura 28: Análisis del parámetro K en la clasificación de la valencia con KNN.

Como podemos observar, un valor de 15 para el parámetro K nos proporciona una tasa de acierto de un 69.1%. El valor de la tasa de acierto crece en función de K hasta los valores 13, 15 y 17, que ofrecen una tasa de acierto de entorno al 69%, y a partir de ahí comienza a bajar el rendimiento del clasificador.

Análisis de reducción de dimensionalidad con PCA

En este análisis se prueba a aplicar PCA sobre el conjunto de datos, quedándonos en cada experimento con un número de componentes diferente. La siguiente figura muestra los resultados de este análisis.

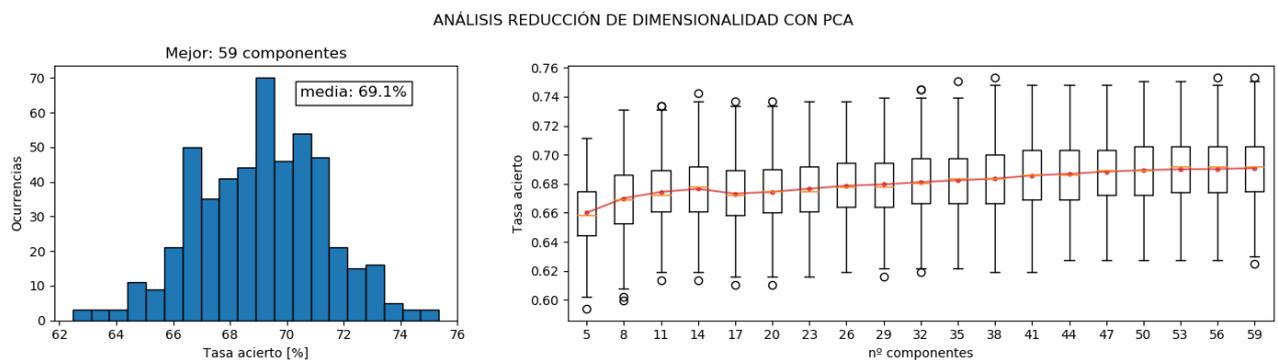


Figura 29: Análisis de la reducción del espacio de características con PCA en la clasificación de la valencia con KNN.

Se observa cómo la tasa de acierto crece a medida que nos quedamos con más componentes. A partir de 53 componentes comienza a saturar y a alcanzarse la tasa máxima de 69.1%. Además, es interesante el hecho de que con sólo 5 componentes se obtiene una tasa de acierto del 66% (recordemos que el conjunto utilizado tiene 230 características).

Se concluye que aplicar PCA en el conjunto de características 1 y con KNN resulta útil. Aunque la tasa de acierto no mejora, podemos obtener la misma pero con un conjunto de datos de una dimensión considerablemente menor.

Análisis de outliers

Para ver cómo de robusto es este clasificador, a un porcentaje de los datos de entrenamiento se les cambia la etiqueta (convirtiendo dichas muestras en muestras erróneas). La siguiente figura muestra cómo varía la tasa de acierto en función del porcentaje de outliers introducido.

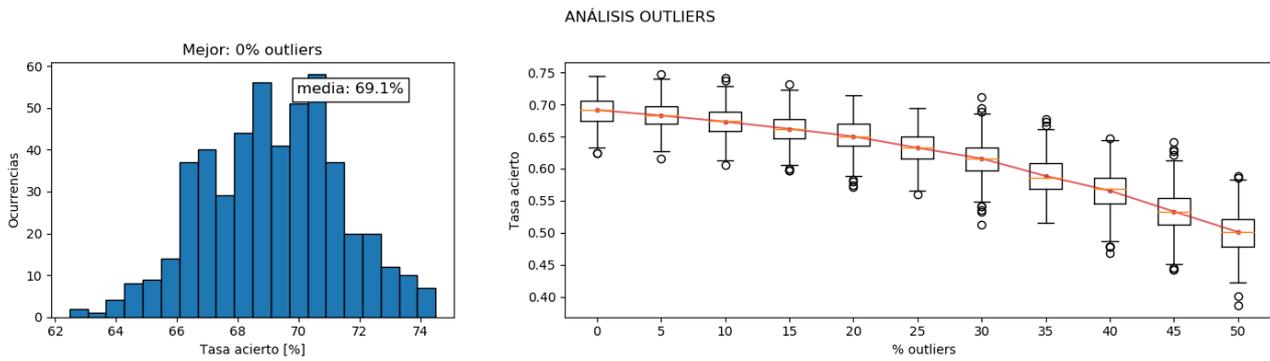


Figura 30: Análisis de outliers en la clasificación de la valencia con KNN.

Como era de esperar, la tasa de acierto baja a medida que se introducen más outliers. Cuando la mitad del conjunto de datos de entrenamiento tiene etiquetas erróneas, la tasa de acierto es de un 50%.

6.1.2 Clasificación del grado de excitación

Del mismo modo que para la clasificación de la valencia, realizamos en primer lugar un análisis de enventanado cuyos resultados se muestran en la siguiente figura

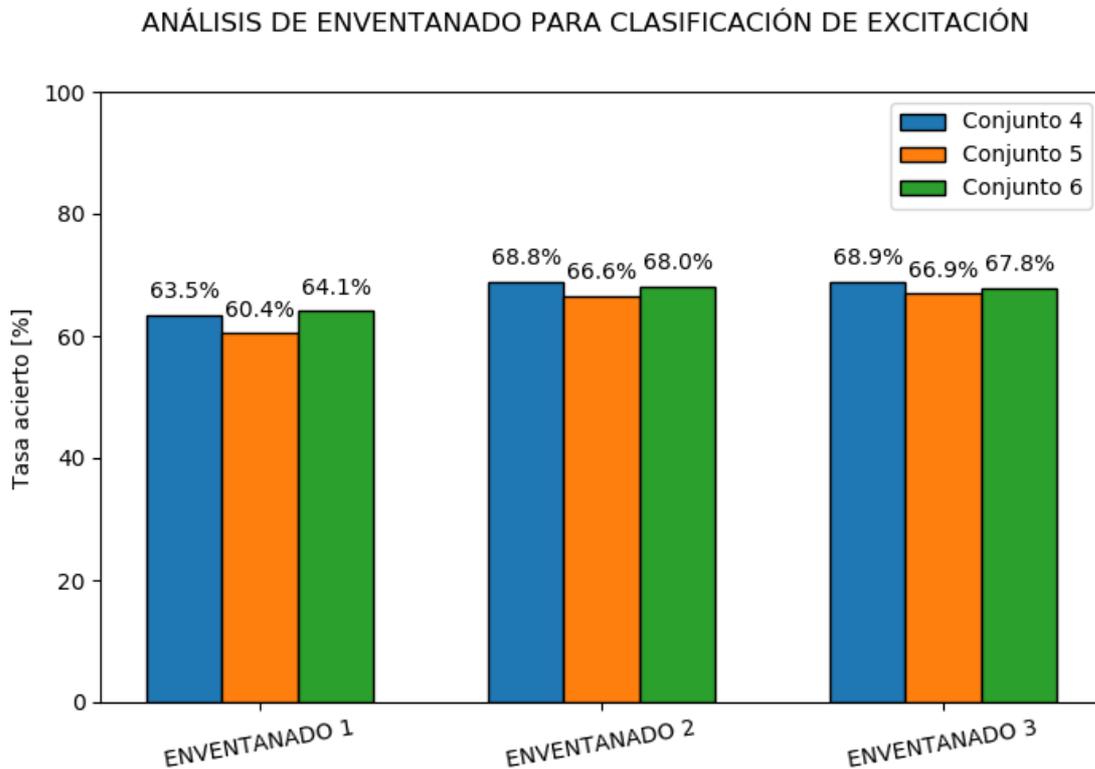


Figura 31: Análisis de las diferentes ventanas para la clasificación del grado de excitación usando KNN.

Los resultados son similares al caso de clasificación de la valencia, y la mejor tasa de acierto para la clasificación binaria del grado de excitación mediante KNN es de un 68.9% y se obtiene con el conjunto 4 y enventanado 3. No obstante, los resultados ofrecidos por los enventanados 2 y 3 son prácticamente idénticos y ambos mejoran significativamente respecto al enventanado 1.

Del mismo modo que ocurría en el caso de la valencia, se observa que el conjunto 6 (17 características) es un resumen increíblemente bueno del conjunto 1 (230 características).

En los análisis que siguen, se utilizará el conjunto 4 con enventanado 3, pues es la combinación que da el mejor resultado.

Análisis del parámetro K

Del mismo modo que para la clasificación de la valencia, se realiza un análisis variando el valor del parámetro K. La figura de a continuación ilustra los resultados obtenidos a partir de este análisis.

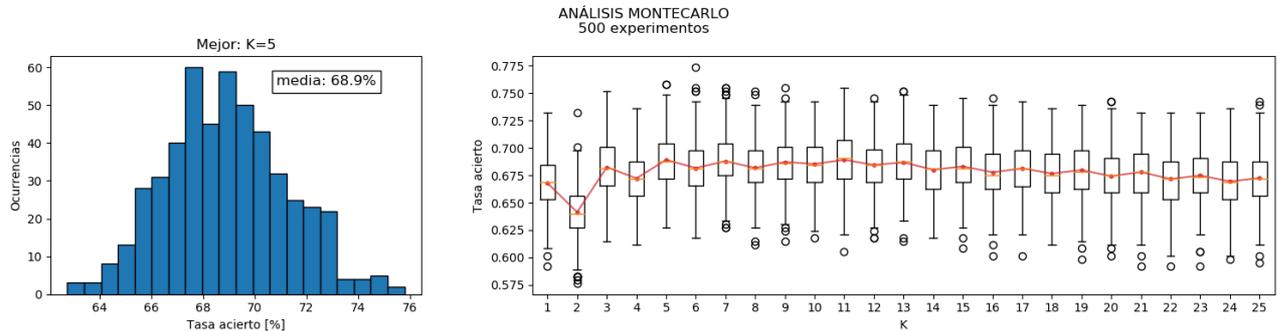


Figura 32: Análisis del parámetro K en la clasificación del grado de excitación con KNN.

Un valor de 5 para el parámetro K nos proporciona una tasa de acierto de un 68.9%. El valor de la tasa de acierto es parecido para los valores de K 5, 7, 9, y 11, y a partir de 13 comienza a bajar el rendimiento del clasificador.

Análisis de reducción de dimensionalidad con PCA

La siguiente figura muestra los resultados de este análisis.

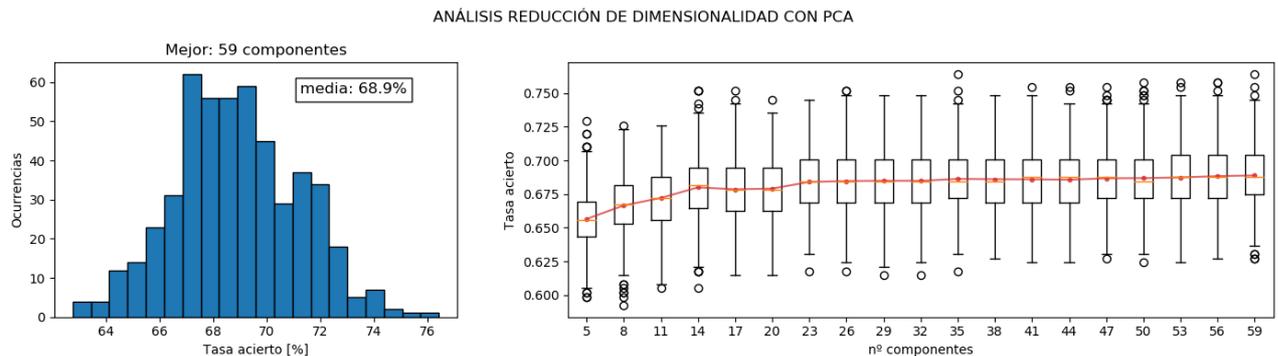


Figura 33: Análisis de la reducción del espacio de características con PCA en la clasificación del grado de excitación con KNN.

Los resultados obtenidos son bastante similares a los obtenidos en el caso de clasificación de la valencia. La tasa de acierto crece a medida que nos quedamos con más componentes, llegando a alcanzarse la tasa máxima de 68.9%. Además, en este caso la tasa de acierto obtenida comienza a saturar en un número de componentes menor, 23, en la que se alcanza un 68.7%.

Por las mismas razones que en el caso de clasificación de la valencia, se concluye que aplicar PCA en el conjunto de características 4 y con KNN resulta útil.

Análisis de outliers

La siguiente figura muestra cómo varía la tasa de acierto en función del porcentaje de outliers introducido.

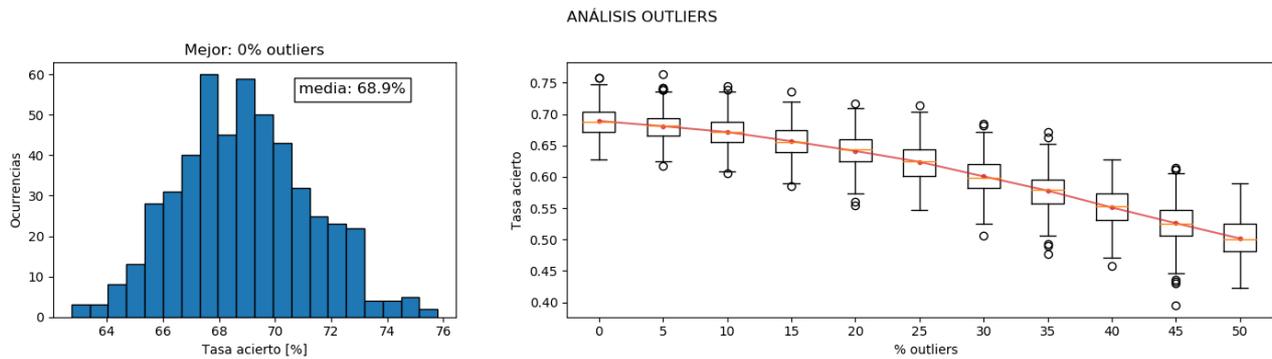


Figura 34: Análisis de outliers en la clasificación del grado de excitación con KNN.

Los resultados son similares al caso de clasificación de la valencia, y cuando la mitad del conjunto de datos de entrenamiento tiene etiquetas erróneas, la tasa de acierto es de un 50%.

6.2 Clasificación mediante Random Forest

El único parámetro del clasificador KNN es K, e indica el número de vecinos a considerar a la hora de clasificar. Para implementar este clasificador, se ha utilizado el módulo `sklearn.ensemble` en Python.

A continuación se comentan los resultados obtenidos utilizando Random Forest para la clasificación binaria de la valencia y del grado de excitación.

6.2.1 Clasificación de la valencia

En primer lugar, realizamos el análisis de enventanado para los conjuntos 1, 2 y 3. La siguiente figura ilustra los resultados del mismo.

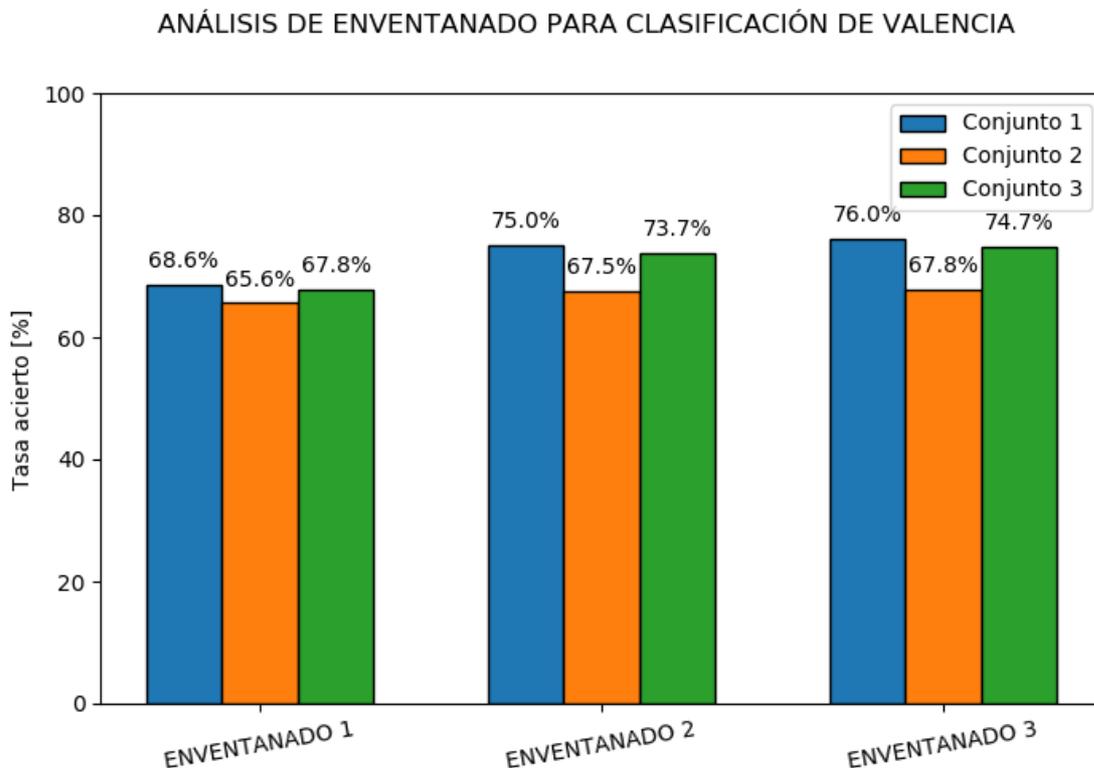


Figura 35: Análisis de las diferentes ventanas para la clasificación de la valencia usando Random Forest.

Se observa que la mejor tasa de acierto para la clasificación binaria de la valencia mediante Random Forest es de un 76% y se obtiene con el conjunto 1 y enventanado 3.

Los conjuntos 1 y 3 arrojan resultados notablemente superiores a los del conjunto 2 cuando se usan los eventanados 2 y 3. Además, se observa de nuevo que el conjunto 3 es un resumen excelente del conjunto 1.

En los análisis que siguen, se utilizará el conjunto 1 con eventanado 3, pues es la combinación que da el mejor resultado.

Análisis del parámetro profundidad máxima

En este análisis se varía el valor del parámetro profundidad máxima. La figura de a continuación ilustra los resultados obtenidos a partir de este análisis.

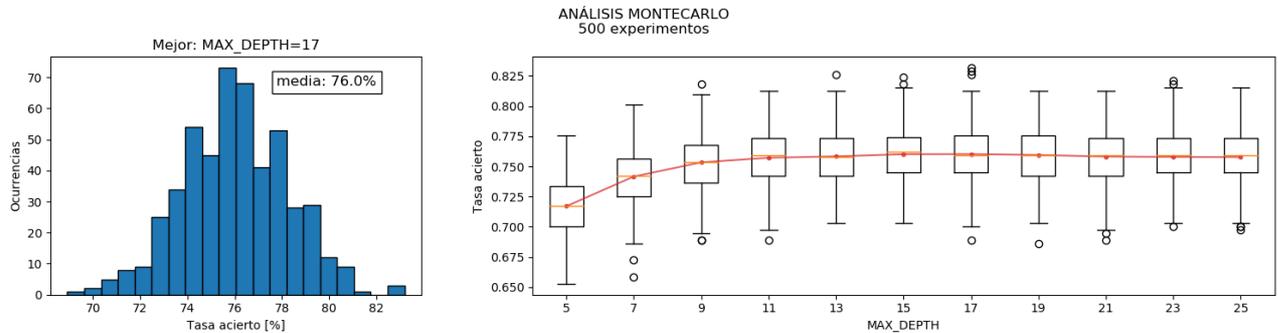


Figura 36: Análisis del parámetro K en la clasificación de la valencia con Random Forest.

Como podemos observar, un valor de 17 para el parámetro profundidad máxima nos proporciona una tasa de acierto de un 76%. El valor de la tasa de acierto crece en función de este parámetro hasta que este toma el valor 13, a partir del cual la tasa de acierto se mantiene en torno al 75.8%.

Análisis de reducción de dimensionalidad con PCA

La siguiente figura muestra los resultados de este análisis.

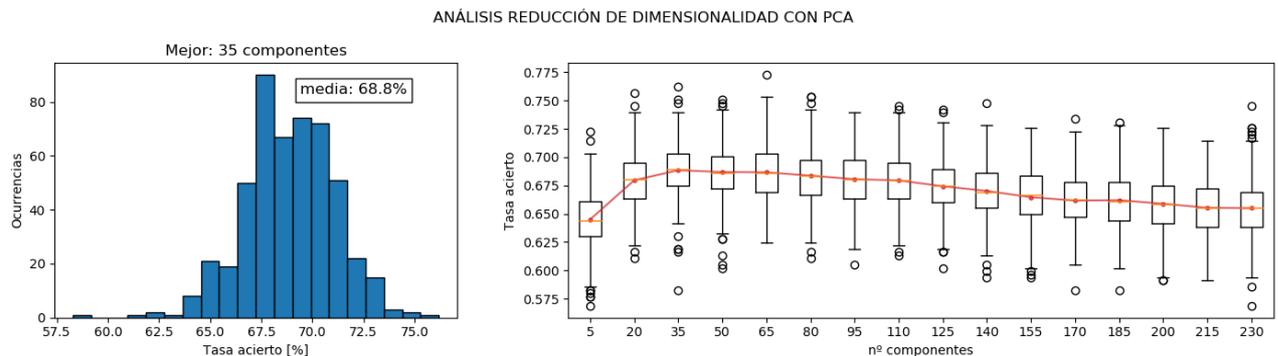


Figura 37: Análisis de la reducción del espacio de características con PCA en la clasificación de la valencia con Random Forest.

Se observa cómo la tasa de acierto crece a medida que nos quedamos con más componentes, hasta que a partir de 35 componentes se mantiene en torno a 68.8% y a partir de 65 comienza a disminuir.

Nótese que para este clasificador, aumentar indefinidamente el número de componentes no lleva a que la tasa de acierto alcance el valor obtenido sin aplicar PCA, por lo que se concluye que aplicar PCA con Random Forest no resulta útil para la clasificación de la valencia.

Análisis de outliers

La siguiente figura muestra cómo varía la tasa de acierto en función del porcentaje de outliers introducido.

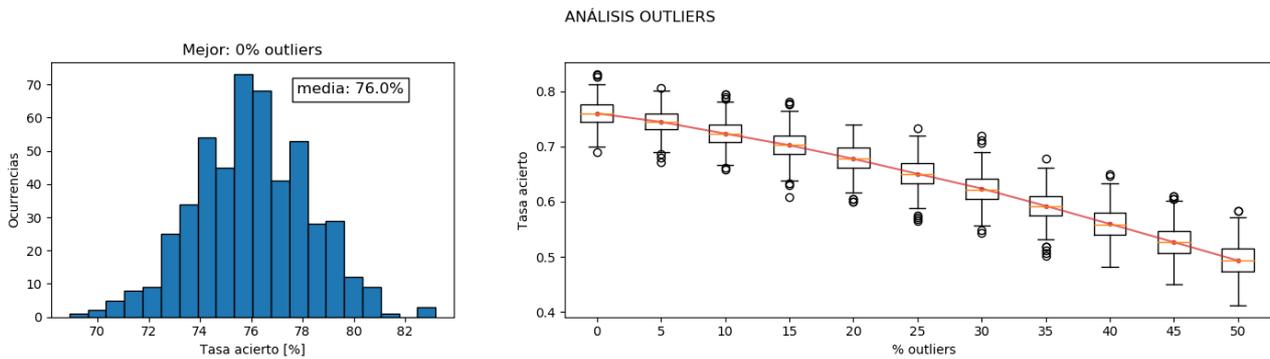


Figura 38: Análisis de outliers en la clasificación de la valencia con Random Forest.

Como era de esperar, la tasa de acierto baja a medida que se introducen más outliers. Cuando la mitad del conjunto de datos de entrenamiento tiene etiquetas erróneas, la tasa de acierto es de un 50%.

Se puede apreciar, comparando con el caso de clasificar con KNN, que ahora la tasa de acierto cae desde el principio de una forma más rápida y siempre uniforme. Para un porcentaje de outliers inferior al 25%, KNN resulta ser más robusto que Random Forest.

6.2.2 Clasificación del grado de excitación

De nuevo, realizamos el análisis de enventanado pero esta vez para los conjuntos 4, 5 y 6. La siguiente figura ilustra los resultados del mismo.

ANÁLISIS DE ENVENTANADO PARA CLASIFICACIÓN DE EXCITACIÓN

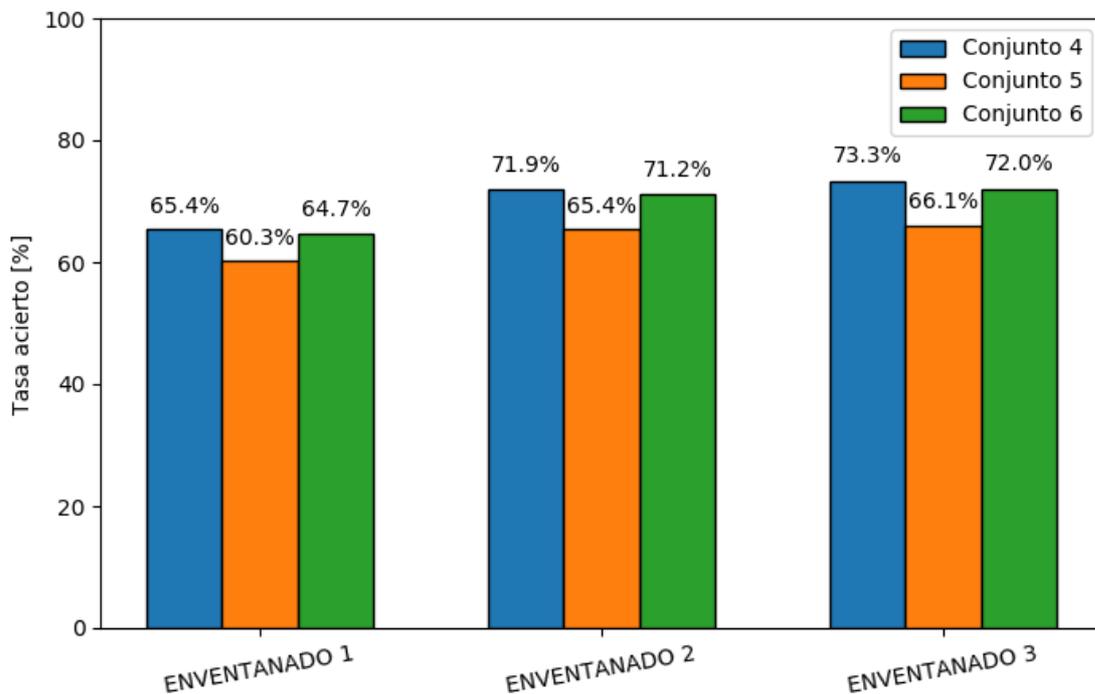


Figura 39: Análisis de las diferentes ventanas para la clasificación del grado de excitación usando Random Forest.

Se observa que la mejor tasa de acierto para la clasificación binaria del grado de excitación mediante Random Forest es de un 73.3% y se obtiene con el conjunto 1 y enventanado 3.

Los conjuntos 4 y 6 arrojan resultados notablemente superiores a los del conjunto 5 cuando se usan los enventanados 2 y 3. Además, se observa de nuevo que el conjunto 6 es un resumen excelente del conjunto 4.

En los análisis que siguen, se utilizará el conjunto 4 con enventanado 3, pues es la combinación que da el

mejor resultado.

Análisis del parámetro profundidad máxima

Del mismo modo que para la clasificación de la valencia, se realiza un análisis variando el valor del parámetro profundidad máxima. La figura de a continuación ilustra los resultados obtenidos a partir de este análisis.

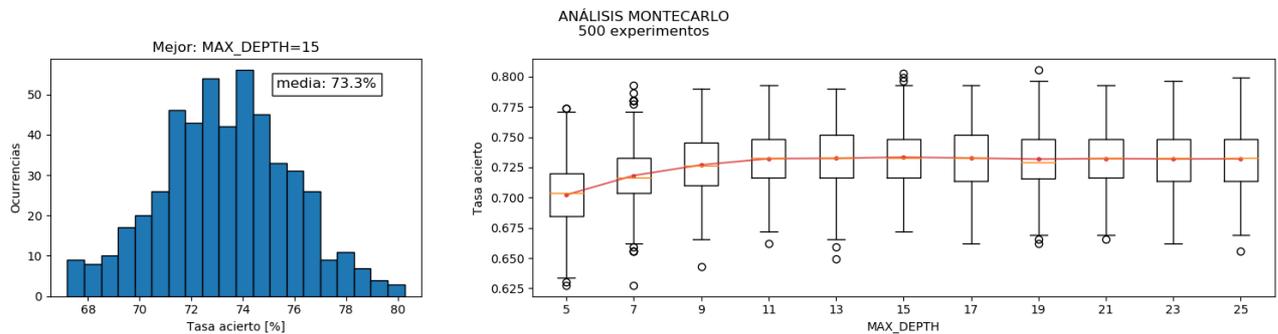


Figura 40: Análisis del parámetro K en la clasificación del grado de excitación con Random Forest.

Se observa que un valor de 15 para el parámetro profundidad máxima nos proporciona una tasa de acierto de un 73.3%. Al igual que en el caso de clasificación de la valencia, el valor de la tasa de acierto crece a medida que aumenta el valor de este parámetro hasta alcanzar un valor en torno al cual permanece constante, en este caso a partir de una profundidad máxima de 11 el valor de la tasa de acierto se mantiene en torno al 73.2%

Análisis de reducción de dimensionalidad con PCA

La siguiente figura muestra los resultados de este análisis.

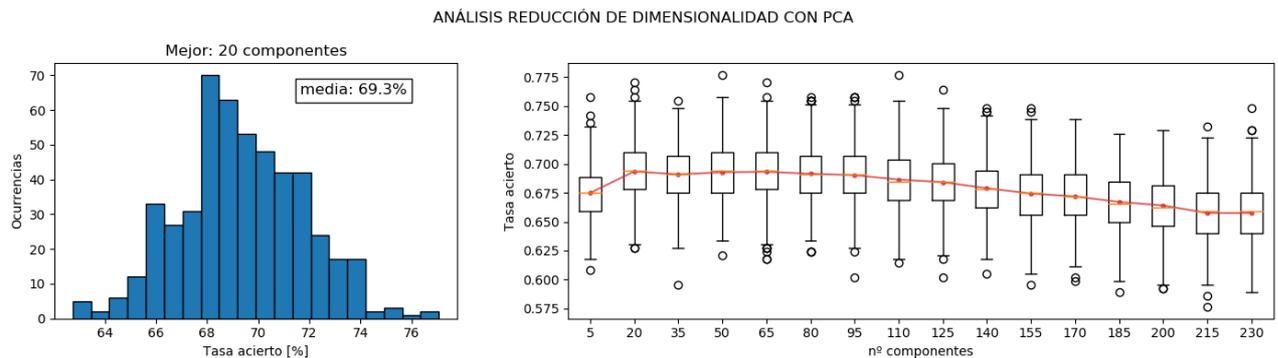


Figura 41: Análisis de la reducción del espacio de características con PCA en la clasificación del grado de excitación con Random Forest.

Los resultados obtenidos son bastante similares a los obtenidos en el caso de clasificación de la valencia, y se concluye que aplicar PCA con Random Forest no resulta útil para la clasificación del grado de excitación.

Análisis de outliers

La siguiente figura muestra cómo varía la tasa de acierto en función del porcentaje de outliers introducido.

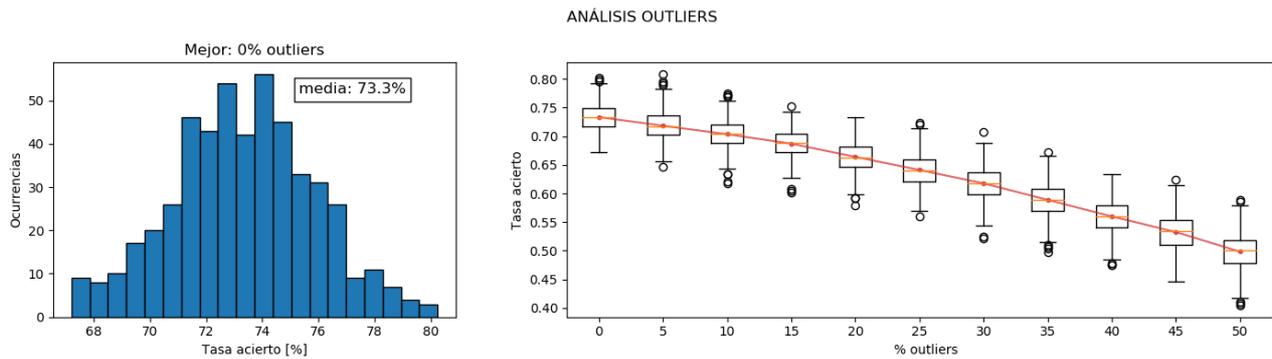


Figura 42: Análisis de outliers en la clasificación del grado de excitación con Random Forest.

Los resultados son similares al caso de clasificación de la valencia, y cuando la mitad del conjunto de datos de entrenamiento tiene etiquetas erróneas, la tasa de acierto es de un 50%. Se observa de nuevo que para un porcentaje de outliers inferior al 25%, KNN resulta ser más robusto que Random Forest.

6.3 Clasificación mediante SVM

Los parámetros del clasificador SVM son los siguientes:

- **Kernel**

Indica el kernel a utilizar. Como se comentó en el capítulo 5, un kernel es una función no lineal que se aplica al conjunto de datos original con el objetivo de obtener un nuevo conjunto de datos en una dimensión mayor y que sea separable.

Los kernels que se van a utilizar son lineal, polinómico, gaussiano (rbf) y sigmoide, y podemos ver cómo separan el conjunto de datos en la siguiente figura.

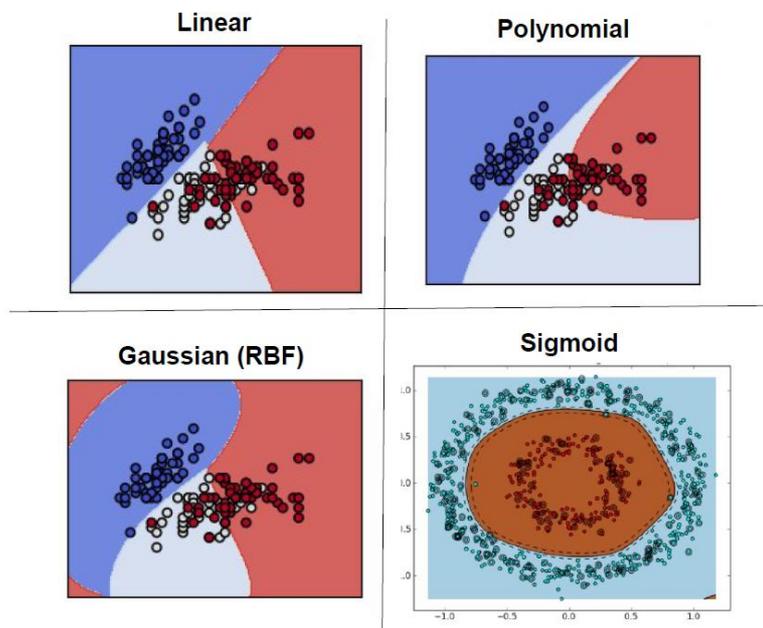


Figura 43: Kernels para SVM utilizados en las simulaciones.

- **C**

C es un parámetro de regularización que controla el balance entre el error obtenido en los datos de entrenamiento y el tamaño del margen máximo. Idealmente, querríamos un tamaño de margen grande y una probabilidad de error en el conjunto de datos pequeña.

Un valor de C pequeño nos proporciona un tamaño de margen mayor pero una probabilidad de error en el conjunto de entrenamiento también mayor; un valor de C grande, lo contrario.

Para implementar este clasificador, se ha utilizado el módulo `sklearn.svm` en Python.

A continuación se comentan los resultados obtenidos utilizando máquinas de soporte vectorial (SVM) para la clasificación binaria de la valencia y del grado de excitación.

6.3.1 Clasificación de la valencia

En primer lugar, realizamos el análisis de eventanado para los conjuntos 1, 2 y 3. A diferencia de KNN y Random Forest, para el caso de SVM se va a analizar además el eventanado para cada una de las mejores combinaciones obtenidas de los valores kernel y C.

Las figuras de a continuación muestran los resultados obtenidos a partir de los análisis de eventanado en los conjuntos 1, 2 y 3 respectivamente.

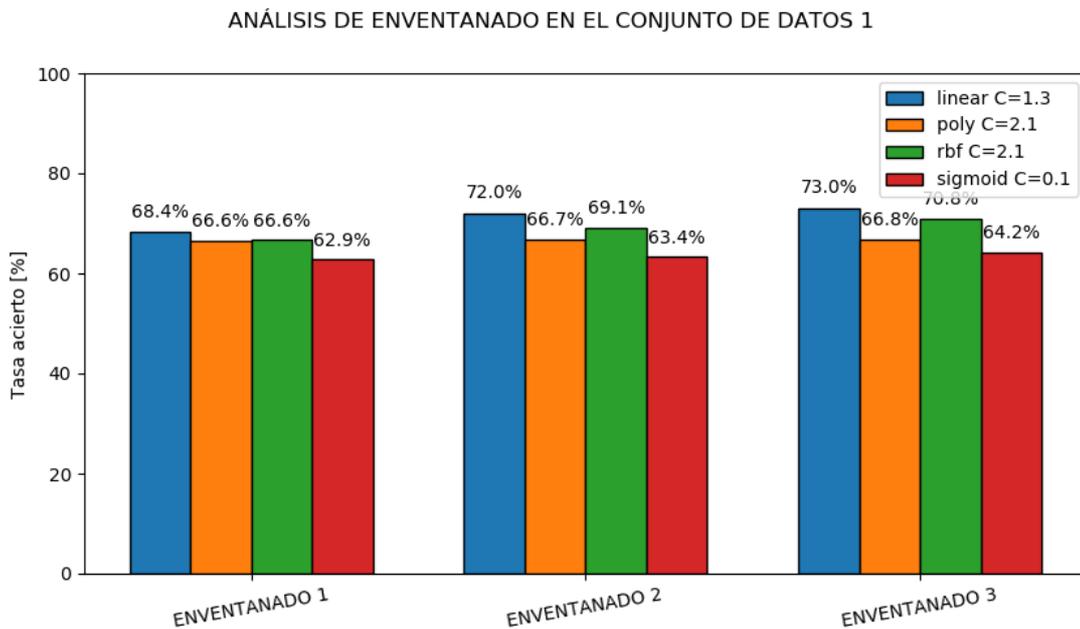


Figura 44: Análisis de las diferentes ventanas para la clasificación de la valencia con el conjunto de características 1 usando SVM.

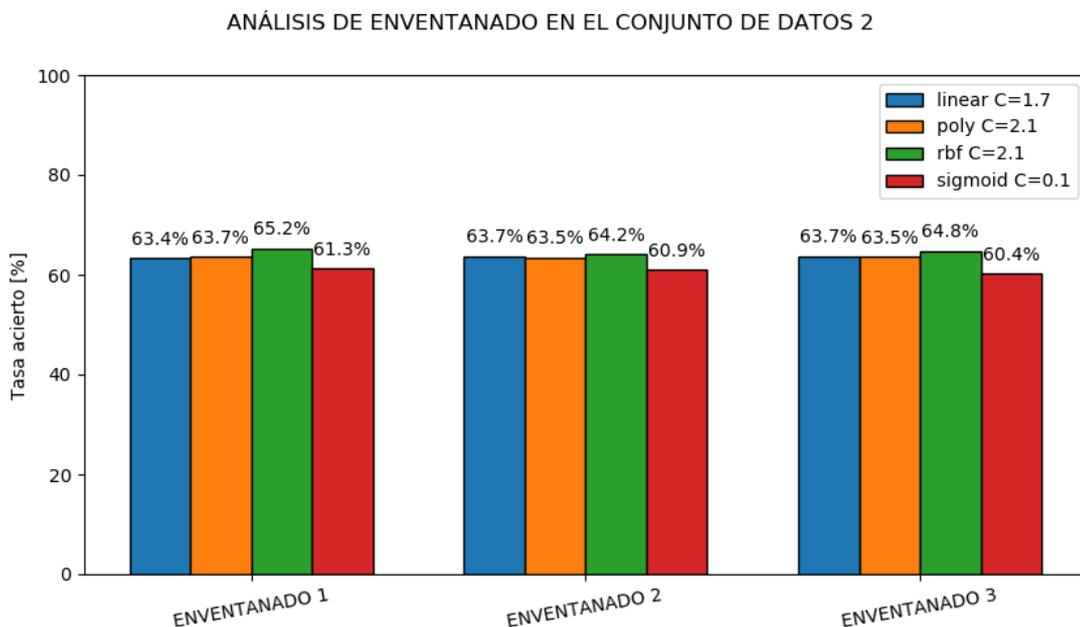


Figura 45: Análisis de las diferentes ventanas para la clasificación de la valencia con el conjunto de características 2 usando SVM.

ANÁLISIS DE ENVENTANADO EN EL CONJUNTO DE DATOS 3

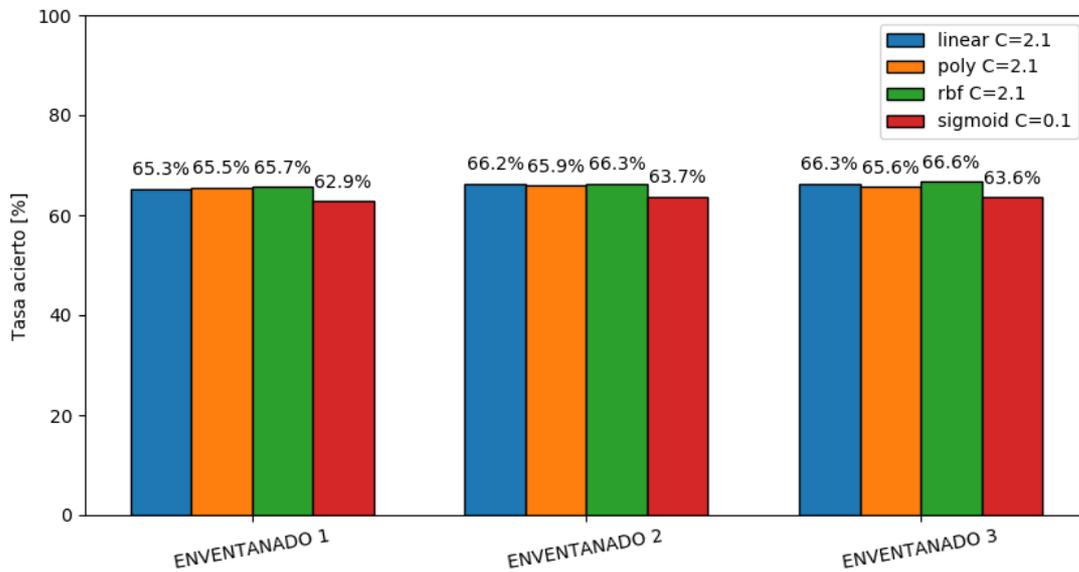


Figura 46: Análisis de las diferentes ventanas para la clasificación de la valencia con el conjunto de características 3 usando SVM.

Se observa que la mejor tasa de acierto para la clasificación binaria de la valencia mediante SVM es de un 73% y se obtiene con el conjunto 1 y enventanado 3 utilizando kernel lineal y valor de C igual a 1.3.

En el caso de SVM los enventanados 2 y 3 suponen mejoras significativas respecto al enventanado 1 sólo en el conjunto de características 1, a diferencia de KNN y Random Forest, en cuyos casos también ocurría esta mejora en el conjunto de características 3.

Además, atendiendo a los resultados obtenidos con SVM, ahora no podemos afirmar que el conjunto 3 sea un resumen muy bueno del conjunto 1.

En los análisis que siguen, se utilizará el conjunto 1 con enventanado 3, pues es la combinación que da el mejor resultado.

Análisis del parámetro C para cada kernel

En este análisis se va a variar el valor del parámetro C para cada uno de los cuatro kernels utilizados (lineal, polinómico, gaussiano (rbf) y sigmoide).

La figura de a continuación muestra los resultados obtenidos para el kernel lineal. En ella se observa cómo la tasa de acierto crece con C hasta que este parámetro toma el valor 1.3, y a partir de ahí la tasa de acierto se mantiene en torno al 73%. El mejor resultado se obtiene para C igual a 1.3.

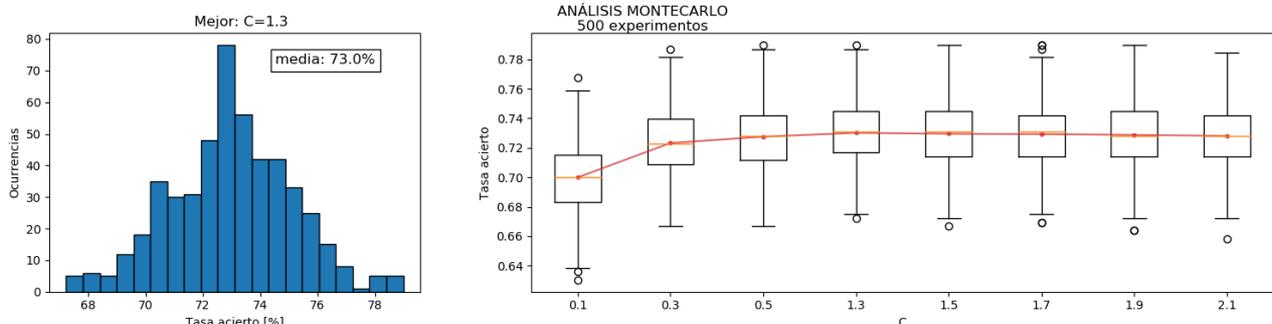


Figura 47: Análisis del parámetro C en la clasificación de la valencia con SVM con kernel lineal.

La siguiente figura ilustra los resultados obtenidos para el kernel polinómico. En ella se observa cómo la tasa de acierto crece con C hasta que este parámetro toma el valor 1.3, y a partir de ahí la tasa de acierto se

mantiene en torno al 68.7%. El mejor resultado se obtiene para C igual a 2.1.

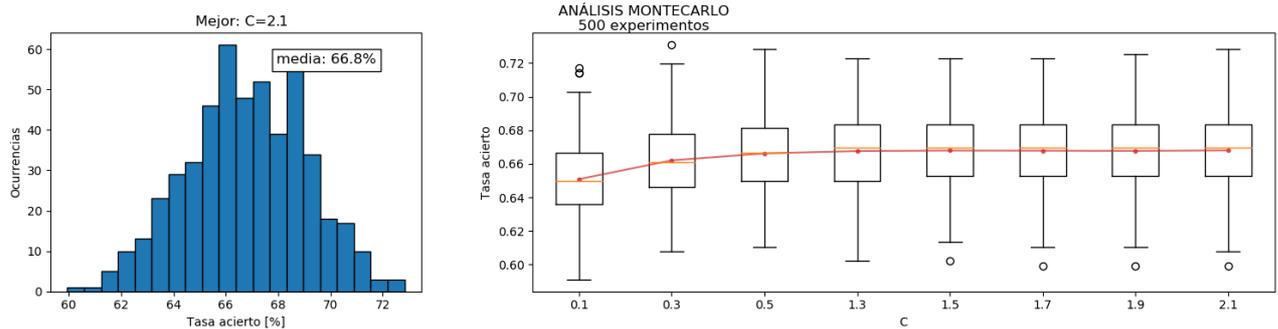


Figura 48: Análisis del parámetro C en la clasificación de la valencia con SVM con kernel polinómico.

La siguiente figura muestra los resultados obtenidos para el kernel gaussiano (rbf). Se observa que la tasa de acierto crece con C hasta que este parámetro toma el valor 2.1, obteniendo una tasa de acierto del 70.8%.

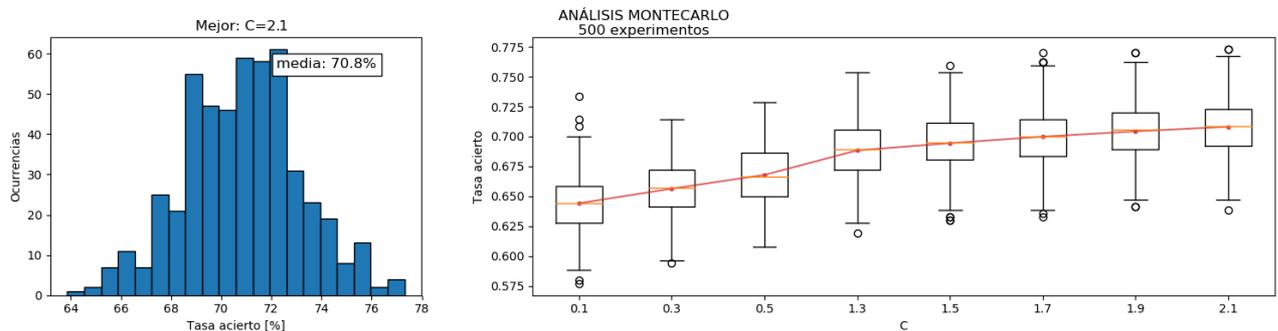


Figura 49: Análisis del parámetro C en la clasificación de la valencia con SVM con kernel gaussiano.

La figura de a continuación ilustra los resultados obtenidos para el kernel sigmoide. En ella se observa cómo la tasa de acierto, a diferencia de los demás kernels, disminuye con C. El mejor resultado se obtiene para C igual a 0.1.

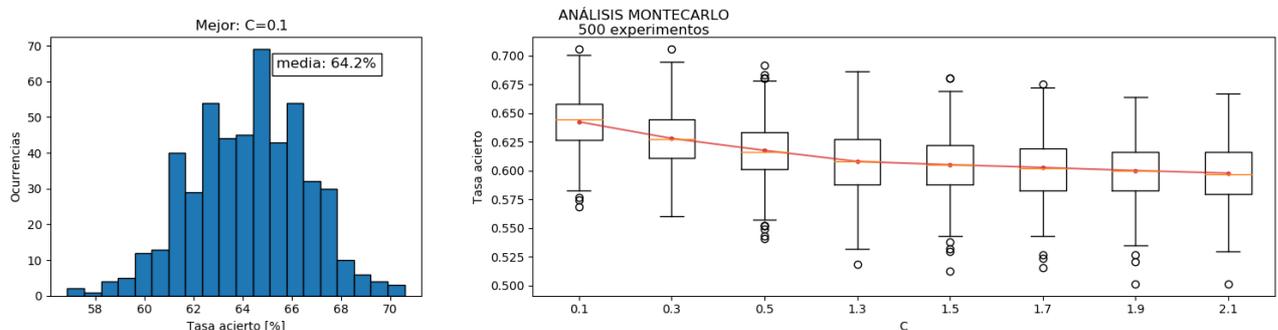


Figura 50: Análisis del parámetro C en la clasificación de la valencia con SVM con kernel sigmoide.

A modo de síntesis, la siguiente figura muestra los resultados obtenidos para cada kernel y su mejor valor de C.

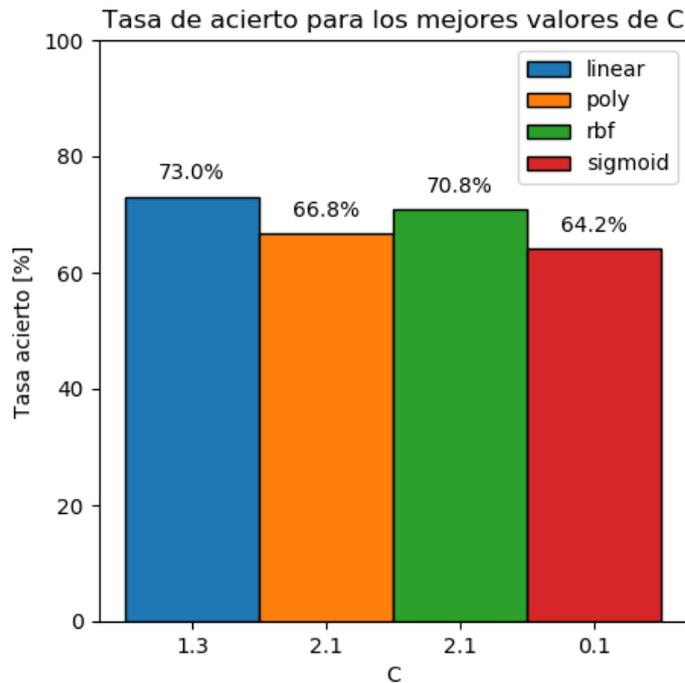


Figura 51: Resultados obtenidos para la clasificación de valencia con los diferentes kernels para SVM.

Análisis de reducción de dimensionalidad con PCA

La siguiente figura muestra los resultados de este análisis, utilizando kernel lineal.

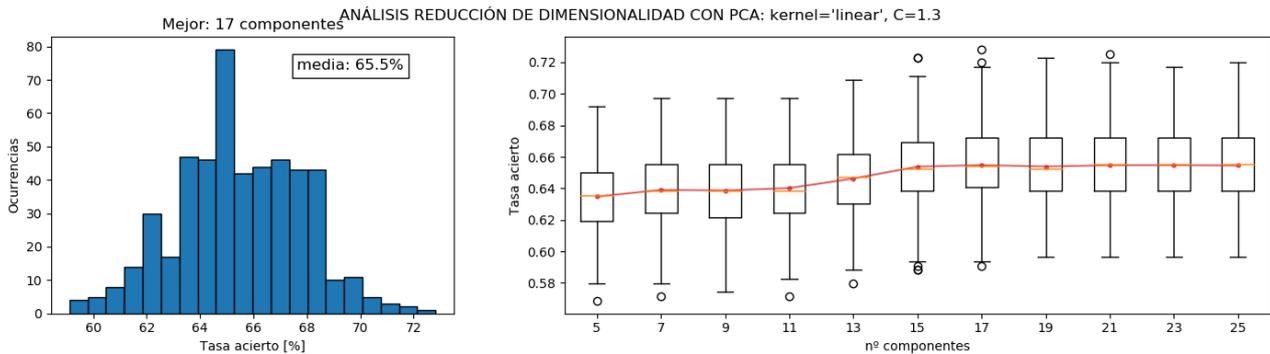


Figura 52: Análisis de la reducción del espacio de características con PCA en la clasificación de la valencia con SVM.

Se observa cómo la tasa de acierto crece a medida que nos quedamos con más componentes, hasta que satura a partir de cierto valor. Sin embargo, al igual que ocurría en Random Forest, aumentar indefinidamente el número de componentes no lleva a que la tasa de acierto alcance el valor obtenido sin aplicar PCA.

Se concluye que aplicar PCA con SVM no resulta útil para la clasificación de la valencia. Al contrario, ofrece peores resultados.

Análisis de outliers

La siguiente figura muestra cómo varía la tasa de acierto en función del porcentaje de outliers introducido.

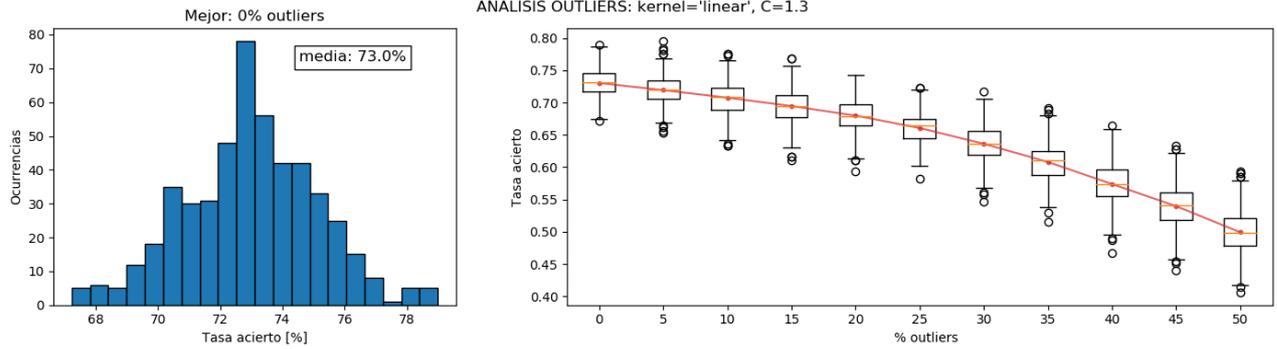


Figura 53: Análisis de outliers en la clasificación de la valencia con SVM.

Como era de esperar, la tasa de acierto baja a medida que se introducen más outliers. Cuando la mitad del conjunto de datos de entrenamiento tiene etiquetas erróneas, la tasa de acierto es de un 50%.

Comparando con Random Forest y KNN, observamos un comportamiento bastante similar al de KNN ante la presencia de outliers y, por lo tanto, que para un porcentaje de outliers inferior al 25% tanto SVM como KNN resultan ser más robustos que Random Forest.

6.3.2 Clasificación del grado de excitación

Las figuras de a continuación muestran los resultados obtenidos a partir de los análisis de eventanado en los conjuntos 4, 5 y 6 respectivamente.

ANÁLISIS DE EVENTANADO EN EL CONJUNTO DE DATOS 4

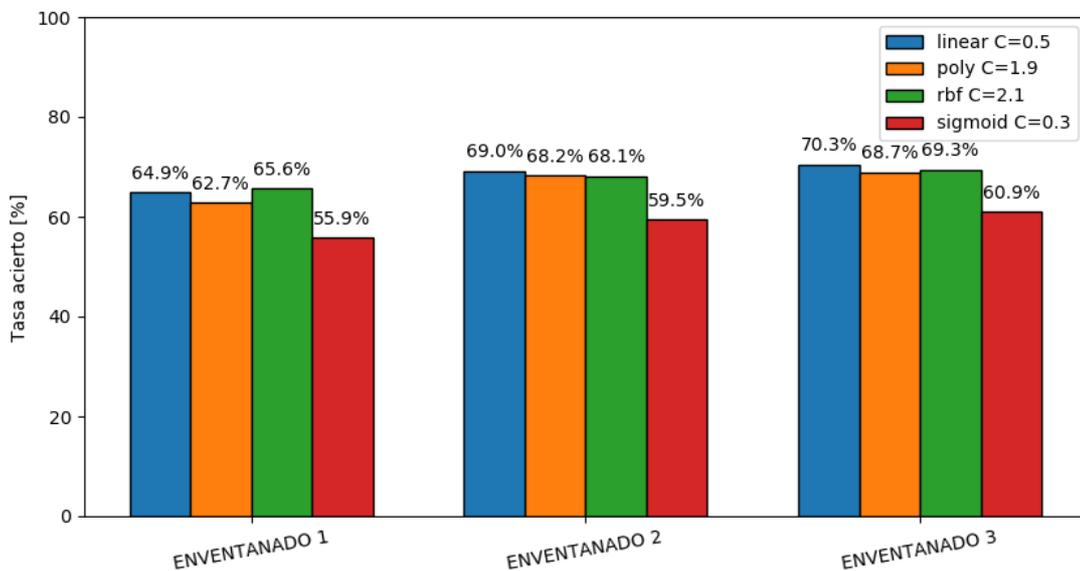


Figura 54: Análisis de las diferentes ventanas para la clasificación del grado de excitación con el conjunto de características 4 usando SVM.

ANÁLISIS DE ENVENTANADO EN EL CONJUNTO DE DATOS 5

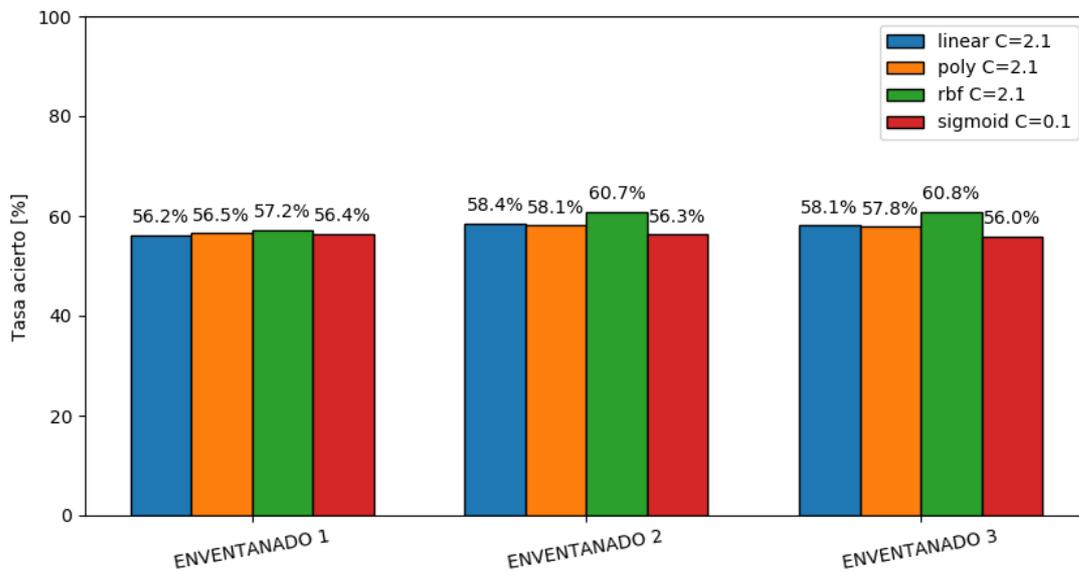


Figura 55: Análisis de las diferentes ventanas para la clasificación del grado de excitación con el conjunto de características 5 usando SVM.

ANÁLISIS DE ENVENTANADO EN EL CONJUNTO DE DATOS 6

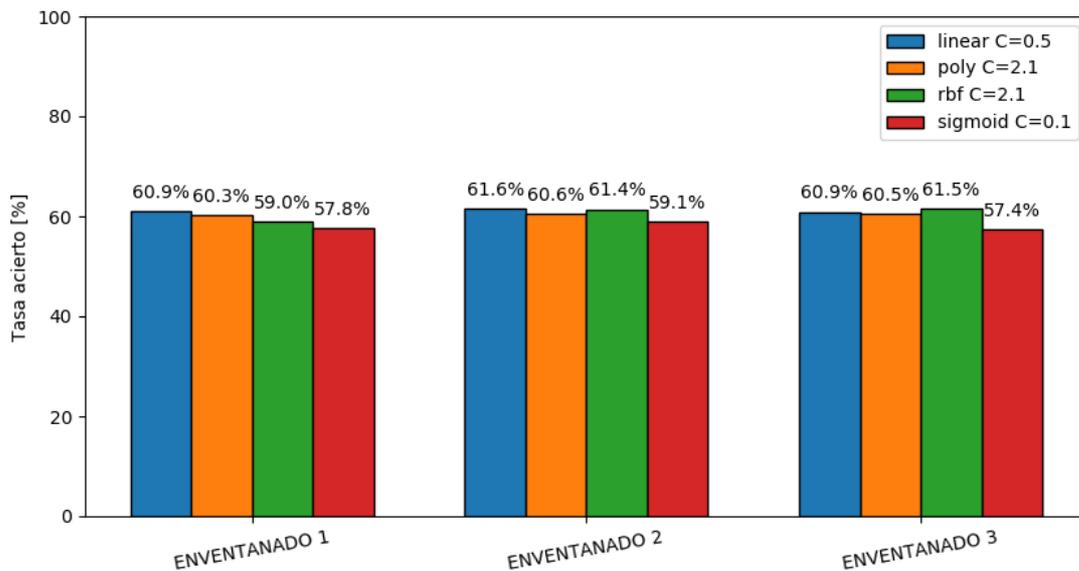


Figura 56: Análisis de las diferentes ventanas para la clasificación del grado de excitación con el conjunto de características 6 usando SVM.

Se observa que la mejor tasa de acierto para la clasificación binaria del grado de excitación mediante SVM es de un 70.3% y se obtiene con el conjunto 1 y enventanado 3 utilizando kernel lineal y valor de C igual a 0.5.

Respecto a los enventanados y los diferentes conjuntos de características, se pueden extraer exactamente las mismas conclusiones que en el caso de clasificación de la valencia.

En los análisis que siguen, se utilizará el conjunto 4 con enventanado 3, pues es la combinación que da el mejor resultado.

Análisis del parámetro C para cada kernel

En este análisis se va a variar el valor del parámetro C para cada uno de los cuatro kernels utilizados (lineal,

polinómico, gaussiano (rbf) y sigmoide).

La figura de a continuación muestra los resultados obtenidos para el kernel lineal. En ella se observa cómo la tasa de acierto crece con C hasta que este parámetro toma el valor 0.3, y a partir de ahí la tasa de acierto se mantiene en torno al 70% hasta que a partir de C igual a 1.9 empieza a disminuir. El mejor resultado se obtiene para C igual a 0.3.

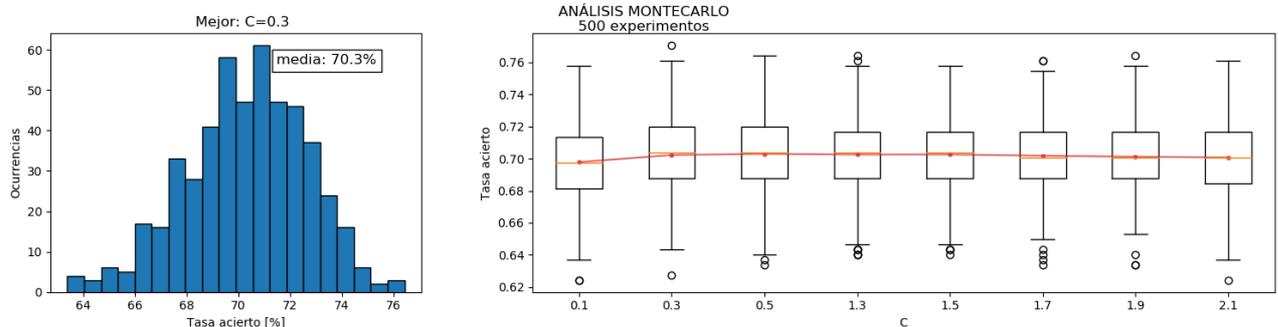


Figura 57: Análisis del parámetro C en la clasificación del grado de excitación con SVM con kernel lineal.

La siguiente figura ilustra los resultados obtenidos para el kernel polinómico. En ella se observa cómo la tasa de acierto crece con C hasta que este parámetro toma el valor 1.3, y a partir de ahí la tasa de acierto se mantiene en torno al 68.7%. El mejor resultado se obtiene para C igual a 1.7.

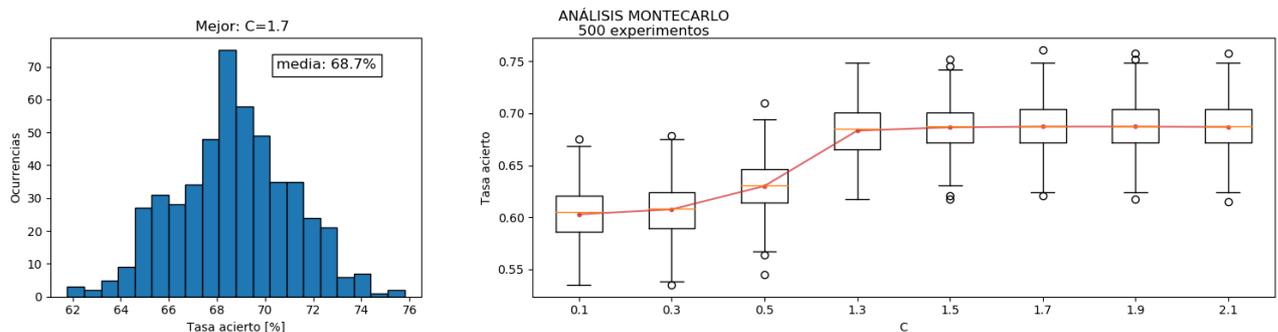


Figura 58: Análisis del parámetro C en la clasificación del grado de excitación con SVM con kernel polinómico.

La siguiente figura muestra los resultados obtenidos para el kernel gaussiano (rbf). Se observa que la tasa de acierto crece con C hasta que este parámetro toma el valor 0.3, a partir de donde la tasa de acierto se mantiene en torno al 69%.

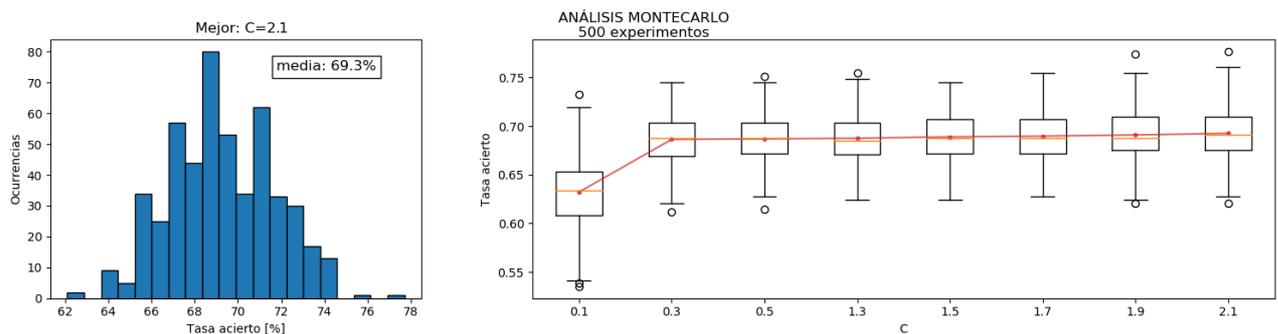


Figura 59: Análisis del parámetro C en la clasificación del grado de excitación con SVM con kernel gaussiano.

La figura de a continuación ilustra los resultados obtenidos para el kernel sigmoide. En ella se observa cómo la tasa de acierto aumenta con C hasta que este parámetro toma el valor 0.3, y a partir de ahí comienza a disminuir.

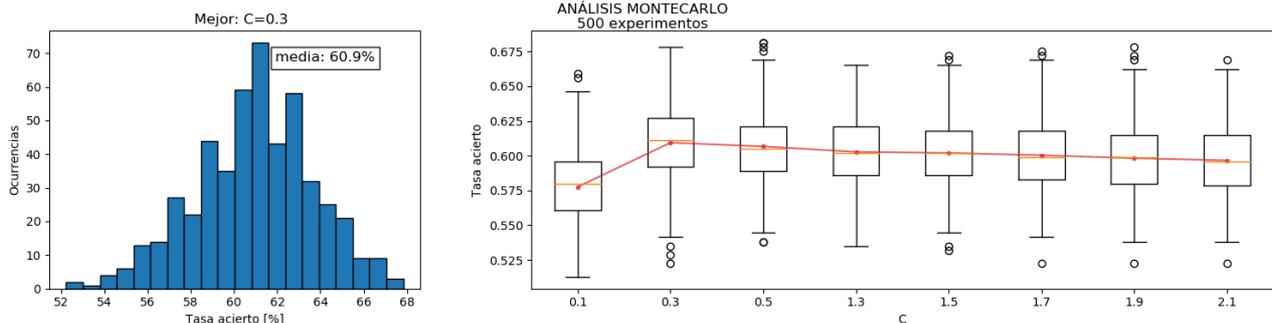


Figura 60: Análisis del parámetro C en la clasificación del grado de excitación con SVM con kernel sigmoide.

A modo de síntesis, la siguiente figura muestra los resultados obtenidos para cada kernel y su mejor valor de C.

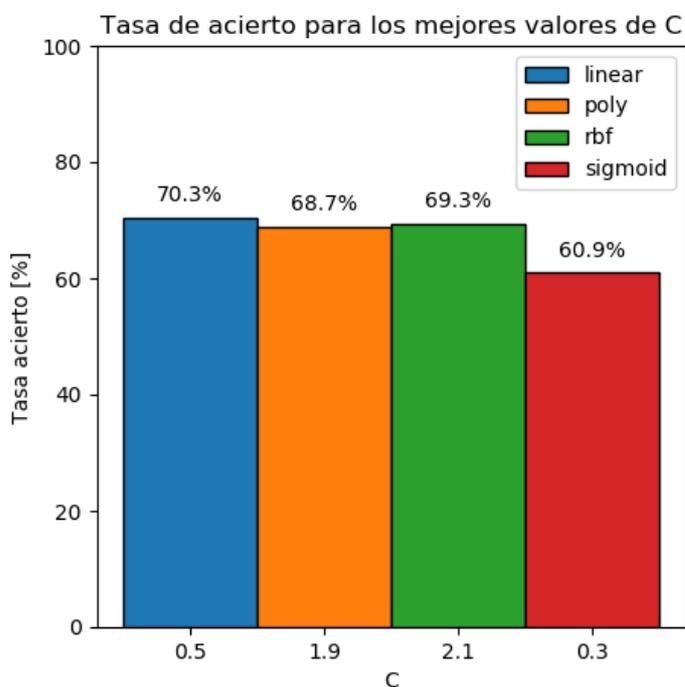


Figura 61: Resultados obtenidos para la clasificación del grado de excitación con los diferentes kernels para SVM.

Análisis de reducción de dimensionalidad con PCA

La siguiente figura muestra los resultados de este análisis.

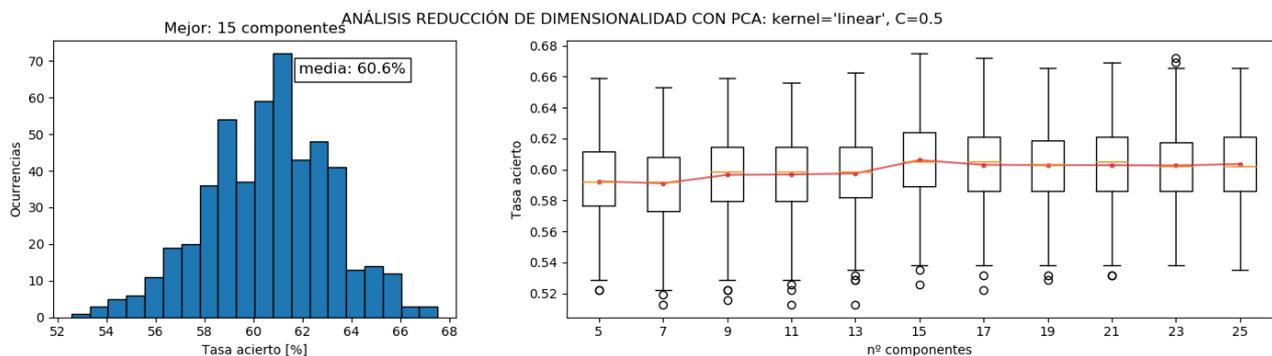


Figura 62: Análisis de la reducción del espacio de características con PCA en la clasificación del grado de excitación con SVM.

Al igual que ocurría en el caso de clasificación de la valencia, aumentar indefinidamente el número de componentes no lleva a que la tasa de acierto alcance el valor obtenido sin aplicar PCA.

Se concluye que aplicar PCA con SVM no resulta útil para la clasificación del grado de excitación. Al contrario, ofrece peores resultados.

Análisis de outliers

La siguiente figura muestra cómo varía la tasa de acierto en función del porcentaje de outliers introducido.

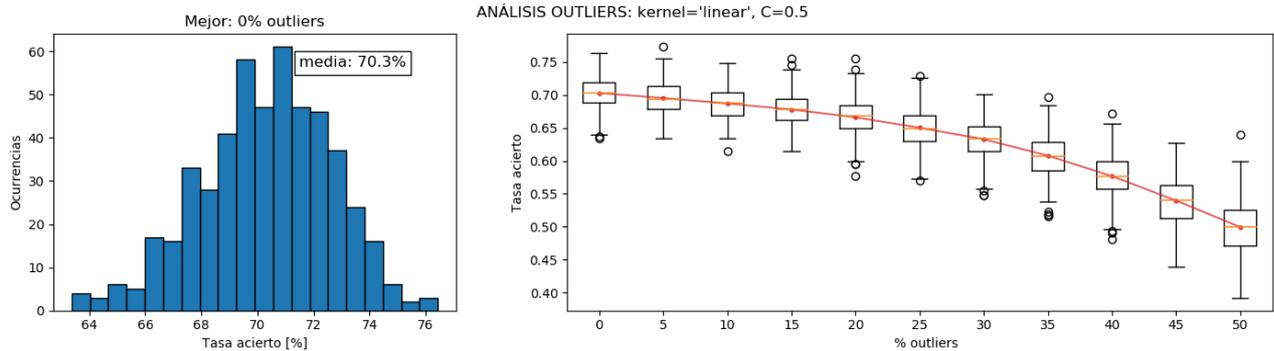


Figura 63: Análisis de outliers en la clasificación del grado de excitación con SVM.

Los resultados son similares al caso de clasificación de la valencia, y cuando la mitad del conjunto de datos de entrenamiento tiene etiquetas erróneas, la tasa de acierto es de un 50%. Se observa de nuevo comportamiento bastante similar al de KNN, y que para un porcentaje de outliers inferior al 25%, tanto SVM como KNN resultan ser más robustos que Random Forest.

6.4 Otros clasificadores

Además de los clasificadores anteriores, se han probado pero no de forma exhaustiva, LDA y redes neuronales para la clasificación binaria de la valencia y del grado de excitación.

LDA

LDA fue el primer clasificador que se probó en el presente trabajo, pero se desechó debido a los pobres resultados que ofrecía (tasas de acierto inferiores al 60% para ambos problemas de clasificación).

Además, LDA es el método de clasificación utilizado por los experimentadores que elaboraron la base de datos utilizada, y los resultados que obtenían eran igual de malos: 57.6% para la clasificación de valencia y 62% para la clasificación del grado de excitación.

LDA se ha implementado en matlab a partir del siguiente pseudocódigo.

```

LEARNING THE LDA (BINARY) CLASSIFIER
% Using the training pairs (c(τ), fτ)
μk = 1/Nck ∑τ:c(τ)=ck fτ for k = 1, 2.
Σ̂f|ck = 1/Nck ∑τ:c(τ)=ck (fτ - μk)(fτ - μk)T, k = 1, 2.
p(ck) = Nck / (Nc1 + Nc2), k = 1, 2.
Σ̂f = p(c1)Σ̂f|c1 + p(c2)Σ̂f|c2
α = Σ̂f-1(μ1 - μ2)
β = 1/2(μ1 + μ2) - (log p(c1) - log p(c2)) / ((μ1 - μ2)T Σ̂f-1 (μ1 - μ2))

```

```

CLASSIFICATION OF TEST DATA
% Implementation using LDA with equal cov.
CYτ = WT CXτ(0) W, τ ∈ Set of test trials.
Evaluate fτ using the same formula as in (F1)
ĉ(τ) = ck where k = 1.5 + 0.5 sign(αT(fτ - β)).

```

Figura 64: Pseudocódigo para la implementación de LDA. Fuente: [46].

Redes neuronales artificiales (ANN)

Se ha utilizado el módulo **keras** en Python para implementar una red neuronal de 3 capas con las siguientes características.

Tabla 13: Capas de la red neuronal implementada.

CAPA	Nº de neuronas	Inicialiación del kernel	Función de activación
1	12	Uniforme	ReLU
2	20	Uniforme	ReLU
3	1	Uniforme	Sigmoide

Con función de pérdidas raíz del error cuadrático medio y optimizador RMSProp. Los conjuntos de características utilizados han sido el 1 y el 4, ambos con enventanado 3, pues son los que mejor resultados han dado en los anteriores clasificadores

Aunque no se han hecho pruebas exhaustivas, las siguientes figuras muestran resultados preliminares que se pueden obtener utilizando redes neuronales.

La figura de a continuación ilustra la tasa de acierto en los conjuntos de entrenamiento y validación en función del número de épocas (“epochs”) para la clasificación de la valencia.

TASA DE ACIERTO EN CONJUNTO DE CARACTERÍSTICAS 1

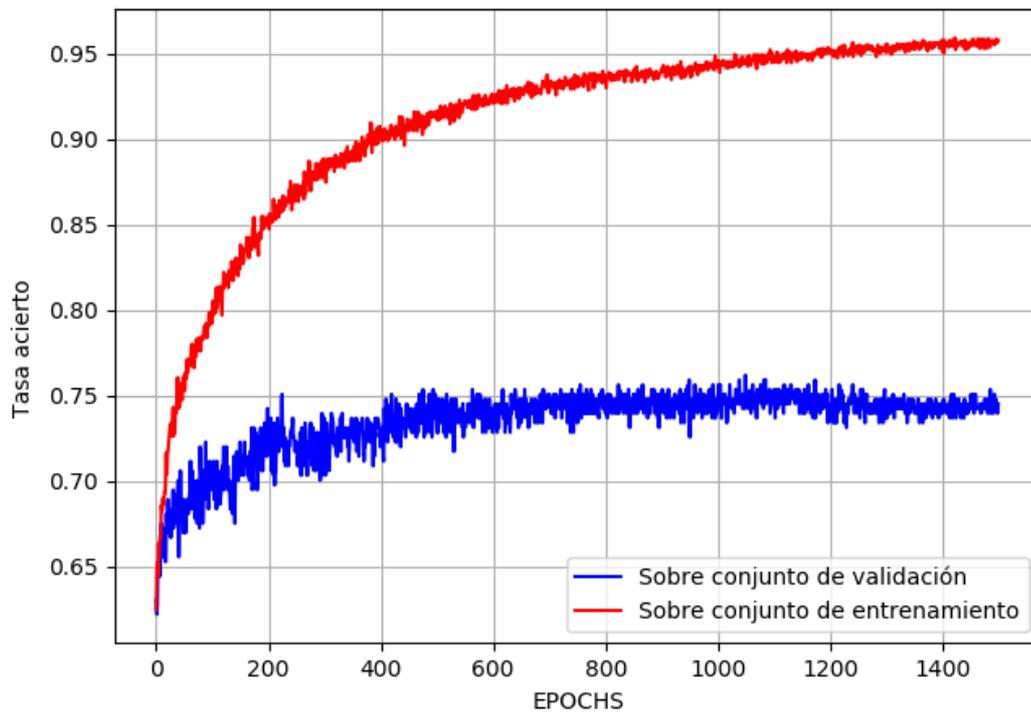


Figura 65: Tasa de acierto en función del número de épocas para la clasificación de la valencia con ANN.

La tasa de acierto que se logra alcanzar es de alrededor de un 75%, alcanzando un rendimiento cercano al de Random Forest (76%).

La siguiente figura muestra la tasa de acierto en los conjuntos de entrenamiento y validación en función del número de épocas para la clasificación de la valencia.

TASA DE ACIERTO EN CONJUNTO DE CARACTERÍSTICAS 4

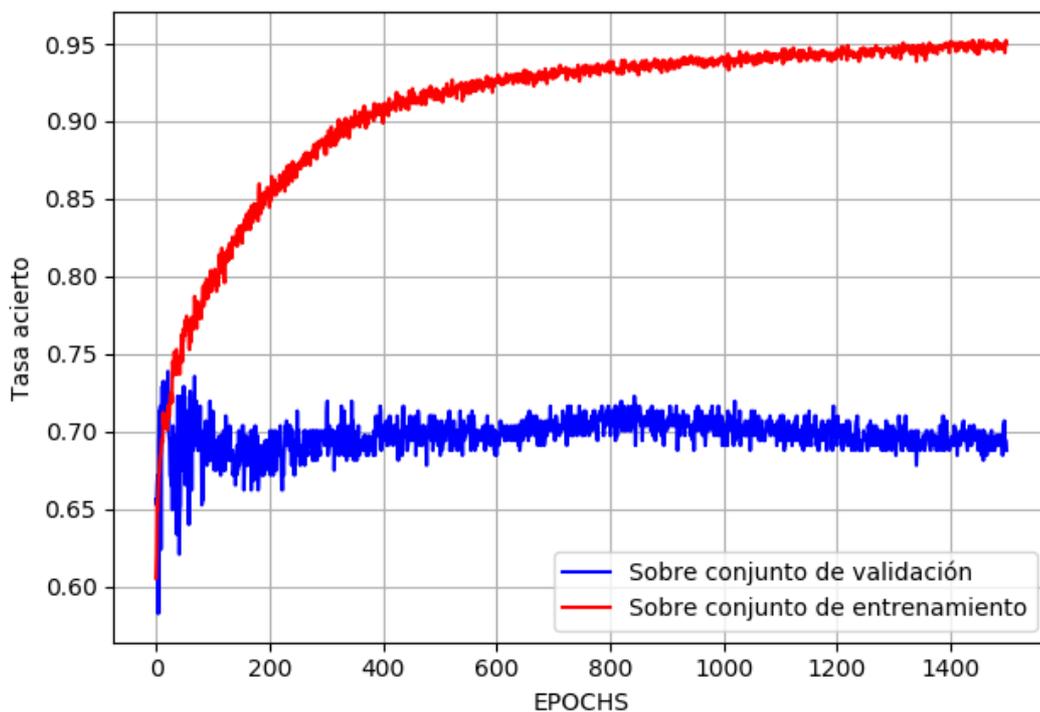


Figura 66: Tasa de acierto en función del número de épocas para la clasificación del grado de excitación con ANN.

Se logra alcanzar una tasa de acierto de hasta el 74% con pocas épocas, superando los resultados obtenidos con Random Forest (73.3%). Sin embargo a medida que el número de épocas crece, la tasa de acierto en el conjunto de validación baja considerablemente debido a que se produce sobreajuste sobre el conjunto de datos de entrenamiento.

6.5 Conclusiones

En este capítulo se han expuesto y comentado los resultados de las simulaciones llevadas a cabo en el trabajo, que han consistido principalmente en el análisis de los clasificadores KNN, Random Forest y SVM para las clasificaciones binarias de la valencia y del grado de excitación.

Los mejores resultados se han obtenido con el clasificador Random Forest (tasas de acierto del 76% para la clasificación de la valencia y 73.3% para la clasificación del grado de excitación).

Aunque no se han hecho simulaciones exhaustivas con redes neuronales, lo poco que se ha probado ha ofrecido resultados que están a la par con los obtenidos con Random Forest, y es por esto que se propone como línea futura del trabajo investigar más a fondo el tema de la clasificación de la valencia y el grado de excitación usando este tipo de clasificador.

Las siguiente muestra las tasas de acierto obtenidas con cada clasificador para cada uno de los problemas de clasificación.

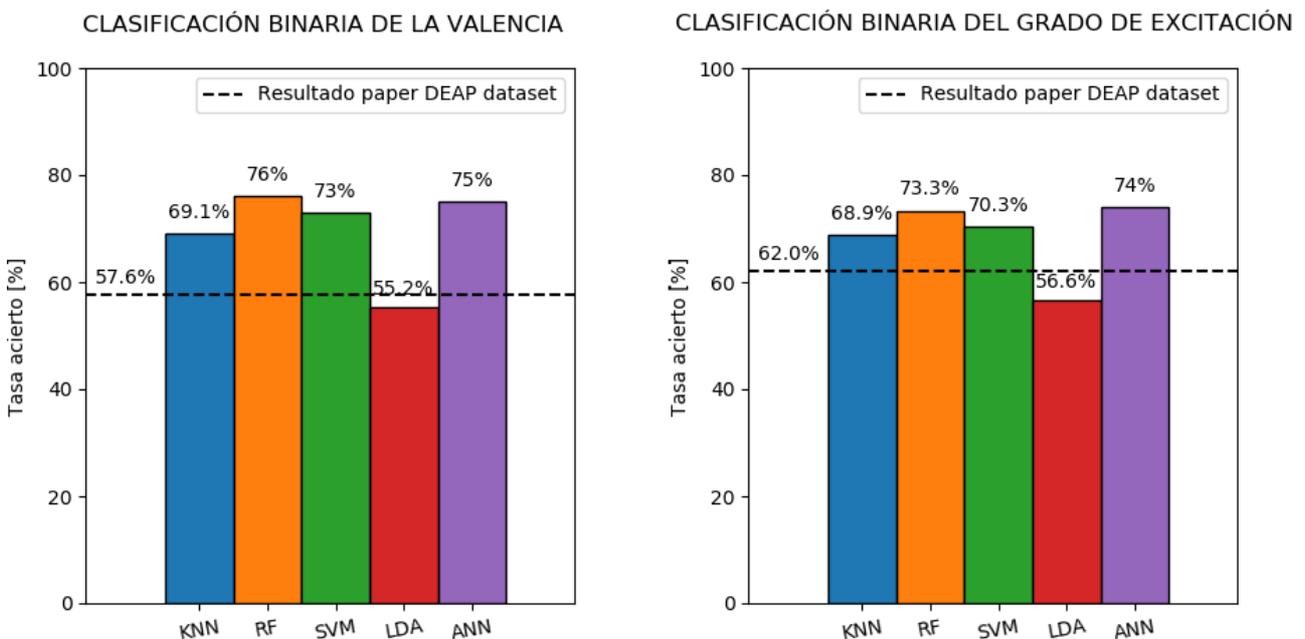


Figura 67: Tasas de acierto obtenidas con los diferentes clasificadores.

Además, a partir de los resultados de las simulaciones realizadas se pueden extraer las siguientes conclusiones:

- El conjunto de características 3 es un subconjunto del conjunto 1, pero de una dimensión muchísimo menor (24 características frente a 230) y las tasas de acierto obtenidas para la clasificación de la valencia usando ambos conjuntos difieren apenas en un 1% cuando se clasifica con KNN y Random Forest, por lo que podemos afirmar que el conjunto de características 3 contiene aquellas más representativas y determinantes.
- Lo mismo que sucede con los conjuntos 3 y 1 sucede con los conjuntos 6 y 4 a la hora de clasificar el grado de excitación (en este caso, 17 características frente a 230) y concluimos también que el conjunto 4 contiene las características más útiles para clasificar el grado de excitación.

- KNN y SVM son ligeramente más robustos que Random Forest ante la presencia de outliers.
- Reducir la dimensionalidad de los datos utilizados utilizando PCA no resulta útil.

7 CONCLUSIONES Y LÍNEAS FUTURAS

En este trabajo se han abordado dos problemas de clasificación binaria: la clasificación de la valencia y la clasificación del grado de excitación, ambas a través de señales de electroencefalograma.

Mediante la clasificación de la valencia y el grado de excitación lo que se pretende en realidad es clasificar emociones de acuerdo al modelo de Russell, que como se ha explicado en la sección 2 del capítulo 2, sugiere que las emociones están distribuidas en un espacio circular de dos dimensiones, excitación y valencia, tal y como se ilustra en la siguiente figura.

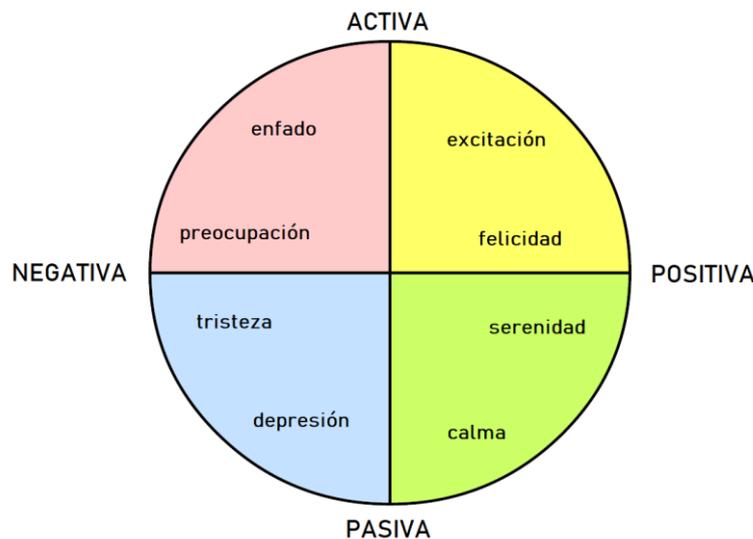


Figura 68: Modelo de Russell.

Para llevar a cabo el trabajo, en primer lugar se hizo una revisión profunda del estado del arte en el reconocimiento de emociones, lo cual se expuso en el capítulo 3.

A partir de ahí, se estudió qué características extraer de las señales EEG para clasificar por un lado la valencia y por otro el grado de excitación, y posteriormente métodos y algoritmos para llevar a cabo las clasificaciones binarias de la valencia y del grado de excitación. De esto se ha hablado en el capítulo 5.

Por último, se realizaron diferentes simulaciones explicadas en el capítulo 6, principalmente utilizando los clasificadores KNN, Random Forest y SVM. Las tasas de acierto que se han logrado conseguir son de un 76% para la clasificación de la valencia y un 73.3% para la clasificación del grado de excitación, ambas notablemente superiores a las obtenidas por los experimentadores que elaboraron la base de datos utilizada (57.6% y 62%, respectivamente).

Los objetivos del presente trabajo eran explorar la clasificación de emociones mediante el electroencefalograma y conseguir clasificar por un lado valencia positiva o negativa y por otro grado de excitación alto o bajo. Dichos objetivos han logrado alcanzarse, y a partir de aquí se sugieren las siguientes líneas futuras:

1) Investigación con redes neuronales.

Este trabajo se ha centrado en los clasificadores KNN, Random Forest y SVM, con los cuales se han hecho diferentes pruebas variando sus parámetros. Adicionalmente, se probaron LDA y redes

neuronales, pero sin entrar en gran profundidad, y los resultados obtenidos haciendo una prueba relativamente rápida con una red neuronal de tres capas alcanzan a los obtenidos con Random Forest, que daba los mejores resultados.

Es por este motivo que se sugiere como línea futura la investigación de redes neuronales para la clasificación de la valencia y el grado de excitación, ya que es posible que ajustando diferentes parámetros se alcancen mejores resultados que los obtenidos en este trabajo.

2) Llevar a cabo la clasificación de forma individual por cada paciente en lugar de clasificar con los datos de todos los pacientes.

En el presente trabajo el entrenamiento de los clasificadores y la posterior validación se ha hecho utilizando el conjunto de muestras de todos los pacientes. Sin embargo, debido a la individualidad de cada persona, la activación cerebral a la hora de sentir una emoción puede variar en cada paciente. El principal hándicap por el que esto no se hizo así en este trabajo es la poca cantidad de muestras que, a priori, se tienen por cada paciente (40).

No obstante, aplicando un eventanado de más de una ventana, el número de muestras que tendremos por paciente se habrá multiplicado por el número de ventanas (por ejemplo, con el eventanado 3, teníamos $40 * 3 = 120$ muestras por paciente).

Además, fue el eventanado 3 el que mejores resultados dio, así que sería interesante utilizar dicho eventanado, o incluso un eventanado con un número de ventanas mayor, para clasificar cada paciente por separado.

3) Clasificar en base a tres clases (valencia positiva, negativa o neutra y grado de excitación alto, medio o bajo).

Actualmente, el entrenamiento y validación de los clasificadores implementados no se realiza con el conjunto de datos completo. En su lugar, se toman aquellas muestras con puntuaciones más en los extremos (Ver Script *características.m*, parámetros `PUNTUACION_CLASE1` y `PUNTUACION_CLASE2`), quedando excluidas las muestras pertenecientes al intervalo de puntuaciones más neutras.

Se podría considerar añadir una nueva clase a la valencia y al grado de excitación y utilizar las muestras que actualmente no se usan, aunque de esta forma los resultados que se obtengan serán peores.

4) Clasificar en conjunto valencia y excitación para determinar una emoción concreta

Hasta ahora se ha separado el problema de la clasificación de emociones en dos problemas de clasificación: clasificar la valencia por un lado y el grado de excitación por otra. Queda pendiente por lo tanto la unificación de ambos para realmente conseguir clasificar emociones.

Esto podría hacerse de dos formas:

- a) Utilizar dos clasificadores en paralelo y a partir del resultado de ambos determinar una emoción. Esta forma, de acuerdo al trabajo ya realizado, sería la más rápida e incluso inmediata de implementar.
- b) Implementar un nuevo clasificador que clasifique en base a un número de clases igual a las emociones que queramos clasificar.

REFERENCIAS

- [1] Emotiv Homepage. [En línea]. Disponible: <http://www.emotiv.com/>
- [2] g.tec. «g.BCIsys», [En línea]. Disponible: <https://www.bluetooth.com/specifications/bluetooth-core-specification>
- [3] C. Wang, C. Guan, H. Zhang. «P300 brain-computer interface design for communication and control applications»
- [4] T. Felzer, R. Fischer, T. Groensfelder, R. Nordmann. «Alternative control system for operating a pc using intentional muscle contractions only»
- [5] New Atlas. «The first commercially available Brain Computer Interface ». [En línea] .Disponible: <https://newatlas.com/the-first-commercially-available-brain-computer-interface/6971/>
- [6] Arstechnica. «Could EEGs have prevented Clippy? Microsoft taps brain scan for UI work». [En línea]. Disponible: <https://arstechnica.com/information-technology/2007/10/could-eegs-have-prevented-clippy-microsoft-taps-brain-scan-for-ui-work/>
- [7] A. Choppin. «EEG-based human interface for disabled individuals: Emotion expression with neural networks»
- [8] Facultad de Psicología de la Universidad de Viena. «EEG Labs». [En línea]. Disponible: <https://psychologie.univie.ac.at/forschung/labors/eeg-labs/>
- [9] Wikipedia. «Electroencephalography». [En línea]. Disponible: <https://en.wikipedia.org/wiki/Electroencephalography>
- [10] A. Heraz, R. Razaki, C. Frasson. «Using machine learning to predict learner emotional state from brainwaves»
- [11] Bradley C. Lega, Joshua Jacobs, Michael Kahana. «Human hippocampal theta oscillations and the formation of episodic memories»
- [12] G. Buzsaki. «Rhythms of the brain»
- [13] Robert Horlings, Dragos Datcu, Léon J. M. Rothkrantz. «Emotion recognition using brain activity»
- [14] J. Villegas. « Definition of emotion»
- [15] P.R. Kleinginna Jr. and A.M. Kleinginna. «A categorized list of emotion denitions, with suggestions for a consensual denition»

- [16] Wikipedia. «Emotion Classification». [En línea]. Disponible: https://en.wikipedia.org/wiki/Emotion_classification
- [17] Daniel Colombo, «Conoce la rueda de las emociones de Robert Plutchik, explicada por Daniel Colombo». [En línea]. Disponible: <https://www.danielcolombo.com/conoce-la-rueda-de-las-emociones-de-robert-plutchik-explicada-por-daniel-colombo/>
- [18] E. R. Kandel, J. H. Schwartz, T. M. Jessell. «Principles of Neural Science», Mc Graw Hill.
- [19] I. Bisio, A. Delfino, F. Lavagetto, M. Marchese. «Opportunistic Detection Methods for Emotion-Aware Smartphone Applications». [En línea]. Disponible: https://www.researchgate.net/publication/258240889_Opportunistic_Detection_Methods_for_Emotion-Aware_Smartphone_Applications
- [20] Abeer Al-Nafjan, Manar Hosny, Yousef Al-Ohali, Areej Al-Wabil. «Review and Classification of Emotion Recognition Based on EEG Brain-Computer Interface System Research: A Systematic Review»
- [21] Sander Koelstra, Christian Mühl, Mohammad Soleymani, Jong-Seok Lee, Ashkan Yazdani, Touradj Ebrahimo, Thierry Pun, Anton Nijholt, Ioannis Patras. «DEAP: A Database for Emotion Analysis using Physiological Signals»
- [22] Erim Yurci. «Emotion Detection from EEG Signals: Correlating Cerebral Cortex Activity with Music Evoked Emotion»
- [23] Wikipedia. «10-20 System». [En línea]. Disponible: [https://en.wikipedia.org/wiki/10-20_system_\(EEG\)](https://en.wikipedia.org/wiki/10-20_system_(EEG))
- [24] I. Daly, D. Williams, A. Kirke, J. Weaver, A. Malik, F. Hwang, E. Miranda, S.J. Nasuto. «Affective brain-computer music interfacing»
- [25] I. Daly, L. Chen, S.J. Zhou, J. Jin. «An investigation into the use of six facially encoded emotions in brain-computer interfacing»
- [26] G.G. Knyazev, J.Y. Slobodskoj-Plusnin, A.V. Bocharov. «Gender differences in implicit and explicit processing of emotional facial expressions as revealed by event-related theta synchronization»
- [27] K. Ishino, M. Hagiwara. «A feeling estimation system using a simple electroencephalograph»
- [28] K. Takahashi. «Remarks on emotion recognition from bio-potential signals»
- [29] D.O. Bos. «EEG-based Emotion Recognition: The Influence of Visual and Auditory Stimuli»
- [30] G. Chanel, J. J. M. Kierkels, M. Soleymani, T. Pun. «Short-term emotion assessment in a recall paradigm»
- [31] Q. Zhang, M. H. Lee. «Analysis of positive and negative emotions in natural scene using brain activity and GIST»
- [32] Y.-P Lin, C.-H. Wang, T.-L. Wu, S.-K. Jeng, J.-H. Chen. «EEG-Based Emotion Recognition in Music Listening»

- [33] Rafael Ramirez, Zacharias Vamvakousis. «Detecting Emotion from EEG Signals Using the Emotive Epos Device»
- [34] M.B. Kostyunina, M.A. Kulikov. «Frequency characteristics of EEG spectra in the emotions»
- [35] N.V. Shemyakina, S.G. Danko. «Influence of the emotional perception of a signal on the electroencephalographic correlates of creative activity»
- [36] F. Schiffer, M.H. Teicher, C. Anderson, A. Tomoda, A. Polcari, C.P. Navalta, S.L. Andersen. «Determination of hemispheric emotional valence in individual subjects: A new approach with research and therapeutic implications»
- [37] W. Heller, J.B. Nitschke, D.L. Lindsay. «Neuropsychological correlates of arousal in self-reported emotion»
- [38] L.I. Aftanas, N.V. Reva, A.A. Varlamov, S.V. Pavlov, V.P. Makhnev. «Analysis of evoked EEG synchronization and desynchronization in conditions of emotional activation in humans: Temporal and topographic characteristics»
- [39] C.M. Krause, V. Viemerö, A. Rosenqvist, L. Sillanmäki, and T. Aström. «Relative electroencephalographic desynchronization and synchronization in humans to emotional film content: an analysis of the 4-6, 6-8, 8-10, and 10-12 hz frequency bands»
- [40] Sander Koelstra, Christian Mühl, Mohammad Soleymani, Jong-Seok Lee, Ashkan Yazdani, Touradj Ebrahimo, Thierry Pun, Anton Nijholt, Ioannis Patras. «DEAP: A Database for Emotion Analysis using Physiological Signals»
- [41] R. Yuvaraj, M. Murugappan, N. M. Ibrahim, M. I. Omar, K Sundaraj, K. Mohamad, R. Palaniappan, E. Mesquita, M. Satiyan. «On the analysis of EEG power, frequency and asymmetry in parkinson's disease during emotion processing»
- [42] Wikipedia. «Análisis Discriminante Lineal» [En línea]. Disponible: https://es.wikipedia.org/wiki/An%C3%A1lisis_discriminante_lineal
- [43] Wikipedia. «Linear Discriminant Analysis» [En línea]. Disponible: https://en.wikipedia.org/wiki/Linear_discriminant_analysis
- [44] Tim Thatcher. «Discriminant Analysis» [En línea]. Disponible: <https://discriminantanalysis.readthedocs.io/en/latest/>
- [45] G. James, D. Witten, T. Hastie, and R. Tibshirani. «An introduction to statistical learning: with applications in r»
- [46] J. Olias, R. Martín-Clemente, M. A. Sarmiento-Vega, S. Cruces «EEG Signal Processing in MI-BCI Applications with Improved Covariance Matrix Estimators»
- [47] Tan, Steinbach, Kumar. «Introduction to Data Mining» [En línea]. Disponible: <https://slideplayer.com/slide/3829911/>
- [48] Wikipedia. «Random Forest» [En línea]. Disponible: https://es.wikipedia.org/wiki/Random_forest

- [49] Univerisdad de Sevilla, asignatura Sistemas Inteligentes, tema 3. «Clasificadores Bayesianos»
- [50] Wikipedia. «K-Nearest Neighbors Algorithm» [En línea]. Disponible:
https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm
- [51] Univerisdad de Sevilla, asignatura Sistemas Inteligentes, tema 4. «Máquinas de Soporte Vectorial»
- [52] AI Trends. «Support Vector Machines (SVM) for AI Self-Driving Cars» [En línea]. Disponible:
<https://www.aitrends.com/ai-insider/support-vector-machines-svm-ai-self-driving-cars/>
- [53] Harish Kandan. «Understanding the kernel trick» [En línea]. Disponible:
<https://towardsdatascience.com/understanding-the-kernel-trick-e0bc6112ef78>
- [54] Wikipedia. «Neurona» [En línea]. Disponible:
<https://upload.wikimedia.org/wikipedia/commons/6/62/Neurona.svg>
- [55] Univerisdad de Sevilla, asignatura Sistemas Inteligentes, tema 8. «Redes neuronales artificiales»
- [56] Digital Trends. «What is an artificial neural network?» [En línea]. Disponible:
<https://www.digitaltrends.com/cool-tech/what-is-an-artificial-neural-network/>
- [57] L. Guesmi, H. Fathallah, M. Menif. «Modulation Format Recognition Using Artificial Neural Networks for the Next Generation Optical Networks» [En línea]. Disponible:
https://www.researchgate.net/figure/McCulloch-Pitts-computational-model-of-a-neuron_fig2_323465059
- [58] Univerisdad de Sevilla, asignatura Sistemas Inteligentes, tema 6. «Reducción del espacio de características»

GLOSARIO

EEG: Electroencefalograma.

HCI: *Human Computer Interaction* (Interacción persona-ordenador).

BCI: *Brain Computer Interface* (Interfaz cerebro-ordenador).

KNN: *K Nearest Neighbors* (K vecinos más cercanos).

SVM: *Support Vector Machines* (Máquinas de soporte vectorial).

ANN: *Artificial Neural Networks* (Redes neuronales artificiales).

