



UNIVERSIDAD DE VALLADOLID



Escuela Técnica Superior de Ingenieros de Telecomunicación  
*Universidad de Valladolid*

## TRABAJO FIN DE GRADO

GRADO EN INGENIERÍA DE TECNOLOGÍAS DE  
TELECOMUNICACIÓN

AYUDA AL DIAGNÓSTICO DEL TDAH EN LA INFANCIA  
MEDIANTE TÉCNICAS DE PROCESADO DE SEÑAL Y  
APRENDIZAJE

---

AUTOR: PATRICIA AMADO CABALLERO  
TUTOR: CARLOS ALBEROLA LÓPEZ  
J.P. CASASECA DE LA HIGUERA

3 de abril de 2019

TÍTULO: AYUDA AL DIAGNÓSTICO DEL TDAH EN LA INFANCIA MEDIANTE TÉCNICAS DE PROCESADO DE SEÑAL Y APRENDIZAJE  
AUTOR: PATRICIA AMADO CABALLERO  
TUTOR: CARLOS ALBEROLA LÓPEZ  
J.P. CASASECA DE LA HIGUERA  
DEPARTAMENTO: TSCIT

### **Miembros del Tribunal**

PRESIDENTE: CARLOS ALBEROLA LÓPEZ  
SECRETARIO: RODRIGO DE LUIS GARCÍA  
VOCAL: J.P. CASASECA DE LA HIGUERA  
SUPLENTE 1: SANTIAGO AJA FERNÁNDEZ  
SUPLENTE 2: FEDERICO SIMMROSS WATTENBERG  
CALIFICACIÓN:

### **Resumen del proyecto**

El Trastorno por Déficit de Atención e Hiperactividad es un trastorno crónico de carácter neuroconductual cuya detección y diagnóstico se enmarca principalmente en la infancia, siendo uno de los trastornos psiquiátricos infantiles más frecuentes. Su diagnóstico actual, marcado fuertemente por la subjetividad de las pruebas, conduce a la necesidad de establecer un método de diagnóstico objetivo. En el presente trabajo, se plantea un método de diagnóstico rápido, fiable y no intrusivo basado en un sistema experto, donde se aúnan la utilización de registros de actigrafía con técnicas de aprendizaje profundo (Deep Learning), aprovechando los avances en materia de hardware que se han sucedido en los últimos años con el desarrollo de las unidades de procesamiento gráfico (GPUs). Para la creación de este sistema experto, se ha realizado primero un procesamiento sobre los registros actigráficos, dividiéndolos en ventanas de actividad de diferentes duraciones y transformando estas nuevas señales en imágenes gracias al espectrograma, sirviendo así como datos de entrada para una red neuronal convolucional, núcleo central del sistema de diagnóstico. Una vez realizada esta primera clasificación de ventanas, se ha llevado a cabo una decisión de diagnóstico para cada paciente en función de la distribución obtenida para dichas ventanas.

### **Palabras clave**

TDAH, Actigrafía, CNN, espectrograma, Aprendizaje Profundo, Clasificación

### **Keywords**

ADHD, Actigraphy, CNN, spectrogram, Deep Learning, Classification

### **Abstract**

Attention Deficit Hyperactivity Disorder is a chronic neurobehavioral disorder that is known and diagnosed mainly in childhood, being one of the most frequent psychiatric disorders in children. Its current diagnosis, strongly marked by the subjectivity of the tests, leads to need to establish an objective diagnostic method. In this work, we propose a rapid, reliable and non-intrusive method based on an expert system that combines the use of actigraphic registries with Deep Learning techniques, taking advantage of the hardware progress that have happened in recent years with the development of graphics processing units (GPU). To create this expert system, we have first processed the actigraphic registries, dividing them into activity windows of different durations, transforming these new signals into images using the spectrogram and feeding them as input data to a convolutional neuronal network, central core of the diagnostic system. After this first classification for the windows, a diagnostic decision has been made for each patient taking the distribution obtained for these windows.

## AGRADECIMIENTOS

En primer lugar, me gustaría agradecerle a mis tutores, Carlos Alberola y Pablo Casaseca la confianza depositada en mí para realizar este trabajo, todo lo que he aprendido estos meses y el tiempo que han invertido en ayudarme a que este trabajo llegara a buen término.

También me gustaría agradecer al Laboratorio de Procesado de Imagen el haberme proporcionado los medios necesarios para el desarrollo de este trabajo y a mis compañeros, que me han ayudado siempre que lo he necesitado.

Me gustaría agradecerle a mi familia todo el apoyo que me han dado durante estos años de carrera. A mis abuelos, mis tíos, mi hermano y en especial a mis padres. Gracias Papá y Mamá por todo lo que habéis hecho por mí. Por haber sido mi apoyo incondicional todo este tiempo, ayudándome a levantarme cuando me caía, llorando y riendo conmigo y sufriendo y festejando las buenas noticias tanto o más que yo. Gracias de corazón.

# ÍNDICE GENERAL

<b>1. Introducción</b>	<b>1</b>
1.1. El diagnóstico del TDAH	1
1.2. Objetivos	2
1.3. Fases y Métodos	2
1.4. Medios necesarios empleados para el desarrollo	3
1.5. Estructura del documento	4
<b>2. Soluciones al problema del TDAH</b>	<b>5</b>
2.1. Técnicas de Imagen Médica (MRI)	5
2.1.1. Radiomics: El futuro de la precisión en la imagen médica	5
2.2. Señales biomédicas	6
2.2.1. Señales bioeléctricas: EEG	6
2.2.2. Señales bioquímicas: HEG	6
2.3. Polisomnografía	6
2.4. Actimetría	7
2.4.1. Otros usos de la actimetría con técnicas de aprendizaje automático	7
2.5. Estado del Arte: Análisis de los resultados obtenidos hasta el momento	7
2.5.1. Estudios previos sobre actimetría	8
2.5.2. Publicaciones que emplean la actimetría y Deep Learning para el diagnóstico	8
2.5.3. Estudios que emplean Deep Learning para el diagnóstico con otras técnicas para la adquisición de las señales	8
2.5.4. Resumen de los métodos de diagnóstico objetivos hasta el momento	9
2.6. Análisis y comparativa de los métodos	13
<b>3. Reconocimiento de patrones</b>	<b>14</b>
3.1. Estructura de un sistema de Reconocimiento de Patrones	14
3.1.1. Extracción de características	14
3.1.2. Clasificación Supervisada	15
3.1.2.1. Clasificadores KNN	15
3.1.2.2. Clasificadores en Árbol de Decisión	16
3.1.2.3. Clasificadores SVM	17
3.2. Técnicas para evaluar el rendimiento del sistema	18
3.2.1. Validación cruzada	19
3.2.2. Métricas para evaluar la validez de los modelos de clasificación	20
3.2.3. Curva Roc	21
<b>4. Deep Learning</b>	<b>23</b>
4.1. Introducción al Deep Learning	24
4.1.1. Tipos de redes para Aprendizaje Supervisado	25

4.1.1.1.	<i>FeedForward Neural Network (FNN)</i>	26
4.1.1.2.	<i>Recurrent Neural Network (RNN)</i>	27
4.1.1.3.	<i>Convolutional Neural Network (CNN)</i>	27
4.2.	<i>Deep learning vs Machine Learning tradicional</i>	29
4.2.1.	<i>Ejemplos de aplicaciones del Deep Learning a señales biomédicas</i>	30
<b>5.</b>	<b>Metodología</b>	<b>32</b>
5.1.	<i>Descripción de las señales de entrada</i>	32
5.1.1.	<i>Preprocesamiento: Análisis de la potencia de las Señales</i>	34
5.2.	<i>Adecuación de los datos</i>	36
5.2.1.	<i>Espectrograma: Representación 2D de señales 1D</i>	36
5.2.2.	<i>Generación de espectrogramas para las señales actigráficas del estudio</i>	37
5.3.	<i>Diseño de la CNN</i>	43
5.4.	<i>Entrenamiento de la red</i>	43
5.4.1.	<i>Parámetros de entrenamiento</i>	44
5.4.2.	<i>Características de los entrenamientos</i>	45
5.5.	<i>Obtención de Resultados: Análisis completo de cada paciente</i>	47
<b>6.</b>	<b>Resultados y discusión</b>	<b>49</b>
6.1.	<i>Prestaciones de la red convolucional para la clasificación de ventanas de actividad</i>	49
6.2.	<i>Clasificación de los pacientes a partir de los resultados obtenidos de la red convolucional</i>	50
6.2.1.	<i>Análisis de la curva ROC para casos concretos</i>	53
6.3.	<i>Prestaciones de la CNN para diferentes proporciones de los conjuntos de entrenamiento y test</i>	56
6.4.	<i>Estado del Arte: Comparativa con otros modelos</i>	58
6.4.1.	<i>Comparativa con estudios anteriores</i>	58
6.4.2.	<i>Comparativa con otros trabajos que emplean la actimetría y aprendizaje profundo para el diagnóstico</i>	59
6.4.3.	<i>Estudio de Muñoz et al</i>	59
6.4.4.	<i>Otros estudios con actimetría</i>	59
6.4.5.	<i>Comparativa con otros trabajos que emplean aprendizaje profundo para el diagnóstico</i>	60
<b>7.</b>	<b>Conclusiones y líneas futuras</b>	<b>61</b>
7.1.	<i>Conclusiones</i>	61
7.2.	<i>Limitaciones y Líneas Futuras</i>	62
<b>A.</b>	<b>Adaptación para otros entornos de programación</b>	<b>69</b>
A.1.	<i>Tensorflow</i>	69
A.2.	<i>Creación de una CNN sobre Tensorflow</i>	70
A.2.1.	<i>Código para la creación de una CNN sobre Tensorflow</i>	70
<b>B.</b>	<b>Clasificación global de pacientes</b>	<b>73</b>
B.1.	<i>Agrupación de las ventanas de cada paciente</i>	73
B.2.	<i>Obtención de los porcentajes de ventanas de caso o control</i>	74
B.3.	<i>Selección de etiquetas y creación de conjuntos de clasificadores</i>	74

# ÍNDICE DE FIGURAS

3.1. Ejemplo de funcionamiento del algoritmo K-NN . . . . .	16
3.2. Ejemplo de algoritmo de árbol de decisión . . . . .	17
3.3. Obtención del hiperplano de Margen Máximo en un SVM lineal . . . . .	17
3.4. Ejemplo de clasificador SVM con kernel RBF de $\gamma = 100$ y con kernel polinómico de orden $q=2$ . . . . .	18
3.5. Ejemplo de funcionamiento de Validación con K-Folds para 3 iteraciones . . . . .	19
3.6. Ejemplo de funcionamiento de Validación Cruzada aleatoria para 3 iteraciones . . . . .	19
3.7. Ejemplo de funcionamiento de Validación Leave One Out para un conjunto de 18 muestras . . . . .	20
3.8. Curva ROC típica . . . . .	22
4.1. Jerarquía de la inteligencia artificial . . . . .	23
4.2. Esquema de funcionamiento de una neurona artificial . . . . .	24
4.3. Gráfico de funcionamiento de un sistema de aprendizaje por refuerzo . . . . .	25
4.4. Estructura básica del Perceptrón Multicapa . . . . .	26
4.5. Estructura básica del Perceptrón Multicapa para una red recurrente . . . . .	27
4.6. Estructura básica de un red Convolutiva . . . . .	28
4.7. Funcionamiento de la operación de convolución . . . . .	28
4.8. Funcionamiento de la operación de Max Pooling . . . . .	29
5.1. Etapas de la adecuación de los registros de información, previa al preprocesado . . . . .	33
5.2. Histograma inicial de Potencias para ventanas de 300 segundos para Casos y Controles . . . . .	34
5.3. Histograma final de Potencias para ventanas de 300 segundos para Casos y Controles . . . . .	35
5.4. Espectrograma de ejemplo para ventanas de 300 segundos. . . . .	37
5.5. Espectrogramas para actividades de 300 segundos con overlap máximo y tamaño de ventana 30 y 200 respectivamente . . . . .	38
5.6. Imágenes de entrenamiento para actividades de 300 segundos con overlap máximo y tamaño de ventana 30 y 200 respectivamente . . . . .	38
5.7. Imágenes de entrenamiento para actividades de 300 segundos con overlap del 50 % y del 67 % respectivamente. . . . .	39
5.8. Imágenes de entrenamiento para actividades de 300 segundos con overlap máximo y tamaño de ventana 200 con DFT de 128 puntos y 512 puntos respectivamente . . . . .	39
5.9. Espectrogramas para ventanas de tamaño 2,6 y 40 respectivamente . . . . .	41
5.10. Imágenes de entrenamiento para una DFT de 128 ,256 y 512 puntos respectivamente. . . . .	41
5.11. Esquema de la generación de las imágenes que nos servirán de entrada para la CNN. . . . .	42
5.12. Estructura de la red Convolutiva utilizada para los entrenamientos . . . . .	43
5.13. Proceso de entrenamiento del sistema de diagnóstico. . . . .	44
5.14. Entrenamiento para la actividad diurna . . . . .	47
5.15. Proceso de entrenamiento para la clasificación por pacientes . . . . .	48

---

6.1. <i>Boxplots de Exactitud, Sensibilidad y Especificidad para los cinco clasificadores utilizados (KNN, Árbol, SVM, SVM con RBF y SVM con <math>q=2</math>)</i> . . . . .	52
6.2. <i>CURVAS ROC para el peor fold en ventanas de 60 segundos</i> . . . . .	53
6.3. <i>Matriz de confusión para el caso de peor AUC</i> . . . . .	54



# ÍNDICE DE TABLAS

2.1. Resumen de estudios más relevantes para el TDAH que no están basados en actigrafía (1/2) . . . .	10
2.2. Resumen de estudios más relevantes para el TDAH que no están basados en actigrafía (2/2) . . . .	11
2.3. Resumen de estudios de actigrafía . . . . .	12
3.1. Estructura de la Matriz de confusión . . . . .	21
5.1. Ventanas iniciales para cada grupo de actividad . . . . .	33
5.2. Ventanas después del análisis de la potencia para cada grupo de actividad . . . . .	36
5.3. Resumen de generación de espectrogramas para actividades de corta duración . . . . .	40
5.4. Parámetros utilizados en los entrenamientos . . . . .	45
5.5. Imágenes Utilizadas en el entrenamiento para un fold cualquiera . . . . .	45
5.6. Porcentaje medio de imágenes empleadas en el entrenamiento . . . . .	46
5.7. Duración media de los entrenamientos para cada fold . . . . .	46
6.1. Accuracy obtenida del entrenamiento de la CNN para el conjunto de test . . . . .	49
6.2. Matriz de confusión para el mejor fold obtenido en el entrenamiento . . . . .	50
6.3. Métricas para ventana de 1800s . . . . .	50
6.4. Métricas para ventana de 300s . . . . .	51
6.5. Métricas para ventana de 60s . . . . .	51
6.6. AUCs de los clasificadores ventana de 60s . . . . .	53
6.7. AUCs de los clasificadores ventana de 60s . . . . .	55
6.8. AUCs de los clasificadores ventana de 60s . . . . .	55
6.9. Accuracy obtenida del entrenamiento de la CNN para el conjunto de test . . . . .	56
6.10. Métricas para ventana de 1800s . . . . .	57
6.11. Métricas para ventana de 300s . . . . .	57
6.12. Métricas para ventana de 60s . . . . .	57
6.13. Métricas mejores obtenidas para ambos estudios . . . . .	58
6.14. Métricas mejores obtenidas para ambos estudios . . . . .	59
6.15. Métricas mejores obtenidas para ambos estudios . . . . .	59
6.16. Métricas mejores obtenidas para ambos estudios . . . . .	60



## Capítulo 1

# INTRODUCCIÓN

En este capítulo, se busca establecer el contexto del presente trabajo, donde se plantea el empleo de señales de actigrafía junto a técnicas de Deep Learning como método para el diagnóstico objetivo del trastorno por déficit de atención e hiperactividad (TDAH). Este trastorno constituye uno de los trastornos psiquiátricos infantiles más frecuentes y su diagnóstico está basado principalmente en criterios clínicos, lo que determina que este tenga un marcado carácter subjetivo. Es por tanto la necesidad de establecer un método de diagnóstico basado en la objetividad el principal motor y objetivo que persigue el presente trabajo.

Así pues, a lo largo de las siguientes secciones se pretende introducir brevemente la patología y sus métodos de diagnóstico actuales, establecer la estructura del trabajo y hablar brevemente de los medios y métodos empleados para su desarrollo.

## 1.1 EL DIAGNÓSTICO DEL TDAH

---

El trastorno por déficit de atención e hiperactividad es un trastorno crónico de carácter neuroconductual cuya detección y diagnóstico se enmarca principalmente en la infancia, y sus consecuencias pueden también observarse en la edad adulta. Se caracteriza, principalmente, por la aparición de patrones de inatención, hiperactividad e impulsividad anómalos respecto a la edad y que no puedan ser explicados debido a la existencia de otros trastornos. No obstante, la observación de estos comportamientos no garantiza la existencia de la patología. Para que esta sea considerada, deben darse unas condiciones muy precisas, que conllevan la aparición de dichos patrones en más de un ámbito de la vida del niño. En líneas generales, los individuos que son diagnosticados con este trastorno suelen presentar problemas para relacionarse en el colegio, además de problemas en el rendimiento académico que al final terminan afectando al día a día[1].

El diagnóstico consiste en un análisis clínico basado en el cumplimiento de los criterios marcados en el *Diagnostic And Statistical Manual Of Mental Disorders (DSM-V)*. Principalmente, consiste en la recopilación de datos a lo largo de seis meses en los que se realizan diferentes entrevistas con los padres y el niño, una evaluación de la información aportada por los profesores, exámenes físicos y pruebas complementarias (nivel intelectual del niño, pruebas del lenguaje y específicas

del aprendizaje, etc). Con toda esta información, los profesionales médicos (neuropediatra, psiquiatra infantil, psiquiatra o neurólogo) realizarán un diagnóstico definitivo.

En la actualidad, se calcula que la prevalencia global del TDAH es de 5,29 % en niños en edad escolar. Estos datos fluctúan en función de los criterios diagnósticos, el método de evaluación, el tipo de muestra, las fuentes de información utilizadas y las características socioculturales de la población evaluada.

## 1.2 OBJETIVOS

---

Como hemos visto anteriormente, en la actualidad el principal método de diagnóstico del TDAH está fuertemente unido a la objetividad de las respuestas proporcionadas por parte del entorno del paciente a los profesionales de la salud. En estudios anteriores, ya se ha observado que el uso de la actimetría para el diagnóstico de esta patología podría garantizar un sistema de diagnóstico rápido, fiable y no intrusivo a la par que económico [2].

En este trabajo se busca ir más allá, planteando la utilización conjunta de técnicas de aprendizaje, que nos faciliten la tarea de extracción de características y estas señales de actimetría. En la actualidad, los avances en el hardware; principalmente con la utilización de GPUs (unidades de procesamiento gráfico) y cálculo en paralelo, han permitido llevar a la práctica los sistemas de aprendizaje inteligente, cuyo principio de funcionamiento se basa en el aprendizaje mediante ejemplos. Aplicado a nuestro problema, estas técnicas nos permitirán diseñar un sistema experto que, mediante una muestra homogénea de pacientes diagnosticados previamente como sujetos sanos o con TDAH, permita determinar correctamente si un paciente cuyo diagnóstico es desconocido para el mismo ha sido o no diagnosticado como tal. Siguiendo esta premisa y de forma más detallada, con este trabajo se pretende:

- Diseñar un sistema experto de aprendizaje inteligente que permita la detección del TDAH en pacientes desconocidos para el sistema.
- Maximizar la eficiencia de la red, encontrando para ello la forma óptima de definir las entradas a la misma a partir de los datos de partida.
- Conseguir un sistema de mejores prestaciones en materia de clasificación de pacientes que los actualmente descritos en la literatura.

## 1.3 FASES Y MÉTODOS

---

En la sección anterior se han planteado los objetivos que se pretenden alcanzar para conseguir resolver el problema al que nos enfrentamos. Para cumplir con dichos objetivos se ha realizado una mecánica de trabajo de acuerdo con las siguientes fases:

- División de la señal de cada paciente en periodos de actividad diurna y nocturna, con el objetivo de realizar un análisis independiente de cada periodo.

- Adecuación de la información actigráfica de cada periodo, fragmentando la señal de cada paciente en ventanas de tiempo más pequeñas que serán consideradas como entradas independientes de la red.
- Una vez realizada la fragmentación, se analizará la información: se estudiará la potencia de cada ventana, y se eliminarán aquellas en las que se registre una actividad muy pequeña, con el objetivo de eliminar aquellos valores que no aporten información al procesamiento.
- Adaptación de los datos para poder ser utilizados como datos de entrada para una red neuronal convolucional (CNN). Este tipo de redes, están pensadas para trabajar con imágenes como datos de entrada, por lo que realizaremos el espectrograma de cada una de las ventanas para obtener una imagen por cada una de ellas.
- Diseño y definición de los parámetros de la CNN que va a funcionar como núcleo central del sistema de diagnóstico.
- Entrenamiento de la red de forma independiente para las señales del día y de la noche, con diferentes conjuntos de entrenamiento.
- Analizar las prestaciones de diferentes clasificadores bidimensionales en términos de exactitud, sensibilidad y especificidad para la clasificación de cada sujeto en base a los resultados obtenidos de los entrenamientos de la CNN.
- Exposición de los resultados obtenidos y discusión sobre los mismos.

## 1.4 MEDIOS NECESARIOS EMPLEADOS PARA EL DESARROLLO

---

Para la correcta realización del proyecto planteado, será necesaria la utilización de diferentes herramientas de *hardware* y *software*:

### Hardware

- PC de sobremesa con distribución Linux y las siguientes características:
  - Procesador 8xIntel®Core™i7-4790 3.60 GHz.
  - 16 GB de memoria RAM.
  - Disco duro HDD de 500 GB de capacidad.
  - GPU NVIDIA GeForce GTX 970 con 4 GB de memoria RAM

### Software

- MATLAB R2017b [3]: Software de lenguaje de alto nivel que permite el desarrollo de algoritmos, el análisis de datos y computación numérica.
- L<sup>A</sup>T<sub>E</sub>X : Software utilizado para la composición de textos.

---

## 1.5 ESTRUCTURA DEL DOCUMENTO

---

El presente documento consta de seis capítulos, incluyendo el capítulo de introducción. El contenido de los posteriores capítulos es el que sigue:

**Capítulo 2- Soluciones al problema del TDAH.** En este capítulo se presentan las soluciones planteadas hasta el momento para la resolución del diagnóstico del TDAH en términos de reconocimiento de patrones y procesamiento de señal.

**Capítulo 3- Deep Learning** En este capítulo se realiza una introducción explicativa de las diferentes técnicas de *Deep Learning*, los diferentes tipos de redes que se pueden plantear con esta filosofía y las principales diferencias con las técnicas de *Machine Learning*.

**Capítulo 4- Metodología** A lo largo de este capítulo se detallan los procedimientos realizados para el preprocesado de las señales, referido este tanto a la adecuación de la información y preparación de la misma para la utilización de la red como a la definición, diseño y posterior entrenamiento de la red convolucional. Posteriormente, se introducen los diferentes métodos de clasificación utilizados para obtener los resultados finales del sistema de diagnóstico planteado.

**Capítulo 5- Resultados y discusión** En este capítulo se procede al análisis y estudio de los resultados obtenidos tanto de la clasificación como del entrenamiento y se realizan las correspondientes comparativas con lo conocido hasta el momento.

**Capítulo 6- Conclusiones y líneas futuras** En este último capítulo se realiza una visión global del Trabajo fin de Grado realizado, detallando aquí las conclusiones extraídas del mismo. Además, se hace mención a las dificultades encontradas a lo largo de su realización y se plantean las líneas de trabajo futuras que pueden surgir a raíz de los resultados obtenidos.

Además del capítulo en el que se incluyen las referencias bibliográficas que sustentan este proyecto, se han añadido apéndices con el objetivo de servir de información aclaratoria para el presente trabajo.

**Apéndice A- Adaptación de la red planteada para otros entornos de programación.** Contiene el código en python de la CNN para poder usarla con las bibliotecas de Keras y Tensorflow.

**Apéndice B- Clasificación global de pacientes.** Código utilizado para poder utilizar los porcentajes de accuracy de la CNN como entradas para los clasificadores bidimensionales utilizados en la clasificación global.

## Capítulo 2

# SOLUCIONES AL PROBLEMA DEL TDAH

Como se ha mencionado en el capítulo anterior, el principal método de diagnóstico para el TDAH consiste en la aplicación de los criterios marcados en el *Diagnostic And Statistical Manual Of Mental Disorders (DSM-V)*. No obstante, existen estudios que plantean el uso de otras técnicas de diagnóstico, basadas en métodos objetivos que permitirían diagnosticar esta patología. En las secciones sucesivas, haremos un breve repaso de algunos de ellos.

### 2.1 TÉCNICAS DE IMAGEN MÉDICA (MRI)

---

La **imagen por resonancia magnética (MRI)** es una técnica no invasiva común en la medicina para la detección de anomalías en los tejidos. Su funcionamiento consiste principalmente en la utilización de campos magnéticos (en lugar de radiaciones ionizantes) para la obtención de las diferentes imágenes.

En el contexto de nuestro problema, se han encontrado diferentes estudios que emplean esta técnica para obtener imágenes del cerebro que permitan detectar diferencias entre los pacientes diagnosticados con TDAH y los pacientes de control. El empleo de imágenes volumétricas del cerebro [4] o del estudio de los niveles de hierro en el cerebro [5] son dos ejemplos del empleo de esta técnica para la búsqueda de un sistema de diagnóstico objetivo.

En algunos estudios sobre el TDAH [6] también se ha empleado una variante de esta técnica conocida como **imagen por resonancia magnética funcional (fMRI)** que permite a mayores el estudio de las zonas activas del cerebro.

#### 2.1.1 RADIOMICS: EL FUTURO DE LA PRECISIÓN EN LA IMAGEN MÉDICA

---

**Radiomics** se refiere a la extracción masiva de características de imágenes médicas, bajo la premisa de que una imagen es en realidad un gran conjunto de datos. Estas características, denominadas **características radiológicas**, proporcionan información relativa a la imagen que no podemos apreciar a simple vista, lo que permite descubrir factores y características de diversas enfermedades antes desconocidos [7].

En relación con el TDAH, el empleo conjunto de esta técnica con la obtención de imágenes por

resonancia magnética, ha proporcionado resultados prometedores, tanto en la separación entre pacientes sanos y enfermos, como en la clasificación de los diferentes subtipos de TDAH. [8].

## 2.2 SEÑALES BIOMÉDICAS

---

Las **señales biomédicas** constituyen todas aquellas señales originadas por el cuerpo que pueden ser utilizadas para el diagnóstico o la investigación médica. Suelen clasificarse en base a su origen y su obtención se realiza mediante sensores biomédicos.

Existen diferentes señales que pueden resultar de interés para el diagnóstico del TDAH.

### 2.2.1 SEÑALES BIOELÉCTRICAS: EEG

---

El **electroencefalograma** (EEG) es una técnica que permite el estudio del sistema nervioso central. Consiste en la obtención; a través de electrodos, de las señales eléctricas generadas en los impulsos nerviosos. Esta técnica no invasiva, se utiliza de forma habitual para la detección de diversas patologías relacionadas con el funcionamiento del cerebro; como pueden ser la epilepsia o determinados tumores cerebrales.

Analizando diferentes estudios que emplean múltiples técnicas para la obtención del EEG [9, 10, 11, 12] puede observarse claramente que se obtienen diferencias entre pacientes sanos y enfermos. No obstante; aunque se presente como un método eficaz para la diferenciación, su coste elevado y la naturaleza de la patología cuestionan su verdadera eficacia como método de diagnóstico seguro [13].

### 2.2.2 SEÑALES BIOQUÍMICAS: HEG

---

Otra de las líneas de investigación para el estudio fisiológico del cerebro se centra en la bioquímica del cerebro. En algunos estudios [14, 15, 16] se ha planteado la **hemoencefalografía** (HEG) como un posible método de diagnóstico. Para la obtención de las señales, se emplean sensores de infrarrojo; lo que hace que la adquisición sea cómoda y sencilla. Mediante esta técnica se estimula el riego sanguíneo, lo que genera una mayor vascularización en localizaciones del lóbulo prefrontal, con el consiguiente aumento en el aporte de oxígeno.

En relación con el TDAH, Rodríguez et al. [14] plantean la eficacia de esta técnica frente al diagnóstico por imagen tradicional; así como frente a la EEG, dado su carácter no invasivo y menor coste.

## 2.3 POLISOMNOGRAFÍA

---

La **polisomnografía** es una prueba de diagnóstico que consiste en el estudio de diferentes parámetros durante el sueño. Mientras el sujeto está dormido o tratando de dormir, se realizan registros de la actividad cerebral (EEG), de la respiración, del ritmo cardiaco, de la actividad



muscular y de los niveles de oxígeno en la sangre. Principalmente, esta prueba se utiliza para diagnosticar los diferentes trastornos del sueño.

No obstante, algunas investigaciones defienden la hipótesis [17] de la relación existente entre el TDAH y los ciclos del sueño y presentan esta prueba como un posible método de diagnóstico objetivo para esta patología. Efectivamente, hay estudios que han demostrado dicha relación [18] en los que se ha interpretado esta técnica como una ayuda para el diagnóstico pero no se considera aún un método de diagnóstico confiable de forma exclusiva.

## 2.4 ACTIMETRÍA

---

La **actimetría** o **actigrafía** es una técnica no invasiva que permite la extracción de señales de movimiento, lo que permite realizar una monitorización de los ciclos de actividad y descanso. La adquisición de los datos se hace típicamente a través de un sensor denominado **actímetro**, cuyo funcionamiento se basa en la obtención del movimiento a través de los tres ejes cartesianos (x,y,z). El actímetro, se coloca normalmente en la muñeca y la toma de las diferentes mediciones no afecta a la vida cotidiana del paciente. Este hecho, hace frente a uno de los hándicap que presentaba la polisomnografía y hace de la actimetría una técnica prometedora en estudio de patologías relacionadas con los trastornos del sueño.

En el marco de nuestro problema, hay estudios previos que han demostrado el potencial de esta técnica para el diagnóstico del TDAH [2]. Tomaremos estos estudios como punto de partida para el desarrollo de este trabajo, aportando un enfoque diferente para el tratamiento de estas señales.

### 2.4.1 OTROS USOS DE LA ACTIMETRÍA CON TÉCNICAS DE APRENDIZAJE AUTOMÁTICO

La utilización de las señales de actimetría ha demostrado ser útil en el estudio de diferentes patologías. El avance de las técnicas de aprendizaje automático ha hecho que la utilización conjunta con dichas señales pueda resultar útil para otro tipo de trastornos relacionados con el espectro neurológico, centrándose en el estudio de las señales durante el sueño. Por ejemplo, Sathyanarayana et al [19] han utilizado ambas técnicas para realizar un estudio sobre la predicción de la calidad del sueño. Otros estudios, como el de Granovsky et al. se han centrado en la búsqueda de patrones en los diferentes estados del sueño. [20]

## 2.5 ESTADO DEL ARTE: ANÁLISIS DE LOS RESULTADOS OBTENIDOS HASTA EL MOMENTO

---

En esta sección, se busca hacer un repaso de las publicaciones previas más recientes relacionadas con las técnicas elegidas en el presente trabajo para realizar el sistema de diagnóstico. Para ello:

- Analizaremos aquellas publicaciones en las que se utilice la actimetría como técnica de obtención de señales para el diagnóstico del TDAH.

- Uso del *Deep Learning* junto con otras técnicas de obtención de señales.

### 2.5.1 ESTUDIOS PREVIOS SOBRE ACTIMETRÍA

---

Los estudios realizados por Martín et al [2, 21], plantean un sistema de diagnóstico basado en dos etapas:

- La primera consistía en dividir la señal en diferentes intervalos de actividad, conocidos como movimientos (actividades de corta duración, entre un minuto y un segundo) y tareas (actividades de larga duración, entre 5 y 30 minutos).
- La segunda se encargaba de la extracción de forma manual de características de dichas señales y de la clasificación multidimensional de las mismas.

En este estudio se consiguieron valores de exactitud de clasificación de **0.8571** con una sensibilidad de **0.9500**.

El problema se encontró en la dificultad de la extracción de características ya que la extracción de características manual, conlleva un trabajo previo muy exhaustivo y largo y en muchas ocasiones no garantiza que estas sean del todo correctas.

### 2.5.2 PUBLICACIONES QUE EMPLEAN LA ACTIMETRÍA Y DEEP LEARNING PARA EL DIAGNÓSTICO

---

En el estudio realizado por Muñoz et al. [22] se empleaban dos acelerómetros; uno de muñeca y otro de tobillo para analizar los datos obtenidos durante 6 horas escolares de un grupo reducido de niños (22 pacientes, 11 diagnosticados como TDAH y 11 sujetos sanos). Para el análisis de estos datos, emplearon una red neuronal convolucional. En términos de resultados, se consiguió una exactitud de **0.8750** en el caso del acelerómetro de muñeca y un **0.9375** para el de tobillo. En cuanto a la sensibilidad, los resultados obtenidos fueron de **0.6** y **0.8** respectivamente.

Otro estudio, realizado en 2014 planteaba el diagnóstico objetivo del TDAH a partir de las señales de actigrafía obtenidas mediante **IMUs** o unidades de medición inercial, que combinan giroscopios y acelerómetros para la obtención de dichas señales [23]. En dicho estudio Mahony et al. utilizaban un clasificador de Machine Learning (SVM) para realizar la clasificación a partir de las *features* con mejor capacidad predictiva extraídas, obteniendo valores de exactitud, sensibilidad y especificidad de **0.9512** , **0.9444** y **0.9565** respectivamente.

### 2.5.3 ESTUDIOS QUE EMPLEAN *Deep Learning* PARA EL DIAGNÓSTICO CON OTRAS TÉCNICAS PARA LA ADQUISICIÓN DE LAS SEÑALES

---

Al comienzo del capítulo se explicaron diversas técnicas utilizadas en la investigación para proporcionar un diagnóstico objetivo de la patología. La mayoría de las técnicas mencionadas, se centraban en la obtención de imágenes médicas de diversos tipos que permitieran distinguir entre sujetos sanos y enfermos.

Existen diversos estudios que aúnan dichas imágenes médicas con técnicas de aprendizaje profundo para proporcionar una solución al diagnóstico. En esta sección, comentaremos algunos de ellos.

En el estudio presentado por Riaz et al [24] se plantea la clasificación de pacientes empleando un clasificador SVM y como datos, los obtenidos del repositorio *NeuroBureau ADHD-200*. Estos datos, consisten en un tipo específico de **resonancia magnética funcional (fMRI)** conocida como *Resting State fMRI* (rsfMRI o R-fMRI) y que evalúa las interacciones regionales que ocurren en un estado de reposo o de tarea negativa, cuando no se está realizando una tarea explícita. En el estudio se alcanzó un accuracy de clasificación de **0.818**.

Otro estudio publicado en 2017, realiza la clasificación utilizando las medidas del potencial relacionado con eventos (ERP) en las señales electroencefalográficas (EEG) durante la realización de una tarea de tipo Stroop. Para obtener las características en el dominio de Tiempo/Frecuencia emplearon la técnica Time Frequency Hermite Atomizer (TFHA). Como clasificador, emplearon un clasificador SVM con eliminación recursiva de *features*.

En relación a los resultados, Öztoprak et al. [25] presentan una accuracy de clasificación del 100 % sobre un conjunto de test de 10 pacientes. En el entrenamiento, consiguieron una exactitud de **0.995** empleando cinco *features*.

En el análisis planteado por Sun et al. [8] se utilizó validación cruzada y bosques aleatorios para la clasificación de pacientes diagnosticados con TDAH, consiguiendo un valor de exactitud de **0.737**.

#### 2.5.4 RESUMEN DE LOS MÉTODOS DE DIAGNÓSTICO OBJETIVOS HASTA EL MOMENTO

En esta sección se resumen los resultados de los métodos más relevantes para el diagnóstico de TDAH . En las tablas 2.1 y 2.2, se muestran los resultados no basados en actigrafía y en la tabla 2.3 un resumen de los estudios de actigrafía. Los resultados se expresan como lo hacen los autores en los trabajos citados.

F. Información	Autores	Sujetos	Métodos	Resultados
<b>Informes Parentales</b>	Cortese et al. [26]	722 pacientes con TDAH y 683 para el grupo de control	Informes subjetivos del sueño proporcionados por los padres	Los mejores resultados obtienen $p < 0,001$ para la resistencia a la hora de dormir, el inicio del sueño y la dificultad para despertarse por la mañana
<b>Análisis Químico</b>	Tuthill et al. [27]	16 pacientes con TDAH y 261 sujetos sanos	Nivel de plomo en el pelo	$p = 0,006$
	Weiss et al. [28]	277 pacientes con TDAH y 106 sujetos sanos	Nivel de TSH en la sangre	Se encontraron niveles anormales en el 2,2% de los pacientes con TDAH y en el 1% de sujetos sanos
	El-Faddagh et al. [29]	129 mujeres y 126 hombres a lo largo de 11 años	Alelo Exon III gen DRD4	$p = 0,061$ en 4.5 y.o., $p = 0,026$ en 8 y.o., $p = 0,005$ en 11 y.o.
	Van Veen et al. [30]	40 sujetos diagnosticados como TDAH y 40 controles emparejados	Melatonina en saliva	Ritmo circadiano retrasado en pacientes con TDAH ( $p < 0,0005$ )
<b>Imagen Médica</b>	Kurtek et al. [31]	19 pacientes con TDAH y 15 controles	Análisis de forma de estructuras cerebrales a partir de resonancia magnética	Exactitud = 0,9118
	Ashtari et al. [32]	18 pacientes con TDAH y 15 sujetos sanos	DT-MRI	Los pacientes con TDAH muestran valores más bajos ADHD patients FA (Anisotropía fraccional) en el cerebro medio: $p < 0,01$
	Silk et al. [33]	15 pacientes con TDAH y 15 sujetos sanos	DT-MRI	Los pacientes con TDAH mostraron valores más altos de FA en diferentes regiones del cerebro: región occipito-parietal derecha $p < 0,001$ , estriado izquierdo $p < 0,001$ , inf. Izquierdo. región temporal $p < 0,005$ , derecha inf. región parietal $p < 0,001$ e izquierda inf. región frontal $p < 0,05$
	Zimmer et al. [34]	Encuesta de diferentes estudios	PET/SPECT	El metabolismo de los pacientes con TDAH presenta varias anomalías: glucosa, dopamina (catecolaminas en general) y noradrenalina.
	Riaz et al. [24]	<i>NeuroBureau ADHD-200</i>	Un tipo de resonancia magnética funcional (fMRI) conocida como <i>Resting State fMRI</i> (rsfMRI o R-fMRI)	Exactitud = 0.818, Sensibilidad=1, Especificidad = 0.75
	Sun et al. [8]	83 niños emparejados por edad y sexo con TDAH recién diagnosticado y nunca tratado (40 con TDAH inatento y 43 con TDAH combinado) y 87 sujetos control sanos	Resonancia magnética anatómica y tensor de difusión Imágenes (MR)	No se encontraron diferencias globales entre los niños con TDAH y los sujetos control en el volumen cerebral total o en el volumen total de materia gris y blanca. Exactitud = 0.737 (discriminación de pacientes) y Exactitud = 0.801 (diferentes subtipos)
	<b>Señales Biomédicas</b>	Modarres-Zadeh et al. [35]	5 pacientes con TDAH y 5 controles emparejados antes y después de la medicación	Análisis espectral EEG y test de comportamiento
Gutierrez et al. [36]		4 pacientes con TDAH no medicados	Relación de la densidad espectral de potencia en las bandas <i>beta</i> y <i>Theta</i> del EEG	La relación se altera en los pacientes con TDAH pero tiende a valores normales después del entrenamiento
Hyun-Lee et al. [37]		13 niños en tres grupos: TDAH, retraso mental y grupo de control	Análisis de wavelets de EEG y mapas autoorganizados	Exactitud = 0,6
Dong-Ming et al. [38]		6 jóvenes	Análisis de EEG por entropía multiescala	Se detectaron diferentes estados de atención con $rmExactitud = 0,63158$ , lo que revela una alta complejidad en la actividad de EEG incluso cuando no se realiza ninguna actividad

TABLA 2.1: Resumen de estudios más relevantes para el TDAH que no están basados en actigrafía (1/2)

F. Información	Autores	Sujetos	Métodos	Resultados
<b>Señales Biomédicas</b>	Nan et al. [39]	6 personas sanas	Análisis del desplazamiento de información mediante EEG cuando se cambia de atención sostenida a falta de atención	$p < 0,005$ en bandas <i>alpha</i> y <i>beta</i> , $p < 0,05$ en la banda <i>Theta</i>
	Hale et al. [40]	35 adultos con TDAH y 104 adultos sanos	Asimetría del EEG en la banda <i>beta</i>	Asimetría atípica en el TDAH ( $p = 0,00001$ )
	Ghasssemi et al. [41]	10 adultos con TDAH y 40 adultos sano	Características morfológicas de las señales ERP	Exactitud = 0,92 en la banda <i>gamma</i>
	Castro-Cabrera et al. [42]	23 sujetos y 23 controles	Análisis de características morfológicas, wavelets y características no lineales de señales ERP.	Morphological: Sensibilidad = 0,96, Especificidad = 0,65, Exactitud = 0,81 $\pm$ 0,1100, AUC = 0,81; dinámicas no lineales: Sensibilidad = 0,69, Especificidad = 0,78, Exactitud = 0,7391 $\pm$ 0,1200, AUC = 0,72; wavelets: Sensibilidad = 1, Especificidad = 0,78, Exactitud = 0,8695 $\pm$ 0,0973, AUC = 0,91; mejor combinación de características: Sensibilidad = 0,96, Especificidad = 0,87, Exactitud = 0,9130 $\pm$ 0,0100, AUC = 0,94
	Demauelle et al. [43]	2 pacientes con TDAH y 2 controles emparejados	Sincronía de fase de señales MEG adquiridas en regiones distantes del cerebro.	No reportado
Oztoprak et al. [25]	Entre 37 y 44 niños en el grupo con TDAH y 32 a 38 niños en el grupo de control sano.	Medidas ERP en las señales EEG durante la realización de una tarea de tipo Stroop.	Exactitud = 0,995 e (train) Exactitud = 1 (10 sujetos test)	
<b>PSG</b>	Gruber et al. [44]	32 pacientes con TDAH y 64 sujetos sanos adultos durante 5 noches consecutivas	Informes de sueño	Los informes de sueño pueden ser usados para el diagnóstico del TDAH
	Silvestri et al. [45]	16 sujetos diagnosticados como TDAH inactivo y 39 con otros tipos de TDAH	Informes de sueño	Tanto los aspectos macro como microestructurales del sueño presentaron diferencias significativas. El sueño de los sujetos con TDAH presentó más trastornos. Las mayores diferencias presentaron $p < 0,0001$
	Cortese et al. [26]	Meta-Analysis incluyendo 722 sujetos con TDAH y 683 controles	Informes de sueño	Los sujetos con TDAH presentan más trastornos del sueño La característica más informativa presento $p < 0,001$
<b>Análisis del movimiento mediante infrarrojos</b>	Teicher et al. [46]	17 sujetos diagnosticados con TDAH y 10 sujetos sanos	Evaluación de los movimientos de la cabeza mediante dispositivos infrarrojos	Sensibilidad= 0,94, Especificidad=1logrado mediante la combinación de la complejidad de los movimientos de la cabeza evaluados mediante el análisis fractal con otras variables significativas (edad, puntuación en una tarea continuada)
	Teicher et al. [47]	18 pacientes con TDAH y 11 de contro	Evaluación de los movimientos de la cabeza mediante dispositivos infrarrojos	Los movimientos en la cabeza para pacientes con TDAH son más frecuentes ( $p < 0,002$ ), 3,4 veces más frecuentes ( $p > 0,01$ ), al cubrir áreas grandes ( $p > 0,001$ ) y más lineales y menos complejos ( $p > 0,00004$ )

TABLA 2.2: Resumen de estudios más relevantes para el TDAH que no están basados en actigrafía (2/2)

F. Información	Autores	Sujetos	Métodos	Resultados
Actividad Diurna	Dane et al. [48]	Se evaluaron 42 sujetos TDAH (20 de tipo inatento, 22 de tipo combinado) y 22 controles.	Registros de Actigrafía de 24 horas de duración promedio en 2h de duración	Solo diferencias durante la tarde. Los tipos de TDAH no presentaron diferencias en el análisis actigráfico.
	Teicher et al. [49]	11 pacientes con TDAH y 12 controles	Registros de Actigrafía de 72 horas promediados y muestreados cada 5 minutos	La distribución de las puntuaciones de actividad (porcentaje de tiempo empleado en actividad baja, media o alta) constituye un índice relevante para el diagnóstico de TDAH (la distribución de la actividad es significativamente sesgado a la derecha en pacientes con TDAH)
	Kam et al. [50]	10 pacientes con TDAH y 7 controles	3 horas de registros de actigrafía muestreados a 32 Hz	La distribución promedio obtenida a partir de los histogramas de actividad de 1 minuto presentó mayor nivel de actividad y variabilidad en el grupo de TDAH ( $p < 0,05$ )
	Martín et al. [2, 21]	31 pacientes con TDAH y 35 controles de 6 años de edad	Registros de Actigrafía durante 24 h muestreados a 1 Hz	Exactitud = 0.8571, Sensibilidad = 0.9500 y Especificidad = 0.7727
	Muñoz et al. [22]	22 pacientes, 11 diagnosticados como TDAH y 11 sujetos sanos	Dos acelerómetros: uno de muñeca y otro de tobillo para analizar los datos obtenidos durante 6 horas escolares y una red convolucional.	Exactitud = 0.8570, Sensibilidad = 0.6 y Especificidad = 1 para la muñeca y Exactitud = 0.937, Sensibilidad = 0.8 y Especificidad = 1 para el tobillo
	Mahony et al. [23]	43 pacientes, 24 con TDAH y 19 controles de 6 a 11 años de edad	Señales de actigrafía obtenidas mediante IMUs o unidades de medición inercial, durante 1h	Exactitud = 0.9512, Sensibilidad = 0.9444 y Especificidad = 0.9565
Informes de Sueño	Cortese et al. [26]	722 sujetos componen el grupo de TDAH y 683 componen el grupo de control	Los informes Actigráficos del sueño	Los niños con TDAH presentaron una menor eficiencia del sueño, un tiempo de sueño real y tiempos promedio para quedarse dormidos. Los resultados más significativos arrojaron $p < 0,001$
	Gruber et al. [44]	38 niños con TDAH y 64 controles durante 5 noches consecutivas	Informes de sueño actigráfico	Se encontró un aumento de la inestabilidad en el inicio del sueño, la eficiencia del sueño y el sueño verdadero en pacientes con TDAH
Ritmo biológico circadiano	Van Veen et al. [30]	40 sujetos diagnosticados como TDAH y 40 controles emparejados	Registros actimétricos de 24 horas	Comienzo y finalización retardados del sueño en sujetos con TDAH. Mayor relevancia lograda para el inicio del sueño: $p = 0,006$

TABLA 2.3: Resumen de estudios de actigrafía

## 2.6 ANÁLISIS Y COMPARATIVA DE LOS MÉTODOS

---

A lo largo de este capítulo se ha hecho un repaso sobre las diferentes líneas de investigación existentes en el diagnóstico objetivo del TDAH. En general, podemos agrupar estas líneas en dos grandes grupos atendiendo a la metodología:

- **Estudios basados en métodos no actimétricos:** Plantean la utilización de MRI y EEG para el diagnóstico. En el primer caso, diversos estudios han conseguido resultados positivos en la clasificación de subtipos de TDAH y razonables para la diferenciación de pacientes diagnosticados. En el caso del EEG, se han obtenido resultados prometedores en cuanto a la diferenciación de pacientes. Son métodos costosos y aunque en algunos casos muestran resultados razonables para el diagnóstico de la patología, la naturaleza de la patología cuestiona si realmente son una opción rentable como diagnóstico.
- **Estudios basados en métodos actimétricos:** Su principal diferencia con el resto de métodos está en que es un método con menor coste que los anteriores y que cumple los requisitos de independencia del observador y ha demostrado ser efectivo en diversos estudios, con resultados bastante buenos.

Con todo lo visto hasta el momento, podemos justificar nuestra propuesta en tres ideas principales:

- La actimetría es más conveniente para el estudio del TDAH, ya que garantiza un método de diagnóstico objetivo y no intrusivo de menor coste y mucho más cómodo para el paciente.
- Los métodos basados en actimetría de los últimos años están presentando resultados comparables a los que se presentaban en los estudios previos de PSG, MRI y EEG.
- Los resultados que se obtienen hasta ahora con actigrafía no son todo lo buenos que se espera. Existe algún trabajo nuestro con buenos resultados, pero se pretende superarlos con el presente estudio.

El mayor hándicap en este tipo de estudio es la extracción de características de las señales. Hasta el momento, se han empleado *hand-crafted features* (extracción manual de características) lo que suponía un sistema costoso y que se va a intentar mejorar empleando técnicas de *Deep Learning*

## Capítulo 3

# RECONOCIMIENTO DE PATRONES

Se entiende por **Reconocimiento de Patrones** la disciplina científica que se encarga de los procesos que permitan extraer información de diferentes objetos, a fin de establecer propiedades asociadas a dichos objetos que permitan su clasificación. El sistema de diagnóstico planteado en este trabajo, se basa en una estructura de reconocimiento de patrones. En este capítulo se presentarán las diferentes técnicas empleadas en el desarrollo y las métricas empleadas para la evaluación de dicho sistema.

## 3.1 ESTRUCTURA DE UN SISTEMA DE RECONOCIMIENTO DE PATRONES

---

Para la realización del sistema de reconocimiento se realizan los siguientes procesos:

- **Adquisición de datos de entrada.**
- **Extracción de Características:** Consiste en generar las características que pueden ser útiles para el proceso de clasificación de datos. Muchas veces este proceso va precedido de un preprocesado de la señal, para corregir posibles deficiencias en los datos o bien para preparar los datos de cara a posteriores procesos en las etapas de extracción de características o clasificación. Típicamente se utiliza el término en inglés *features*.
- **Clasificación:** Consiste en la asignación del conjunto de datos de estudio a grupos o clases, basándose en las características extraídas. Esta clasificación puede ser de dos tipos: supervisada, que entrena un modelo a partir de entradas y salidas conocidas y aprendizaje no supervisado, donde únicamente se disponen de datos de entrada y una función de coste a minimizar.

### 3.1.1 EXTRACCIÓN DE CARACTERÍSTICAS

---

La extracción de características es fundamental para diseñar un buen sistema de análisis de datos. Es necesario dar una correcta interpretación a los datos en bruto para poder realizar correctamente este proceso y hallar aquellas que sean útiles para la posterior clasificación. El problema está, en que en muchas ocasiones la gran dimensionalidad del espacio de características



puede causar problemas para el desarrollo del modelo de aprendizaje automático. Existen dos enfoques para la reducción de dicho espacio [51]:

- **Selección de características:** Permite la reducción de la dimensionalidad buscando un subconjunto de características útiles para el modelo. Se pueden utilizar dos métodos:
  - *Filtering Approach:* A cada una de las funciones se le asigna una puntuación para reflejar su poder discriminativo individual. Cuando esta puntuación supera un determinado umbral se selecciona dicha característica. Constituye una herramienta simple y efectiva para eliminar características irrelevantes y reducir considerablemente la dimensionalidad.
  - *Wrapper Approach:* Depende del clasificador adoptado para el posprocesamiento, ya que requiere que la clasificación real se realice utilizando el conjunto completo de características. La selección de final dependerá de la influencia que tengan dichas características en el resultado final de la clasificación.
- **Proyección subespacial:** Las nuevas características se utilizan para crear combinaciones lineales de las originales

En los modelos de aprendizaje automático, la extracción de características se realizaba típicamente mediante el uso de diversos algoritmos que permitían extraer diferentes propiedades de las señales. Actualmente, gracias al avance a nivel de hardware, este proceso de extracción de características se puede realizar mediante redes de *Deep Learning* de forma automática.

### 3.1.2 CLASIFICACIÓN SUPERVISADA

---

Su funcionamiento se basa en el conocimiento a priori de modelos ya clasificados. En el marco de nuestro problema, conocemos a priori el diagnóstico de cada uno de los pacientes. Existen diversas técnicas de clasificación supervisada. En nuestro caso, hemos empleado tres algoritmos de clasificación: algoritmo **K-NN** (*K-Nearest Neighbour*), clasificación en **árbol** y **Support Vector Machine (SVM)**

#### 3.1.2.1 CLASIFICADORES KNN

Estos clasificadores constituyen un método de clasificación supervisada cuyo algoritmo de funcionamiento se fundamenta en que cada suceso nuevo se va a clasificar en la clase más frecuente a la que pertenecen sus vecinos más cercanos. Este método de clasificación constituye un método no paramétrico que estima la probabilidad a posteriori de que un elemento pertenezca a una clase concreta. El proceso de decisión consiste en:

- Primero, se calcula la distancia entre el elemento que queremos clasificar y el resto del conjunto de elementos del entrenamiento.
- Después se seleccionan los “k” elementos más cercanos, es decir aquellos que en base a la función utilizada para calcular la distancia se encuentren a menor distancia del elemento que queremos clasificar.

- Por último, entre los elementos que hemos encontrado que cumplen las condiciones de menor distancia, los de la clase dominante decidirán su clasificación final.

En la figura 3.1 se ve un ejemplo de funcionamiento del algoritmo tomando diferentes valores para  $k$ . Si tenemos en cuenta los 3 vecinos más cercanos, clasificaría el elemento como de la clase B, en cambio tomando los 6 vecinos más cercanos el elemento sería clasificado como de clase A.

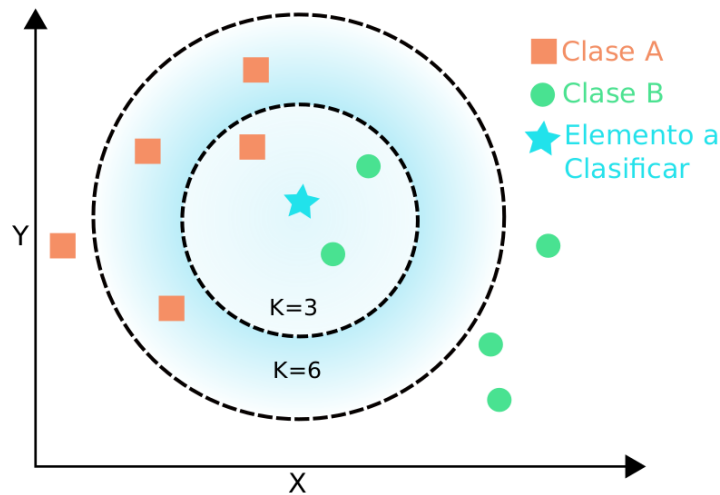


FIGURA 3.1: Ejemplo de funcionamiento del algoritmo K-NN

### 3.1.2.2 CLASIFICADORES EN ÁRBOL DE DECISIÓN

Consisten en un modelo predictivo no paramétrico que permite decidir qué clasificación es la más adecuada, a partir de aplicar un conjunto de reglas sobre diferentes ejemplos. Su principal objetivo es predecir el valor de una variable destino en función de las diferentes variables de entrada.

Los árboles de decisión se componen principalmente de dos tipos de nodos:

- **Nodos de decisión o raíz:** Esta asociado a cada uno de los atributos a tener en cuenta para la decisión de la clasificación y tiene varias ramas que salen de él, que se corresponden con sus posibles valores.
- **Nodos respuesta:** Están asociados a la clasificación que se quiere proporcionar y devuelven la decisión respecto al ejemplo de entrada.

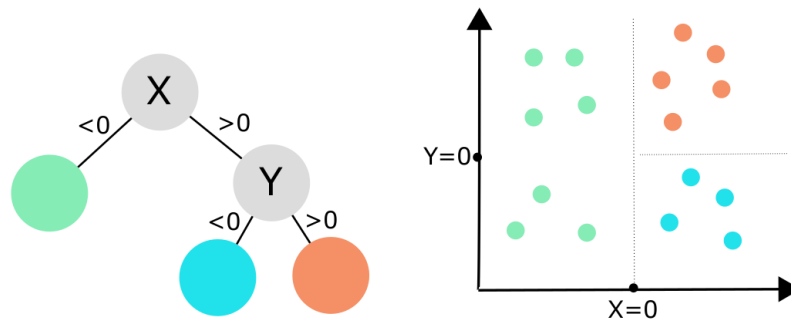


FIGURA 3.2: Ejemplo de algoritmo de árbol de decisión

### 3.1.2.3 CLASIFICADORES SVM

Las máquinas de soporte vectorial, máquinas de vectores de soporte o máquinas de vector soporte (*Support Vector Machines, SVMs*) comprenden un conjunto de algoritmos de aprendizaje supervisado, dentro de la categoría de los clasificadores lineales, aunque existen variaciones que permiten la clasificación no lineal.

Su funcionamiento se basa principalmente en la utilización de separadores lineales o **hiperplanos** de dimensionalidad muy alta. Como podemos ver en la figura 3.3 la idea principal es seleccionar un hiperplano que equidiste de los ejemplos más cercanos de cada clase, para conseguir el margen máximo a cada lado del hiperplano. A la hora de definir el hiperplano, sólo se tienen en cuenta los ejemplos de entrenamiento de cada clase que caen justo en la frontera de dichos márgenes. Estos ejemplos serán los que definiremos como **vectores de soporte**. [52]

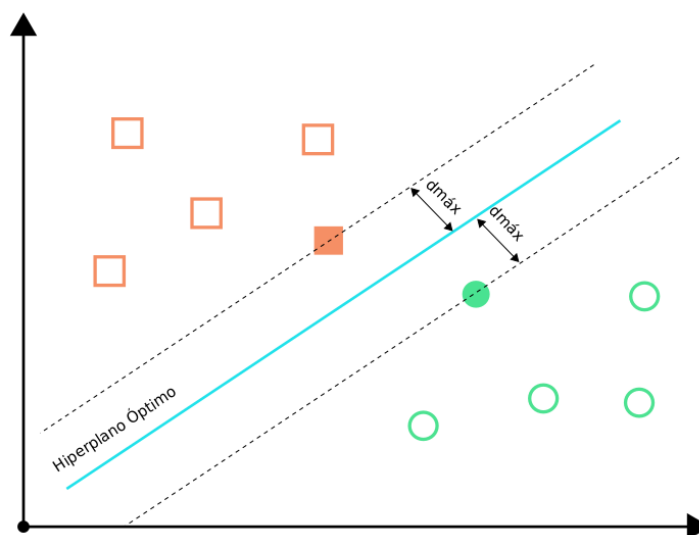


FIGURA 3.3: Obtención del hiperplano de Margen Máximo en un SVM lineal

El hiperplano separador de margen máximo ha demostrado tener una buena capacidad de

generalización, evitando en gran medida el problema del sobreajuste a los ejemplos de entrenamiento. No obstante, existe variaciones de este clasificador en las cuales se hace uso de las denominadas *funciones kernel*. Estas funciones permiten la transformación del espacio de características de que disponemos a otro equivalente, lo que nos permitirá realizar tareas de clasificación en aquellos problemas no separables linealmente. En nuestro caso, hemos empleado dos clasificadores con función kernel:

- **Clasificadores SVM<sub>rbf</sub>** : Utilizan un kernel de base radial gaussiana. El hiperplano se aproxima según el cuadrado de la distancia euclídea:

$$G(\mathbf{x}_j, \mathbf{x}_k) = e^{-\gamma \|\mathbf{x}_j - \mathbf{x}_k\|^2} \quad (3.1)$$

- **Clasificadores SVM<sub>q</sub>** : Representa los vectores de soporte en un espacio de características cuyo hiperplano es construido sobre polinomios de las variables originales:

$$G(\mathbf{x}_j, \mathbf{x}_k) = (1 + \mathbf{x}'_j \mathbf{x}_k)^q \quad (3.2)$$

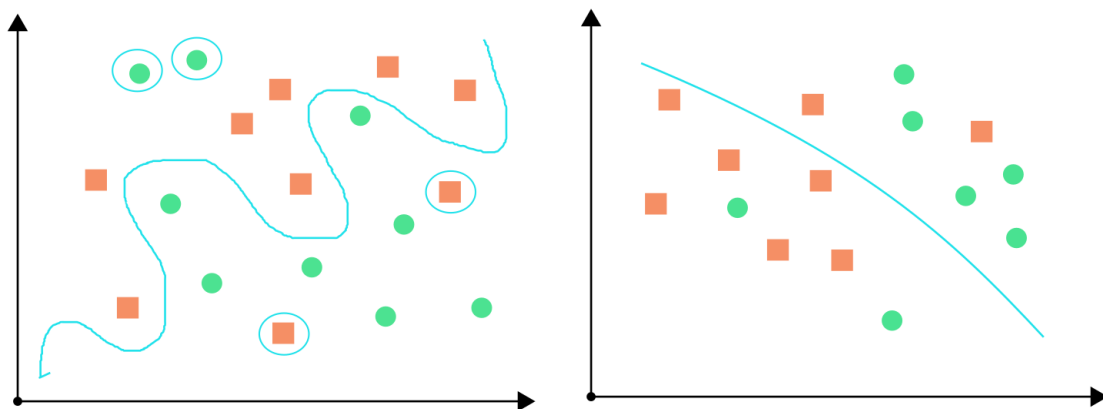


FIGURA 3.4: Ejemplo de clasificador SVM con kernel RBF de  $\gamma = 100$  y con kernel polinómico de orden  $q=2$

Los modelos basados en SVMs constituyen sistemas de clasificación muy robustos en cuanto a la generalización.

### 3.2 TÉCNICAS PARA EVALUAR EL RENDIMIENTO DEL SISTEMA

Cuando se diseña un sistema de reconocimiento de patrones y clasificación es muy importante evaluar el funcionamiento de dicho sistema. A lo largo de la siguiente sección hablaremos de las técnicas que hemos utilizado en este trabajo para evaluar el sistema de diagnóstico propuesto.

### 3.2.1 VALIDACIÓN CRUZADA

Se utiliza en el análisis estadístico para evaluar los resultados y garantizar que estos son independientes de la partición de entrenamiento y test empleada. Su funcionamiento se basa principalmente en repetir el análisis variando los conjuntos y calcular la media aritmética de los resultados obtenidos para las diferentes particiones. Existen diferentes tipos en función de los conjuntos elegidos para la evaluación:

- Validación cruzada con k-folds:** En esta técnica los datos de muestra se dividen en  $K$  subconjuntos. Uno de los subconjuntos se utiliza como datos de test y el resto ( $K-1$ ) como datos de entrenamiento. El proceso de validación cruzada se repite durante  $k$  iteraciones, evaluando cada uno de los subconjuntos de test disponibles. Este método evalúa todos los datos, pero requiere de una mayor carga computacional. evaluarlas.

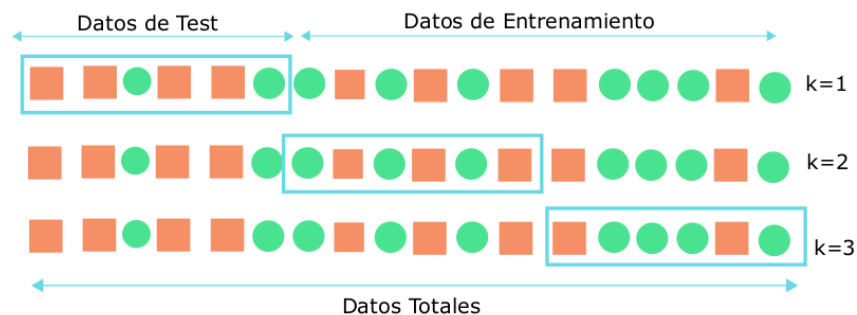


FIGURA 3.5: Ejemplo de funcionamiento de Validación con K-Folds para 3 iteraciones

- Validación cruzada aleatoria:** Consiste en dividir aleatoriamente el conjunto de datos de entrenamiento y el conjunto de datos de prueba. Con este método, hacemos que los subconjuntos de test sean independientes de las iteraciones, a cambio de que algunas muestras puedan ser bien sobre evaluadas o, por el contrario, no evaluarlas.

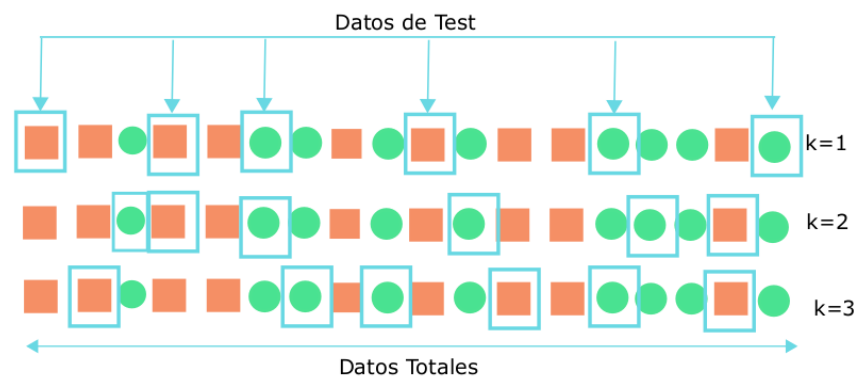


FIGURA 3.6: Ejemplo de funcionamiento de Validación Cruzada aleatoria para 3 iteraciones

- **Validación *Leave One Out*:** Los datos se separan de forma que para cada iteración solo tenemos una muestra de test. En este tipo de validación se ajusta muy bien el error, pero el coste computacional es muy elevado debido a que tenemos tantas iteraciones como muestras de las que dispongamos.

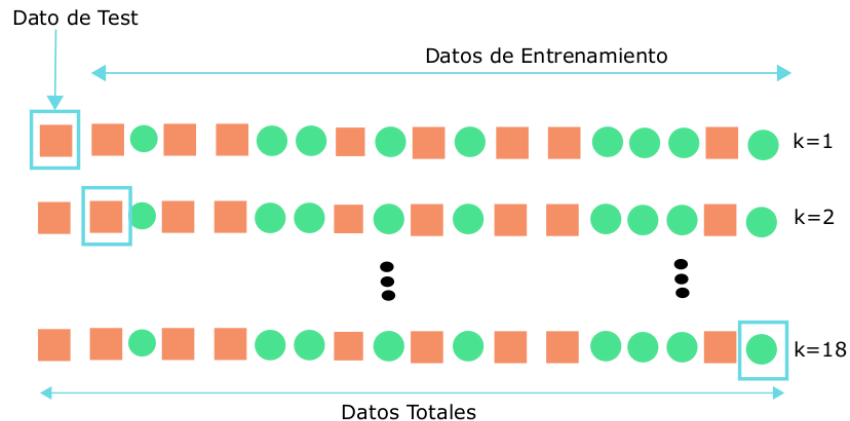


FIGURA 3.7: Ejemplo de funcionamiento de Validación Leave One Out para un conjunto de 18 muestras

### 3.2.2 MÉTRICAS PARA EVALUAR LA VALIDEZ DE LOS MODELOS DE CLASIFICACIÓN

Existen diferentes parámetros que permiten evaluar como se comporta un sistema de clasificación. Algunos de ellos son:

- **Exactitud:** Se refiere al porcentaje de acierto global del clasificador.
- **Sensibilidad:** Nos da la probabilidad de que si la observación se refiere a un paciente diagnosticado con la patología, el modelo lo clasifique como tal.
- **Especificidad:** Es la probabilidad de que si la muestra a clasificar pertenece a un paciente sano, el modelo la clasifique correctamente.
- **Precisión:** También llamado valor de predicción positiva, es la probabilidad de que, dada una predicción positiva, la realidad sea positiva también.
- **Valor de Predicción Negativa:** Nos da la probabilidad de que, dada una predicción negativa, la realidad sea también negativa.

Todos estos valores pueden ser representados mediante la matriz de confusión, muy utilizada para evaluar clasificadores. En la tabla 3.1 vemos un ejemplo de como seria la estructura de una matriz de confusión

	Predicción Clase A	Predicción Clase B	
Clase A	Verdaderos Positivos	Falsos Negativos	Sensibilidad
Clase B	Falsos Positivos	Verdaderos Negativos	Especificidad
	Precisión	Valor de Predicción Negatiiva	<b>Exactitud</b>

TABLA 3.1: Estructura de la Matriz de confusión

Cada columna de la matriz nos indica el número de predicciones para cada clase realizadas por el modelo, y cada fila los valores reales por cada clase. Los datos quedan divididos en cuatro subtipos:

- **Verdaderos Positivos (VP):** Numero de predicciones correctas de la clase A.
- **Falsos Negativos (FN):** Predicciones negativas cuando el valor en verdad es positivo.
- **Falsos Positivos (FP)** Predicciones positivas cuando el valor en verdad es negativo.
- **Verdaderos Negativos (VN):** Numero de predicciones correctas de la clase B.

La matriz de confusión nos da una idea de cómo está clasificando nuestro modelo, permitiéndonos comprobar si nuestro modelo está confundiendo entre clases, y en qué medida.

### 3.2.3 CURVA ROC

Un parámetro a tener en cuenta a la hora de evaluar la bondad de una prueba diagnóstica es la **Curva ROC (Receiver Operating Characteristic)**, que permite representar gráficamente la sensibilidad frente a la especificidad para un clasificador binario. Típicamente, este parámetro analiza el valor del área (*AUC*) bajo dicha curva.

Este parámetro nos permite evaluar el rendimiento del modelo en todos los umbrales de clasificación. En la figura 3.8 podemos ver un esquema genérico de la Curva ROC.

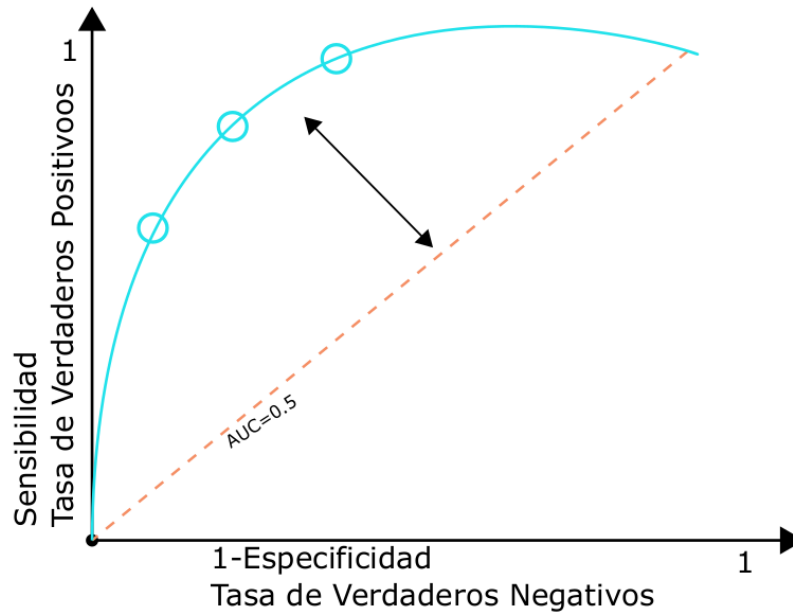


FIGURA 3.8: Curva ROC típica

Cada punto de la curva ROC (círculos negros) corresponde a un posible punto de corte del test diagnóstico, y nos informa de su respectiva sensibilidad (eje Y) y 1-especificidad (eje X). Ambos parámetros toman valores del 0 al 1 (0% a 100%). La línea trazada desde el punto 0,0 al punto 1,1 recibe el nombre de diagonal de referencia, o línea de no-discriminación, y se corresponde con una AUC de 0.5, lo que indicaría que el test de diagnóstico realizado no es capaz de discriminar entre los pacientes de ambas clases. El AUC proporciona una medición agregada del rendimiento en todos los umbrales de clasificación posibles. Una forma de interpretar el AUC es como la probabilidad de que el modelo clasifique un ejemplo positivo aleatorio más alto que un ejemplo negativo aleatorio [53]. Su valor oscila entre 0 y 1. Entre las ventajas de este método de evaluación del modelo frente a otros, destacan:

- Es una medición invariable frente a la escala.
- Es invariable respecto al umbral de clasificación

No obstante, siempre debemos estudiar el problema al que nos enfrentamos para considerar dicha métrica frente a otras, ya que según el tipo de problema puede no resultar una medición eficiente.



## Capítulo 4

# DEEP LEARNING

Las técnicas de aprendizaje automático (*Machine Learning*) constituyen una rama de la inteligencia artificial cuyo objetivo principal consiste en la creación de sistemas que sean capaces de extraer generalidades a partir de ejemplos.

Para el desarrollo de nuestro trabajo, vamos a emplear un conjunto específico de algoritmos de aprendizaje automático conocidos bajo el nombre de *Deep Learning (DL)* y que detallaremos a lo largo del presente capítulo.

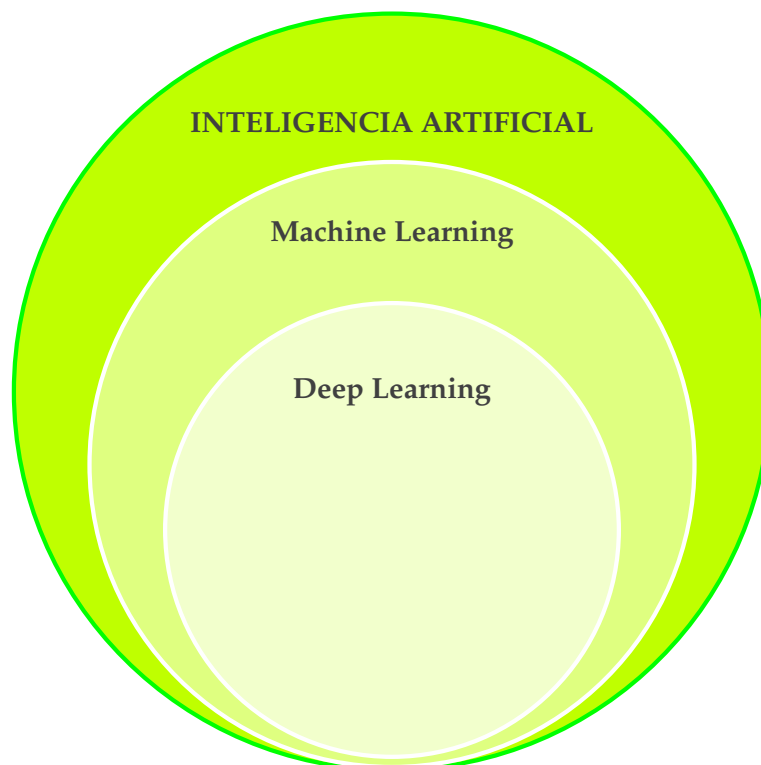


FIGURA 4.1: Jerarquía de la inteligencia artificial

## 4.1 INTRODUCCIÓN AL DEEP LEARNING

Se entiende como **Deep Learning** aquel conjunto de algoritmos de *Machine Learning* que permiten realizar una abstracción de alto nivel sobre un conjunto de datos, sin necesidad de la intervención humana [54]. Esto es posible gracias a la utilización de modelos jerárquicos que permitirán procesar los datos con un enfoque no lineal. Estos modelos se crean a partir de arquitecturas de redes neuronales artificiales.

Las **redes neuronales artificiales (NN)** constituyen una arquitectura computacional cuyo funcionamiento se basa en la interconexión en diferentes niveles de un gran conjunto de unidades de procesamiento denominadas **neuronas artificiales**, de forma aproximadamente análoga a lo que sería el comportamiento de las neuronas en el cerebro. [55]. Estas neuronas artificiales constituyen una unidad de cálculo cuyo resultado se obtiene a partir de la suma ponderada de las entradas y una función de activación, según el esquema de la figura 4.2 :

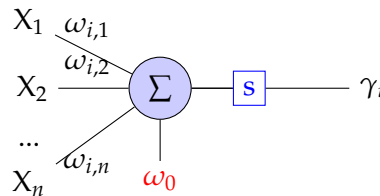


FIGURA 4.2: Esquema de funcionamiento de una neurona artificial

Podríamos expresarlo matemáticamente como:

$$\gamma_i = s\left(\sum \omega_{ij}, X_j - \omega_0\right) \quad (4.1)$$

Dónde  $\omega_{ij}$  serían los pesos de las conexiones entre las entradas y la neurona, equivalentes a las conexiones sinápticas de una neurona real,  $X_j$  serían las diferentes entradas o estímulos,  $\gamma_i$  la salida y  $s$  la función de activación. Entre las funciones de activación más comunes podemos encontrar la función escalón o la función sigmoidea. Su funcionamiento consiste en evaluar las entradas ponderadas por sus respectivos pesos combinándolos con la función de activación de la red para obtener un potencial. Si este potencial supera el umbral prefijado, se activará la salida única. [56][57][58][59]

El algoritmo de funcionamiento de las estructuras de *Deep Learning* se encarga de ajustar sus pesos (parámetros libres) para alcanzar los requerimientos de desempeño del problema basado en muestras representativas. Para llevar a cabo este proceso se crea un sistema adaptativo a través de redes neuronales multicapa, que son redes formadas por más de una capa oculta de interconexiones neuronales y cuya capacidad de aprendizaje constituye su principal característica. Este proceso de aprendizaje, también denominado entrenamiento de la red permite encontrar un modelo de ajuste para los datos a partir de un conjunto de patrones [57]. Puede ser de tres tipos:

- **Aprendizaje Supervisado** : Consiste en entrenar a partir de un conjunto de datos o patrones de entrada y salida conocidos. El objetivo del algoritmo de aprendizaje es ajustar los

pesos  $w_{ij}$  de la red para conseguir a la salida una solución lo más parecida posible a la verdadera. Se emplea sobretodo en tareas de clasificación y regresión.

- **Aprendizaje no Supervisado** : Se ajusta el modelo a partir de las observaciones a la entrada. Su principal diferencia con respecto al aprendizaje supervisado es la falta de conocimiento *a priori*. Los datos a la entrada son principalmente interpretados como un conjunto de variables aleatorias. En el contexto del aprendizaje profundo, generalmente el objetivo es aprender la distribución de probabilidad completa que generó un conjunto de datos, ya sea explícitamente, como en la estimación de la densidad, o implícitamente, para tareas como síntesis o eliminación de ruido. Existen algoritmos que permiten realizar otras funciones; como la agrupación en clusters, que consiste en dividir el conjunto de datos en grupos de ejemplos similares. [60]
- **Aprendizaje por Refuerzo** : Este tipo de algoritmos se basa en el aprendizaje a partir del entorno. Este se corresponde con el conjunto de datos de entrada. A partir de estos datos, el algoritmo genera algún tipo de realimentación o recompensa acerca de la idoneidad de la respuesta dada. El entorno, es formulado generalmente como un proceso de decisión de Markov (MDP) [61, 62]

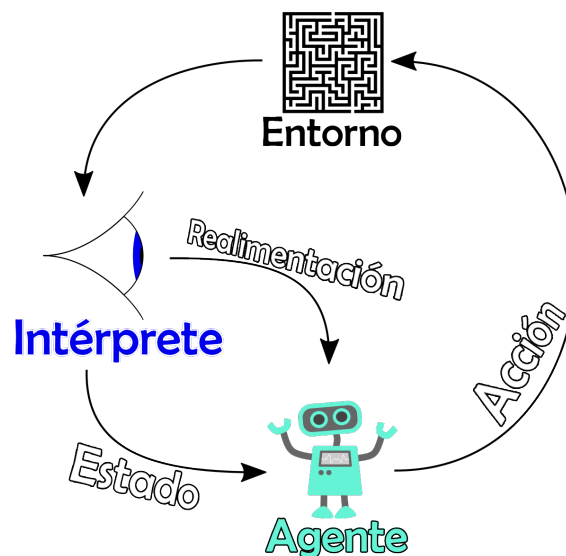


FIGURA 4.3: Gráfico de funcionamiento de un sistema de aprendizaje por refuerzo

Nos centraremos en las redes sobre las que podamos realizar entrenamiento **supervisado**.

#### 4.1.1 TIPOS DE REDES PARA APRENDIZAJE SUPERVISADO

En función del algoritmo que utilizan para llevar a cabo las tareas de aprendizaje es posible distinguir varios tipos de redes: **Red Neuronal Prealimentada (FNN)**, red neuronal que se enmarca dentro del *Machine Learning* y dos de las estructuras de red neuronal más típicas de *Deep Learning*: **Red Neuronal Recurrente (RNN)** y **Red Neuronal Convolutiva (CNN)**.

## 4.1.1.1 FEEDFOWARD NEURAL NETWORK (FNN)

La red neuronal prealimentada (feed-forward), también denominada **perceptrón multicapa** (*MLP*) o red de avance profundo constituye la estructura típica de los modelos de aprendizaje y se toma como base para el desarrollo del resto de redes. Su objetivo principal consiste en aproximar una función determinada  $f(x)$ . Para ello, define una asignación y aprende el valor de los parámetros que resultan en la mejor aproximación de la función. [60]

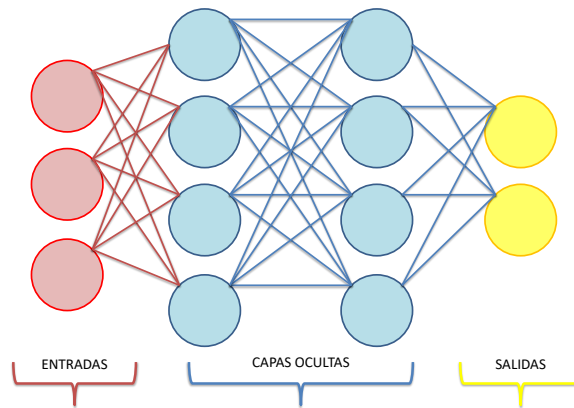


FIGURA 4.4: Estructura básica del Perceptrón Multicapa

Como podemos observar en la figura, cada neurona perteneciente a una capa tiene conexiones dirigidas a las neuronas de la capa siguiente. Existe una gran variedad de técnicas de aprendizaje que se emplean para redes multicapa, pero la más utilizada se conoce como **propagación hacia atrás**. Esta técnica calcula el gradiente de la función de error empleando un ciclo de propagación:

$$\frac{\partial E}{\partial \mathbf{W}^{(l)}} = \frac{\partial E}{\partial \mathbf{a}^{(L)}} \frac{\partial \mathbf{a}^{(L)}}{\partial \mathbf{a}^{(L-1)}} \cdots \frac{\partial \mathbf{a}^{(l+2)}}{\partial \mathbf{a}^{(l+1)}} \frac{\partial \mathbf{a}^{(l+1)}}{\partial \mathbf{a}^{(L)}} \frac{\partial \mathbf{a}^{(l)}}{\partial \mathbf{z}^{(l)}} \frac{\partial \mathbf{z}^{(l)}}{\partial \mathbf{W}^{(L)}} \quad (4.2)$$

Donde  $E$  es la función de error,  $\mathbf{W}^{(l)}$ ,  $\mathbf{z}^{(l)}$  y  $\mathbf{a}^{(l)}$  los pesos y las funciones de pre-activación y activación de la de la capa  $l$  de la red respectivamente y  $\mathbf{a}^{(L)}$  la salida. Tras aplicar un patrón a la entrada de la red, este se propaga desde la primera capa a través de las capas siguientes de la red, hasta obtener la salida. Esta señal de salida se compara con la salida deseada y se calcula una señal de error para cada una de las salidas. La señal de error de cada salida es propagada hacia atrás a todas las neuronas de las capas ocultas implicadas en el cálculo de dicha salida, recibiendo cada neurona implicada una señal de error proporcional a su contribución relativa al error total. Esto permite a las neuronas de las capas intermedias reconocer características del espacio total de entrada, que podrán identificar cuando a la entrada se presente una señal arbitraria ruidosa. [63]

#### 4.1.1.2 RECURRENT NEURAL NETWORK (RNN)

Las redes neuronales recurrentes son un tipo de redes neuronales artificiales basadas en el perceptrón multicapa pero cuyas conexiones entre los nodos forman un grafo dirigido a lo largo de una secuencia. Las RNN pueden usar su estado interno (memoria) para procesar secuencias de entradas, lo que hace de ellas una opción de red aplicable a tareas como el reconocimiento de voz o el reconocimiento de patrones de escritura manuscrita.

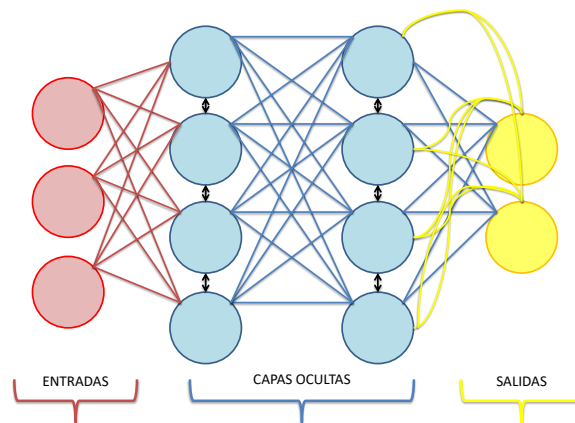


FIGURA 4.5: Estructura básica del Perceptrón Multicapa para una red recurrente

Como podemos observar en la figura 4.5, existen caminos de retroalimentación entre todos los elementos de la red. Una sola neurona está conectada a las neuronas posteriores en la siguiente capa, las neuronas pasadas de la capa anterior y a ella misma a través de vectores de pesos variables que sufren alteraciones en cada época del entrenamiento.

#### 4.1.1.3 CONVOLUTIONAL NEURAL NETWORK (CNN)

Una red neuronal convolucional (*ConvNet*) es un tipo de red neuronal artificial donde las neuronas corresponden a campos receptivos simulando el comportamiento de las neuronas en la corteza visual primaria (V1) de un cerebro biológico. Aunque este tipo de arquitectura está pensada principalmente para que sus entradas sean imágenes, permite procesar datos que vienen en forma de múltiples arreglos, como 1D para señales y secuencias y 3D para vídeo o imágenes volumétricas.

Este tipo de redes son muy similares a las redes neuronales convencionales, su principal diferencia radica en que realiza el aprendizaje de características utilizando la operación de convolución. Se emplean diferentes capas para crear la red, entre las que destacan:

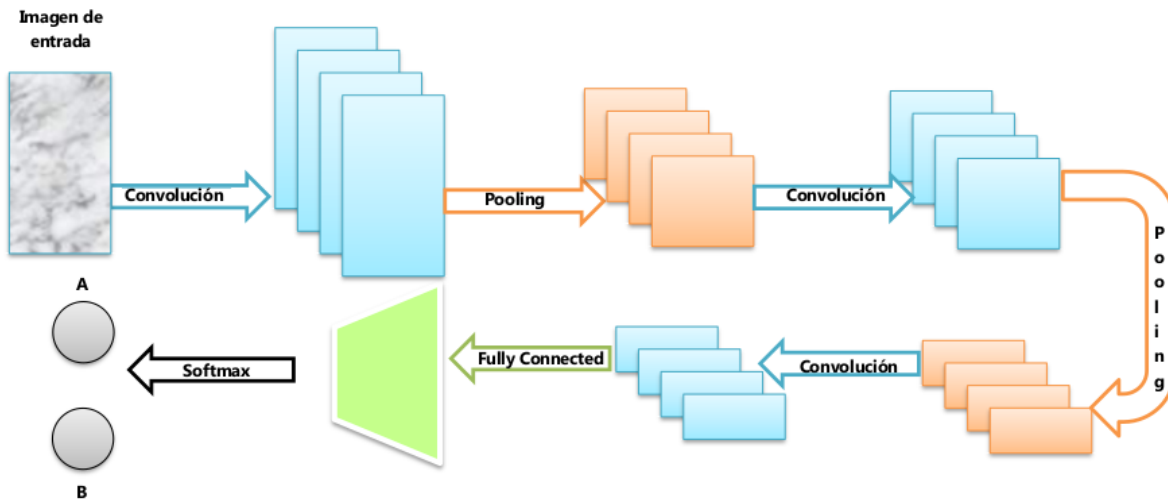


FIGURA 4.6: Estructura básica de un red Convolucional

- Capa Convolutiva** : La convolución consiste en realizar operaciones de suma y multiplicación entre la señal de entrada a dicha capa y los  $n$  filtros o *kernels* para generar un mapa de características correspondientes a la posible ubicación del filtro en la imagen original. Una de las principales ventajas de esta operación es que como un mismo filtro nos sirve para extraer la misma característica en cualquier punto de la red, es factible reducir el número de conexiones y el número de parámetros a entrenar en comparación con el perceptrón multicapa totalmente conectado.

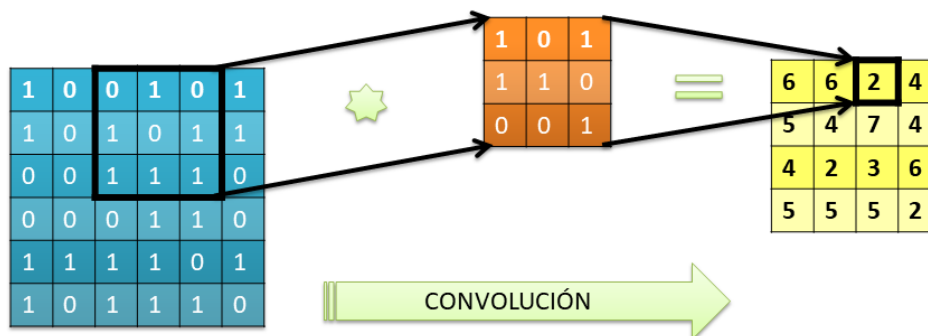


FIGURA 4.7: Funcionamiento de la operación de convolución

Una vez realizada la convolución se suelen emplear capas de activación para aumentar las propiedades no lineales de la función de decisión y de la red general. La más utilizada es la función **ReLU** conocida también como unidad lineal rectificadora, debido a que permite entrenar la red neuronal varias veces más rápido sin una penalización significativa a la precisión de la generalización sin cambiar el tamaño de su entrada. Permite una capacita-

ción mucho más rápida cuanto más grande sea la red. La capa ReLU realiza una operación de umbral [3, 64, 65] para cada elemento de la entrada, donde cualquier valor menor que cero se establece en cero.

$$f(x) = \begin{cases} x & x \geq 0 \\ 0 & x < 0 \end{cases} \quad (4.3)$$

- **Capa de Pooling o Reducción** : El objetivo de esta capa es la reducción del tamaño espacial de los datos, para quedarse solo con aquellas características que sean más comunes. Esta reducción de parámetros se realiza mediante la extracción de estadísticas como el promedio o el máximo de una región fija del mapa de características.[66] [67]

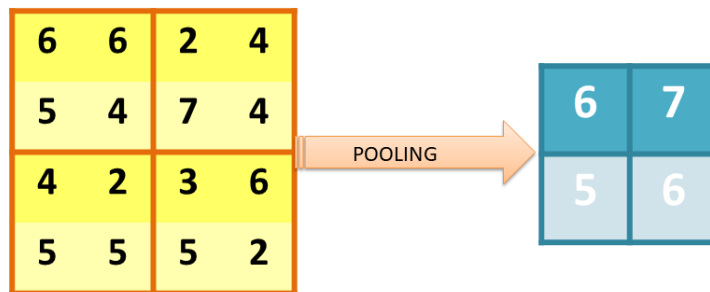


FIGURA 4.8: Funcionamiento de la operación de *Max Pooling*

La reducción de características hace que el sistema mejore su compatibilidad favoreciendo la obtención de una representación invariable a pequeñas traslaciones en la entrada y previene de un posible sobreajuste. Su principal desventaja es que al realizar esta operación, el sistema se vuelve ligeramente menos preciso.

- **Capa Fully Connected o totalmente conectada** : Su estructura es similar a las redes neuronales convencionales, pero toma cada pixel en la imagen como una neurona independiente. Esta capa combina todas las características (información local) aprendidas por las capas anteriores en la imagen para identificar los patrones más grandes. Para problemas de clasificación, la última capa completamente conectada combina las características para clasificar las imágenes. Además, suele ir acompañada de una capa **SoftMax** o función exponencial normalizada para obtener las probabilidades de cada clase posible.

## 4.2 DEEP LEARNING VS MACHINE LEARNING TRADICIONAL

En las últimas décadas se ha observado un desarrollo considerable en el campo de la Inteligencia Artificial (IA), gracias en parte a la evolución y el uso de las técnicas de *Machine Learning* y un subconjunto de estas, *Deep Learning*. Aunque en algunas ocasiones se utilizan ambos términos indistintamente, existen diferencias tanto de concepto como de funcionamiento que sitúan al *Deep Learning* a la cabeza de los próximos avances en el campo de la IA.

- Los algoritmos de *Deep Learning* se centran exclusivamente en las técnicas de aprendizaje mediante **ejemplo**, mientras que el término *Machine Learning* engloba a todos aquellos algoritmos que permitan parsear datos, aprender de ellos y luego ser capaces de hacer una predicción.
- La **extracción de características** a partir de los datos de entrada se hace típicamente de forma manual en *ML* mientras que en *DL* se obtienen **directamente del entrenamiento** de la red, por lo que requerirá sistemas más complejos y equipos con una capacidad de procesamiento mayor. Es decir, nos permite realizar de forma automática este proceso a cambio de utilizar una mayor cantidad de datos para el entrenamiento.
- Mientras que para *ML* se emplean redes neuronales sencillas; como el perceptrón simple que solo tiene una capa oculta, para poder realizar los algoritmos de *DL* necesitaremos **redes neuronales multicapa**, con más de una capa oculta, llegando a existir sistemas que soporten cientos de capas ocultas. El hecho de no ver las salidas en todos los niveles, dará un grado mayor de complejidad y constituye otra de las causas de los requerimientos específicos de procesamiento.
- Otra de las grandes diferencias existentes es que para los algoritmos de *Deep Learning* la escala aumenta con los datos. Esto quiere decir que cuanto mayor sea la cantidad de datos de la que disponemos, mayor será el rendimiento del sistema. En el aprendizaje automático tradicional, el aprendizaje que se realiza es superficial y se produce convergencia, a cierto nivel de rendimiento el sistema no mejora por muchos datos que se le agreguen.
- El *Deep Learning* es capaz de soportar y trabajar con big data, lo que supone una gran ventaja con respecto al *Machine Learning*

#### 4.2.1 EJEMPLOS DE APLICACIONES DEL DEEP LEARNING A SEÑALES BIOMÉDICAS

Actualmente, se están empleando este tipo de algoritmos para la creación de sistemas como traductores inteligentes, reconocimiento facial o visión computacional; entre otros, pero es su capacidad para obtener conocimiento de enormes volúmenes de datos lo que suscita interés en el campo de la investigación científica.

En el contexto de nuestro problema existen estudios que han profundizado en el uso del *Deep Learning* para el análisis del sueño. Sathyanarayana et. al [19] se sirven de estos algoritmos para el estudio de la influencia de la actividad física en el sueño, con resultados interesantes. Lo mismo ocurre en el estudio de Tsinalis et. al [68] donde emplean redes neuronales convolucionales para la detección automática de las fases del sueño.

Con los resultados optimistas que se observan en estos estudios, parece evidente optar por esta vía para continuar con la investigación iniciada por Martin et. al [2]. En ella, se utilizaban técnicas de *Machine Learning* para el procesado de las señales de actigrafía. Se comprobó que la mayor dificultad de utilizar *Machine Learning* convencional estaba en que las características que eran objeto del análisis estadístico tenían que ser obtenidas de forma manual. El uso de *Deep Learning* resolvería este problema, ya que los algoritmos están diseñados para desarrollar criterios propios que le permitan obtener las características importantes para realizar la clasificación y tomar



decisiones empleando únicamente el conjunto de datos de entrada, que deberá ser más grande que el de un sistema convencional para que el rendimiento en el aprendizaje sea lo mejor posible.

En estudios previos a este trabajo, se planteó la utilización de redes convolucionales para señales 1D, pero no se consiguieron los resultados esperados. Es por eso que en este caso enfocamos el problema como un sistema para señales en 2D, adaptando las señales actigráficas mediante un procesado previo que nos permita explotar al máximo la versatilidad de las redes neuronales convolucionales.

## METODOLOGÍA

Como se ha mencionado anteriormente, el objetivo de este trabajo es diseñar un sistema de diagnóstico que permita distinguir entre pacientes sanos y pacientes diagnosticados con TDAH. Para ello, se tomará un registro actigráfico obtenido del estudio de la actividad diaria en niños de entre 6 y 15 años y se diseñará una red que aplique algoritmos de *Deep Learning*. En este capítulo, nos centraremos en detallar el procedimiento que se ha empleado para el desarrollo del sistema, concretamente:

- Presentación de las señales obtenidas del registro, procesamiento preliminar y adaptación para poder funcionar como datos efectivos de entrada para una red convolucional.
- Diseño de la **red convolucional** y explicación del proceso de entrenamiento de la misma.
- Adaptación de los resultados del entrenamiento y explicación de los diferentes clasificadores empleados para el análisis completo por paciente.

### 5.1 DESCRIPCIÓN DE LAS SEÑALES DE ENTRADA

---

Las señales actigráficas del registro han sido obtenidas mediante un actígrafo de muñeca (*ActiGraph GT3x*). Este actígrafo se encarga de medir la aceleración en los tres ejes cartesianos, registrando una muestra cada segundo (*frecuencia de funcionamiento de 1 Hz*).

Disponemos de información actigráfica de un total de 148 pacientes, de los cuales 73 han sido clasificados como afectados por TDAH según el *Diagnostic And Statistical Manual Of Mental Disorders (DSM-V)* [1] y a los que de ahora en adelante denominaremos **Casos**. Los 75 restantes se denominarán **Controles**.

Se va a trabajar con la media de las tres medidas de los ejes cartesianos, obteniendo así una única señal 1D por cada paciente. Esta señal 1D será procesada según el esquema de la figura 5.1

Una vez que tenemos elegidos los datos con los que vamos a trabajar, adecuamos las señales correspondientes a cada sujeto, para que sean comparables. Es decir, ajustamos el número de muestras hasta un máximo de 24 horas por cada sujeto. Teniendo en cuenta que el actígrafo con el que se realizaron las mediciones trabajaba a una frecuencia de muestreo de 1 Hz; significa que

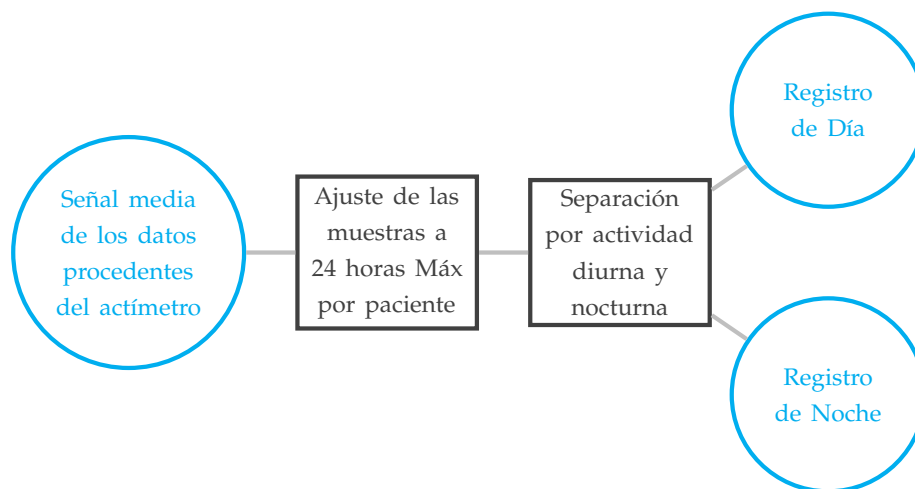


FIGURA 5.1: Etapas de la adecuación de los registros de información, previa al preprocesado

tendremos hasta un máximo de 86400 muestras para cada sujeto. Con las señales equiparables entre sí, dividimos la señal de cada paciente en dos subconjuntos diferentes en base a los periodos de actividad. Estos dos subconjuntos se corresponden con la actividad diurna y nocturna y servirán como conjuntos independientes de entrenamiento a la red, con el objetivo de identificar las diferencias entre Casos y Controles en periodos de actividad equivalente.

Para ambos periodos de actividad, realizamos una fragmentación de la señal en ventanas del mismo tamaño, de forma que cada fragmento lo tomaremos como una señal independiente. Para tomar la decisión del tamaño óptimo de ventana hay que tener en cuenta que estamos trabajando con señales de actividad. Por lo tanto, tal y como se ha demostrado en estudios anteriores [2] resulta de interés dividir las actividades en grupos de diferente duración, lo que permitirá estudiar la actividad asociada a movimientos o bien a tareas. En nuestro caso trabajaremos con tres grupos distinguibles: actividades de corta duración (60 segundos) , actividades de duración media (5 minutos) y actividades de larga duración (30 minutos). Para realizar la fragmentación se ha tomado un solapamiento del 66.67 %, obteniendo los conjuntos de señales que se detallan en la tabla 5.1.

TAMAÑO MUESTRA	MUESTRAS SOLAPADAS	CASOS		CONTROLES	
		<i>Día</i>	<i>Noche</i>	<i>Día</i>	<i>Noche</i>
60	40	217747	214192	279988	265266
300	200	43403	42708	55836	52884
1800	1200	7064	6959	9131	8661

TABLA 5.1: Ventanas iniciales para cada grupo de actividad

### 5.1.1 PREPROCESAMIENTO: ANÁLISIS DE LA POTENCIA DE LAS SEÑALES

Es necesario que las señales estén normalizadas para que podamos hallar patrones en las señales fragmentadas, ya que los valores máximos y mínimos de actividad siempre variarán en función de las diferentes actividades que esté realizando cada paciente en el momento en el que se realizó la medición. Por ello, se han ajustado los valores de cada ventana por el máximo, de tal forma que los valores de todas las ventanas estarán entre 0 y 1.

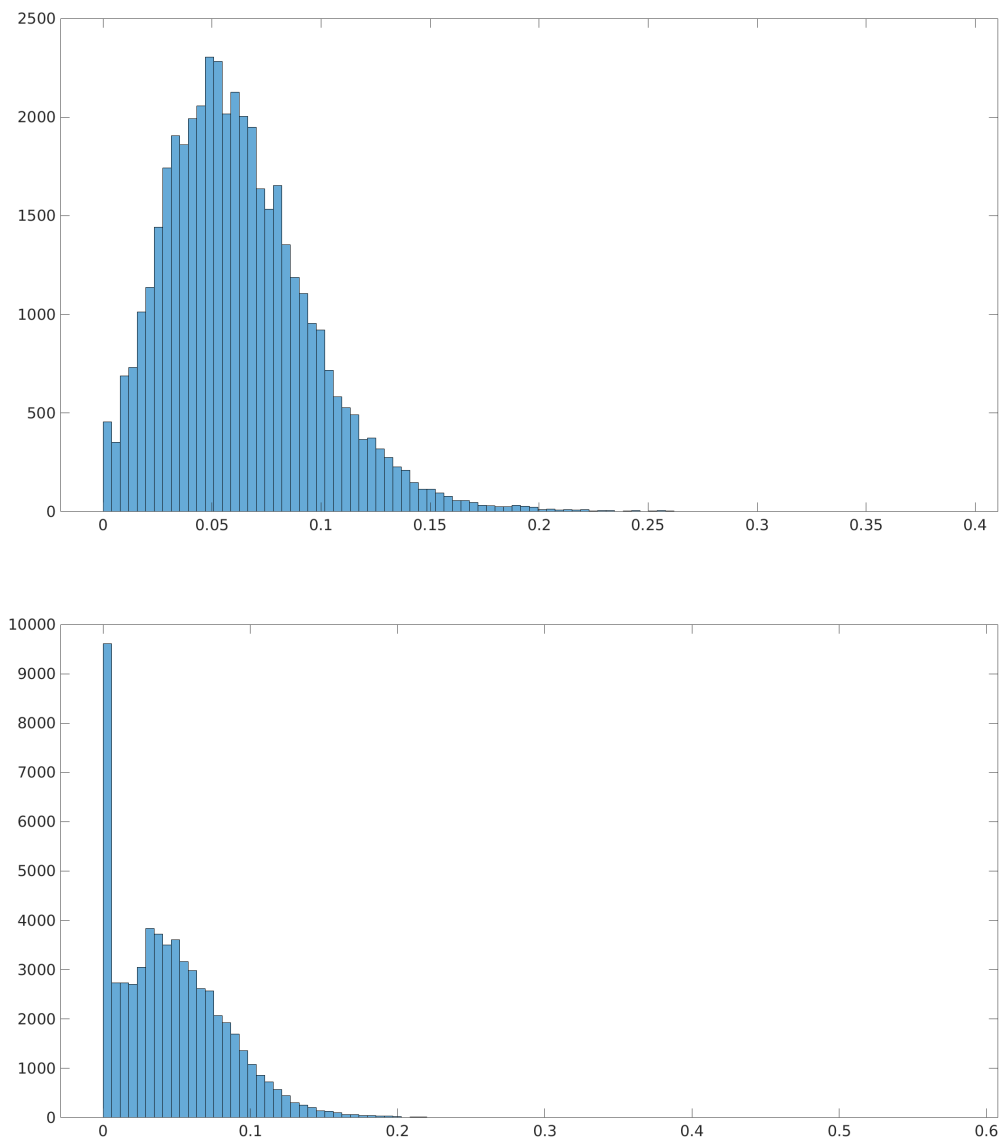


FIGURA 5.2: Histograma inicial de Potencias para ventanas de 300 segundos para *Casos* y *Controles*

Otro de los factores a tener en cuenta es que, a lo largo del día, se dan muchos periodos de

inactividad en los que el actígrafo registra valores nulos o demasiado pequeños. Estas señales de poca o nula actividad no resultan de interés en nuestro estudio, debido a que no nos van a mostrar ninguna característica representativa y pueden saturar la red de forma que los entrenamientos no resulten eficientes. Para localizar aquellas señales que nos interesan, se ha realizado un estudio de la potencia de las muestras de cada uno de los grupos de estudio y se han tomado los criterios de decisión a partir de los histogramas obtenidos. En la figura 5.2 podemos ver el histograma obtenido para el estudio de actividad diurna de duración media. Vemos en la figura que, especialmente en el caso de los controles, hay una acumulación importante de ventanas cuyo valor de potencia es nulo. Tomando el histograma para controles como referencia, ajustamos la potencia mínima a un valor de 0.022. Podemos ver en la figura 5.3 que la distribución de potencia ahora es análoga para ambos grupos. En el ejemplo, vemos el caso para señales de 300 segundos, pero operamos de la misma forma para los tres tamaños de ventana.

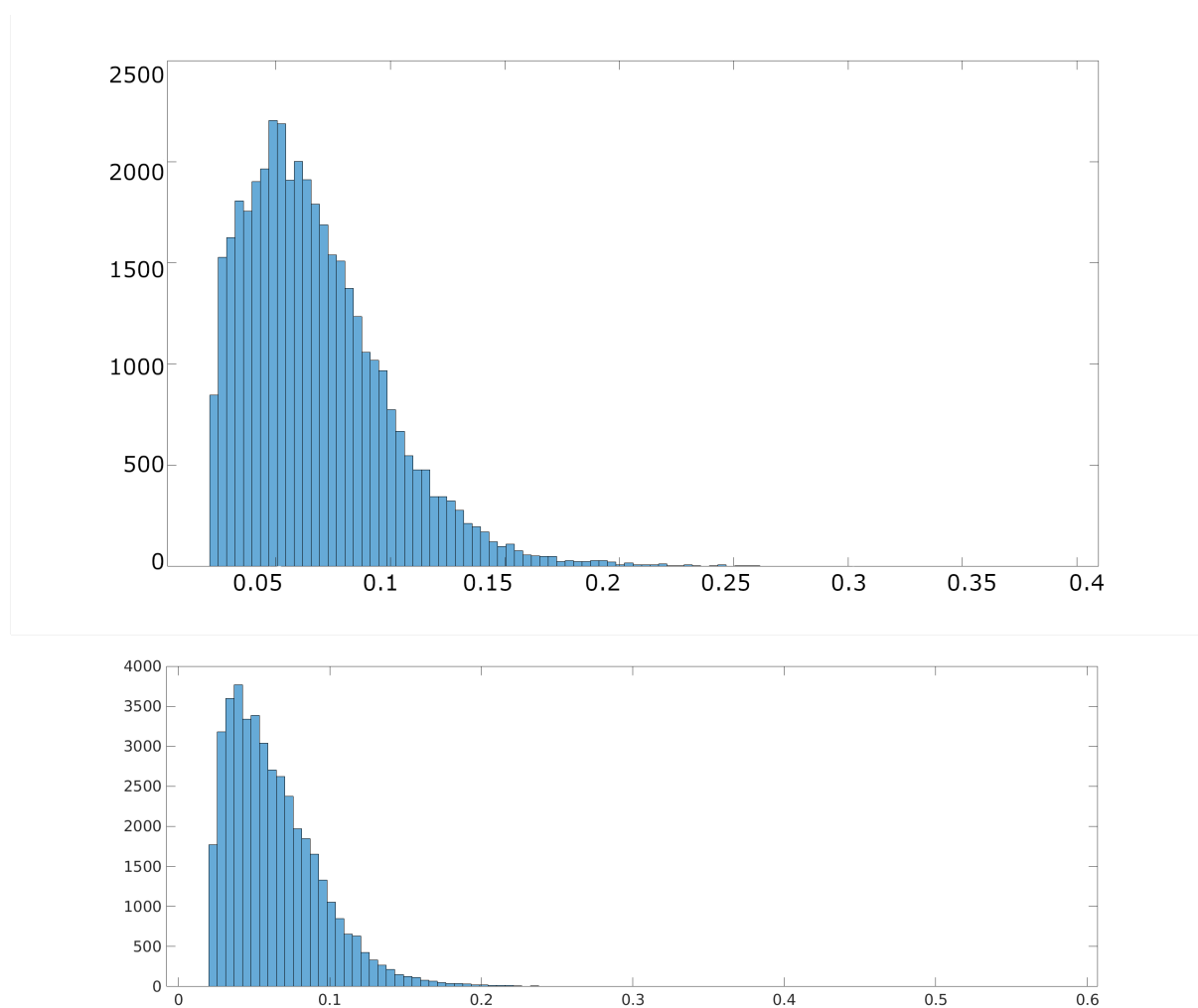


FIGURA 5.3: Histograma final de Potencias para ventanas de 300 segundos para *Casos* y *Controles*

En la tabla se recogen los conjuntos de muestras finales que tendremos para trabajar con cada subgrupo.

TAMAÑO MUESTRA	MUESTRAS SOLAPADAS	CASOS		CONTROLES	
		Día	Noche	Día	Noche
60	40	211105	42397	237301	59398
300	200	43105	22410	50689	25262
1800	1200	7032	4920	8170	5576

TABLA 5.2: Ventanas después del análisis de la potencia para cada grupo de actividad

## 5.2 ADECUACIÓN DE LOS DATOS

Como se ha mencionado anteriormente, las CNN son especialmente eficientes en la detección de patrones en imágenes. En nuestro caso, las señales actigráficas son señales 1D, por lo que necesitamos realizar un procesamiento sobre la señal que permita expresar dicha información en una representación 2D. La técnica que vamos a emplear para realizar esta representación 2D es conocida como **espectrograma**.

### 5.2.1 ESPECTROGRAMA: REPRESENTACIÓN 2D DE SEÑALES 1D

El espectrograma es una representación gráfica tridimensional del espectro de la señal a partir de tramas enventanadas de dicha señal. En ella se representa la variación de la energía del contenido frecuencial a lo largo de un eje temporal. Para la creación del espectrograma:

- Se realiza un análisis por tramas para determinar el tamaño óptimo de ventana.
- Una vez hecho este análisis y elegido el tamaño de ventana, se realiza la transformada de Fourier de los datos de cada ventana, empleando una Transformada de Fourier de Tiempo Corto (**STFT, (short time Fourier transform)**). La STFT es un método de procesamiento de señales no estacionarias. Matemáticamente, se define como:

$$\text{STFT}(x[n]) = X(m, \omega) = \sum_{n=-\infty}^{\infty} x[n]w[n-m]e^{-j\omega n} \quad (5.1)$$

En la práctica, el procedimiento para calcular las STFT consiste en dividir una señal de tiempo más larga en segmentos más cortos de igual longitud (ventanas) y luego calcular la transformada de Fourier por separado en cada segmento más corto, lo que nos permite obtener el espectro de Fourier de cada uno de ellos. Después, se realiza la gráfica de los espectros de dichos segmentos, de forma que podemos evaluar los cambios del espectro en función del tiempo. Normalmente, existe un porcentaje de solapamiento entre las ventanas, con el objetivo de evitar pasar por alto las discontinuidades que se producen en el cálculo del espectro.

- Por último, se calcula el espectrograma como la representación de la magnitud cuadrada de la señal  $X(m, \omega)$  en unidades logarítmicas:

$$\mathbf{x}(\mathbf{n})(m, \omega) = |X(m, \omega)|^2 \quad (5.2)$$

En la figura 5.4 se puede observar un ejemplo de cómo se vería un espectrograma. En este caso, se representan las variaciones de energía en una ventana de 300 segundos. Aquellas zonas cuya potencia es más elevada (zonas anaranjadas) se corresponderían con los picos de actividad.

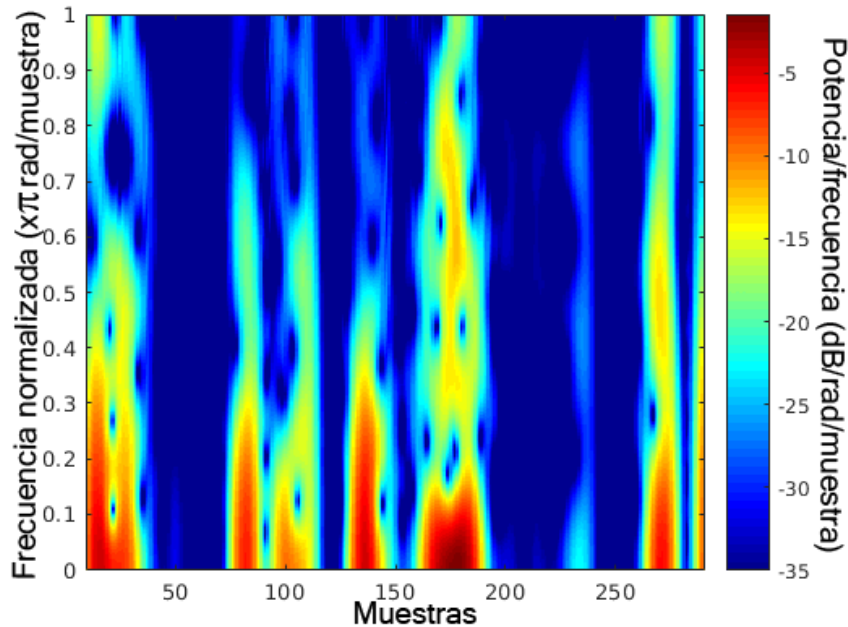


FIGURA 5.4: Espectrograma de ejemplo para ventanas de 300 segundos.

El uso de espectrogramas como señales de entrada a una red neuronal, ha demostrado ser efectivo para las señales obtenidas a través de un sensor de señales actigráficas [69] teniendo para cada señal espectrogramas diferentes, los cuales pueden ser oportunamente distinguidos. Sobre esta idea vamos a desarrollar nuestro sistema de diagnóstico, con el objetivo de encontrar las diferencias entre espectrogramas de niños diagnosticados con TDAH y sujetos sanos.

### 5.2.2 GENERACIÓN DE ESPECTROGRAMAS PARA LAS SEÑALES ACTIGRÁFICAS DEL ESTUDIO

Se pretende crear una única red convolucional en la que podamos utilizar como datos de entrada los diferentes conjuntos de actividad mencionados anteriormente y observar el comportamiento de la red en cuanto a la precisión de la clasificación. Para ello, alimentaremos la red con imágenes como señal de entrada, que deberán ser del mismo tamaño para los tres conjuntos. Para elegir el tamaño óptimo de la imagen tenemos que tener en cuenta:

- La relación entre el tiempo que duran los entrenamientos y el tamaño de la imagen. Tendremos que buscar un tamaño de imagen que nos permita visualizar correctamente los picos frecuenciales y que nos garantice una duración moderada de tiempo en los entrenamientos, especialmente en los entrenamientos para actividades de corta duración, ya que la cantidad de datos a procesar será mucho mayor.

- La relación entre los parámetros modificables para la generación del espectrograma para conseguir una buena resolución tanto frecuencial como temporal, debido a que se pretende localizar aquellos instantes de tiempo en los que se den picos de frecuencia donde la energía sea elevada.

En la creación de espectrogramas existen varios parámetros que influyen directamente en la resolución y la dimensionalidad la imagen obtenida:

- **Longitud de la ventana:** Al aumentarla, se gana en resolución frecuencial pero se pierde resolución temporal. Debemos tener siempre presente que el objetivo es construir imágenes para alimentar la red convolucional, por tanto las imágenes tendrán que tener un tamaño razonable para que la red pueda identificar las *features* (características) correctamente.

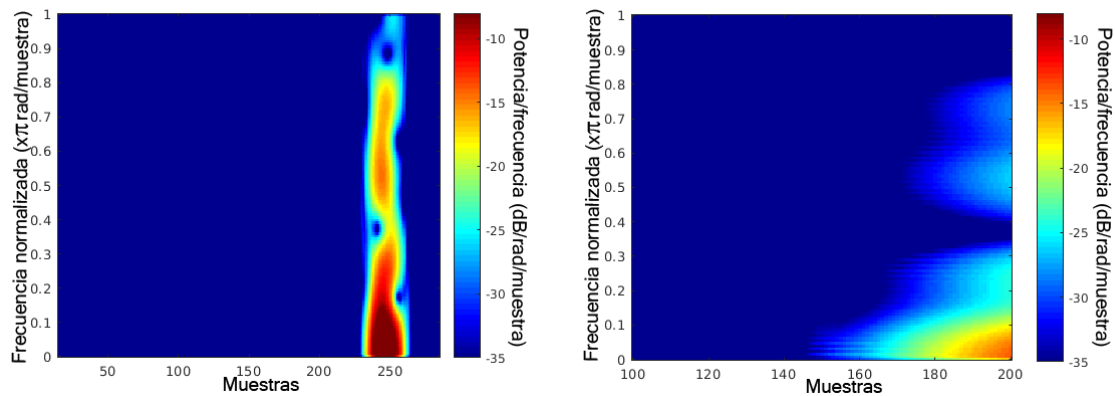


FIGURA 5.5: Espectrogramas para actividades de 300 segundos con overlap máximo y tamaño de ventana 30 y 200 respectivamente

En la figura 5.5 se aprecian las diferencias de resolución para diferentes tamaños de ventana. El efecto de pérdida de resolución temporal se ve mucho más claro a la hora de generar la imagen que serviría como entrada a la CNN, ya que, como se muestra en la figura 5.6 se pierde dimensionalidad pasando de tener una imagen de 129x271 píxeles a una imagen de 129x101 píxeles.

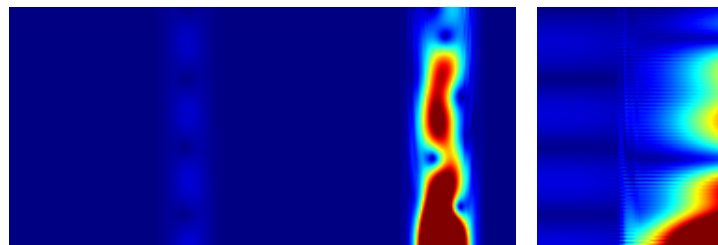


FIGURA 5.6: Imágenes de entrenamiento para actividades de 300 segundos con overlap máximo y tamaño de ventana 30 y 200 respectivamente

- **Tamaño de overlap:** Se corresponde con el número de muestras que se solapan en una ventana. Cuanto mayor sea el porcentaje de muestras solapadas, mayor será la resolución



temporal del espectrograma, pero también aumentará el tamaño de la imagen de entrada para la CNN. Podemos ver esta pérdida de resolución en la figura 5.7, donde visualizamos espectrogramas con overlap del 50 % y del 67 %.

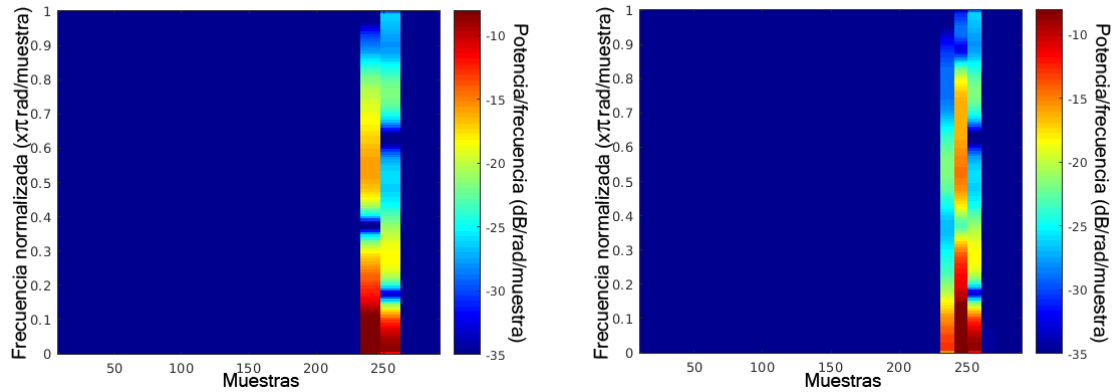


FIGURA 5.7: Imágenes de entrenamiento para actividades de 300 segundos con overlap del 50 % y del 67 % respectivamente.

- Tamaño de la DFT:** Nos permite cambiar el número de puntos que utilizamos para hacer la Transformada Discreta de Fourier (DFT), de forma que al tomar más puntos dotamos de una mayor finura a la imagen. Podemos ver en la figura 5.8 que esta mejora hace que necesitemos imágenes de entrada más grandes; generando imágenes de 65x101 píxeles en el caso de utilizar una DFT de 128 puntos frente a imágenes de 257x101 píxeles si utilizamos una DFT de 512 puntos. Tendremos que ajustar este parámetro de forma que la finura de la imagen sea adecuada y obtener imágenes de un tamaño razonable sin ralentizar excesivamente el tiempo de entrenamiento de la red.

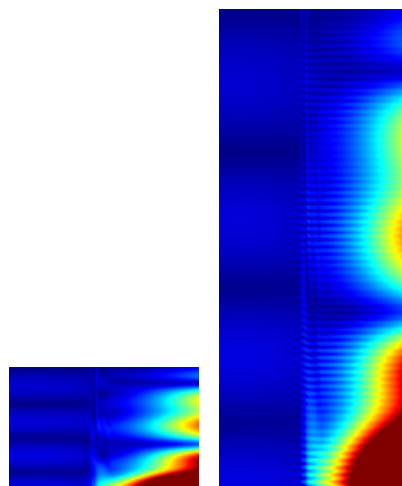


FIGURA 5.8: Imágenes de entrenamiento para actividades de 300 segundos con overlap máximo y tamaño de ventana 200 con DFT de 128 puntos y 512 puntos respectivamente

El objetivo es conseguir espectrogramas que sean suaves visualmente, sin discontinuidades, que permitan una correcta visualización de las zonas donde encontremos picos de frecuencia; siguiendo una metodología similar a la empleada por Wagner et al. [69]. En nuestro caso, se pretende crear una única **red convolucional** que entrenaremos para los diferentes conjuntos de actividad, lo que se traduce en que todas las imágenes serán del mismo tamaño independientemente del tamaño de ventana que estemos tomando para el estudio. Por tanto, para conseguir que los espectrogramas sean equivalentes entre sí:

- Utilizaremos el mismo número de puntos para la Transformada Discreta de Fourier.
- El porcentaje de solapamiento entre ventanas será el mismo.

Tomaremos como referencia el conjunto de actividades de corta duración (60 segundos), ya que al tener menos datos en cada ventana tendremos más limitaciones a la hora de generar el espectrograma. En la tabla 5.3 se puede observar la relación entre el porcentaje de overlap máximo para diferentes tamaños de ventana y el tamaño de la imagen obtenida.

TAMAÑO VENTANA (muestras)	PORCENTAJE OVERLAP (%)	TAMAÑO IMAGEN (píxeles)
2	50 %	129x59
4	75 %	129x57
<b>6</b>	<b>83 %</b>	<b>129x55</b>
10	90 %	129x51
40	97.5 %	129x21

TABLA 5.3: Resumen de generación de espectrogramas para actividades de corta duración

En la figura 5.9 se muestran algunos de los espectrogramas obtenidos para los conjuntos especificados en la tabla anterior.

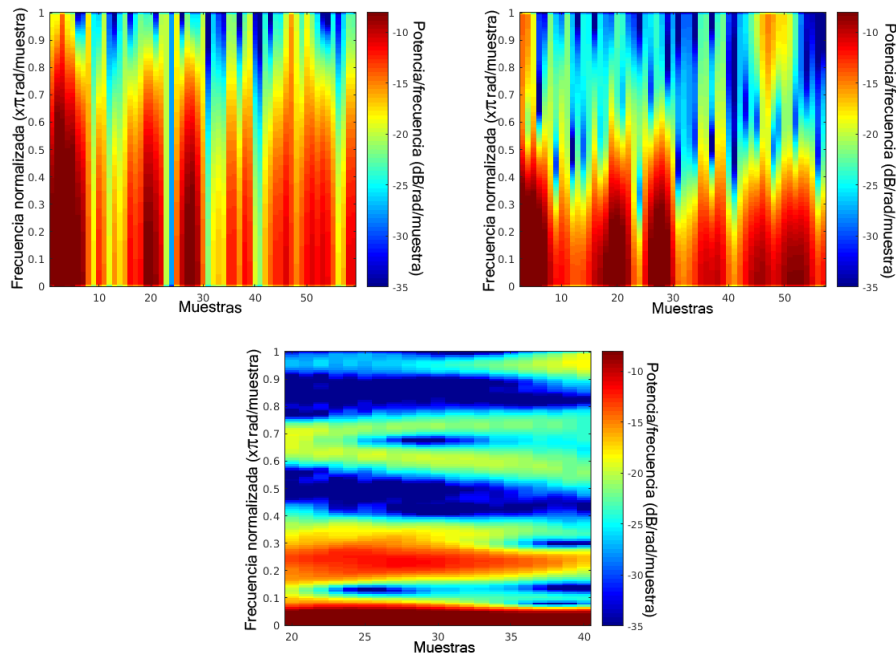


FIGURA 5.9: Espectrogramas para ventanas de tamaño 2,6 y 40 respectivamente

Se observa, que a partir de un overlap del 80 % el espectrograma cumple con los requisitos que perseguimos en cuanto la generación de espectrogramas visualmente suaves, por lo que elegimos el tamaño de ventana de solapamiento en función de aquel que nos ofrezca la máxima resolución temporal para el porcentaje de overlap más pequeño que nos proporcione un espectrograma sin discontinuidades, lo que sucede en el caso de elegir ventanas de 6 segundos para generar el espectrograma, cuyo overlap es del 83 %.

En relación con el número de puntos de la DFT, elegiremos una DFT de 256 puntos, lo que nos dará una resolución frecuencial óptima para la resolución temporal elegida como se muestra en la figura 5.10

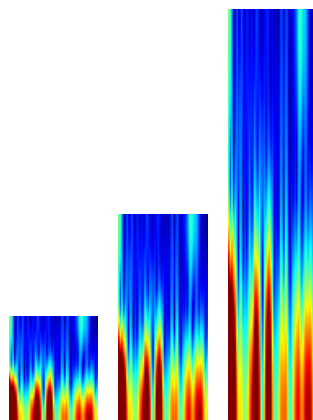


FIGURA 5.10: Imágenes de entrenamiento para una DFT de 128 ,256 y 512 puntos respectivamente.

En el esquema de la figura 5.11 se resume el proceso para generar todas las imágenes que nos servirán como entrada de la red en los diferentes entrenamientos.

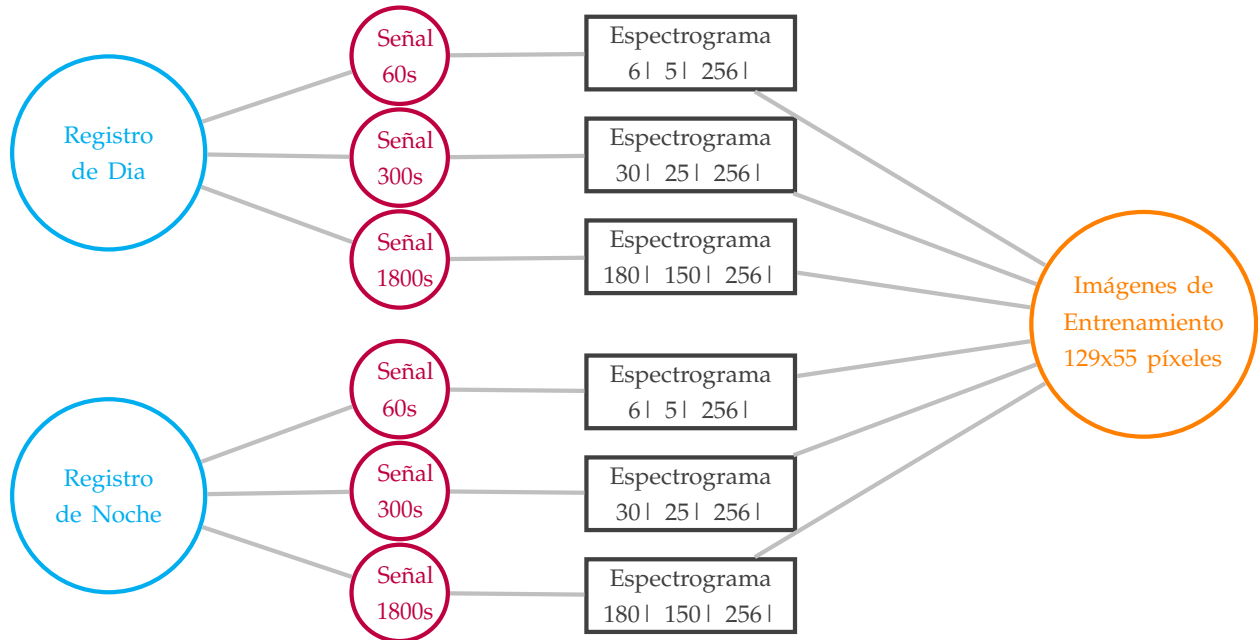


FIGURA 5.11: Esquema de la generación de las imágenes que nos servirán de entrada para la CNN.

### 5.3 DISEÑO DE LA CNN

Como se ha mencionado en el apartado 4.1.1, las redes convolucionales son muy eficientes en la clasificación de imágenes, por lo que vamos a desarrollar nuestro sistema sobre una red de estas características.

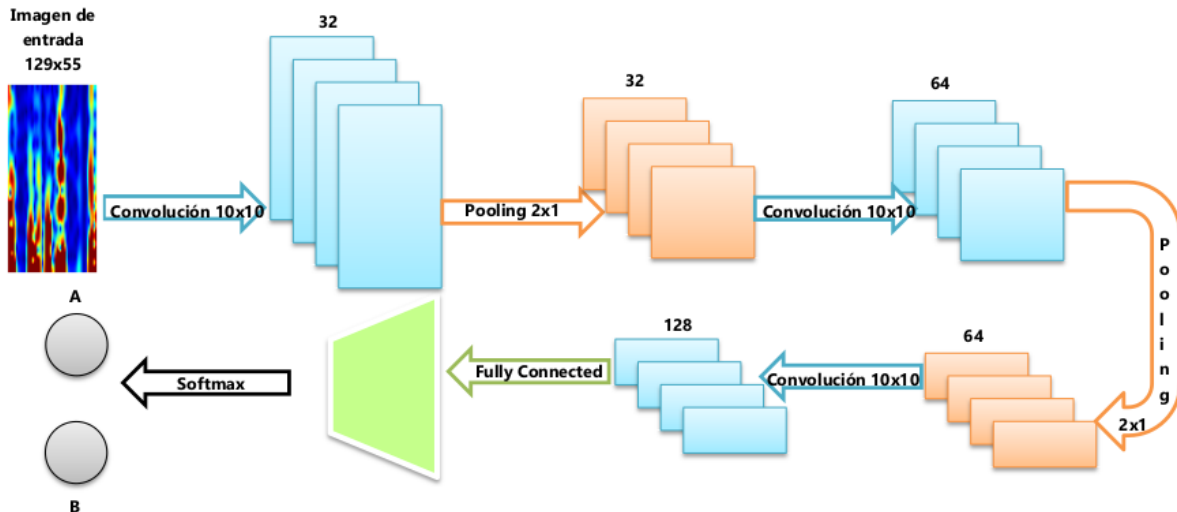


FIGURA 5.12: Estructura de la red Convolucional utilizada para los entrenamientos

Como podemos ver en la figura 5.12 la red que hemos utilizado para desarrollar el sistema de diagnóstico esta formada por 3 capas convolucionales con 32, 64 y 128 filtros respectivamente, dos capas de Pooling, una capa Fully Connected y una capa Softmax. Para La elección de estos parámetros, se partió de una red convolucional sencilla, con una única capa convolucional y se realizaron pequeños entrenamientos con el objetivo de evaluar la influencia de los diferentes parámetros en el porcentaje de *accuracy* (acierto en la clasificación) obtenido. Por ejemplo, en relación con el número de capas convolucionales se notó una mejora considerable al pasar de 1 a 3 capas convolucionales. Al seguir aumentando el número, los entrenamientos tenían una duración mucho más elevada y la mejora no era tan relevante, por lo que el número de capas convolucionales se fijo en 3. Se siguieron metodologías similares para decidir la dimensionalidad de los filtros y el número de los mismos, ajustando finalmente los parámetros para conseguir una buena relación entre el *accuracy* y la duración de los entrenamientos.

### 5.4 ENTRENAMIENTO DE LA RED

El objetivo del entrenamiento es dotar a la red de la capacidad para distinguir entre pacientes diagnosticados con TDAH (*Caso*) y pacientes sanos (*Control*). Se emplearán como señales de entrada las diferentes señales de actividad que hemos procesado y convertido en imágenes anteriormente.

Realizaremos diferentes tipo de entrenamientos, que agruparemos de la siguiente forma:

- **Entrenamientos según los periodos de actividad:** Se entrenará por separado con las imágenes correspondientes a los periodos de actividad corto, medio y largo que hemos definido.
- **Entrenamientos según el momento del día:** Entrenaremos de forma independiente la red para encontrar patrones en las señales pertenecientes a la actividad nocturna o diurna de un paciente.

En la figura 5.13 se puede ver el esquema completo del entrenamiento

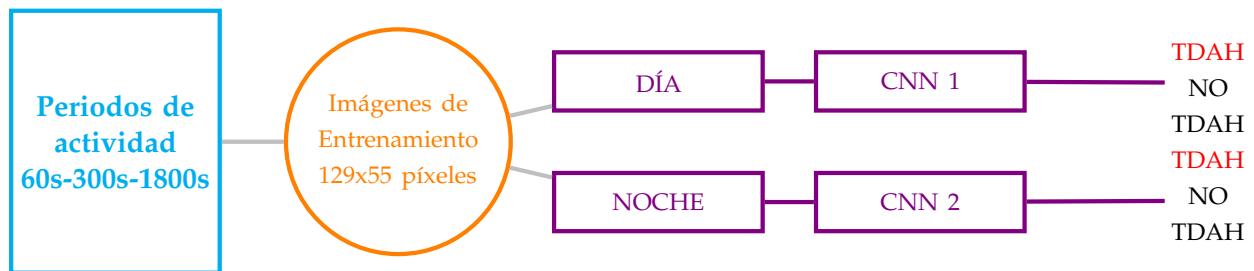


FIGURA 5.13: Proceso de entrenamiento del sistema de diagnóstico.

Sobre las imágenes disponibles para cada uno de los entrenamientos, realizamos diferenciación sobre dos conjuntos:

- **Conjunto de Entrenamiento:** Son las imágenes que la red utiliza para realizar el aprendizaje.
- **Conjunto de Test:** Conjunto de imágenes empleado para la comprobación de la red una vez que esta ha sido entrenada.

Así pues, inicialmente tomaremos el 70 % de los pacientes para formar el conjunto de entrenamiento y el resto para definir el conjunto de *test*.

#### 5.4.1 PARÁMETROS DE ENTRENAMIENTO

El algoritmo empleado en el aprendizaje se conoce como **Stochastic Gradient Descent with Momentum (SGDM)** y es un método iterativo que consiste en minimizar la función de error buscando en la dirección contraria al gradiente de dicha función [70]:

$$\theta_{l+1} = \theta_l - \alpha \nabla E(\theta_l) + \gamma(\theta_l - \theta_{l-1}) \quad (5.3)$$

Donde  $l$  es el número de iteración,  $\alpha$  es la tasa de aprendizaje,  $\gamma$  es el momento,  $\theta$  el vector de parámetros y  $E$  la función de error.

El gradiente se evalúa sobre todo el conjunto de entrenamiento y los parámetros se van actualizando utilizando un subconjunto de entrenamiento llamado **mini-batch**. Cada una de estas evaluaciones se conoce como iteración. Cuando se realiza este proceso sobre todo el conjunto de datos se finaliza una **época**.

La elección de este algoritmo en vez de un algoritmo adaptativo (como puede ser el algoritmo *Adam* o de estimación del momento adaptativo para el aprendizaje) se debe a que existen estudios que han demostrado que el algoritmo SGD es más eficiente en problemas de optimización y aunque es más lento el aprendizaje, las tasas de acierto sobre el conjunto de test son más elevadas [71].

En la tabla 5.4 figuran los valores elegidos para los parámetros configurables.

Learning Rate	Momentum Coefficient	Minibatch Size	Epoc Number
0.01	0.9	32	75

TABLA 5.4: Parámetros utilizados en los entrenamientos

Para la elección de estos parámetros se han tomado los valores por defecto de *Matlab* para el algoritmo SGDM. En cuanto al número de épocas, primero se entreno con un número de épocas mayor. En dicho entrenamiento se observó que, a partir de aproximadamente las 75 épocas las tasas de acierto y pérdida se mantenían estables para el modelo, por lo que se fijo en este valor dicho parámetro.

#### 5.4.2 CARACTERÍSTICAS DE LOS ENTRENAMIENTOS

En esta sección vamos a ver diferentes parámetros asociados al proceso de entrenamiento de la red, así como los resultados de algunos de los entrenamientos realizados sobre el conjunto de datos. Primero, vamos a evaluar el conjunto de imágenes con el que se ha realizado cada entrenamiento. Para evitar sobre entrenar la red para una de las dos clases lo ideal es que el número de imágenes pertenecientes a ambas clases sea **balanceado**. En la tabla 5.5 se muestra el ejemplo de los conjuntos de entrenamiento para uno de los *fold* entrenados.

VENTANA (s)	CASO		CONTROL	
	Día	Noche	Día	Noche
60	150090	31733	165480	41878
300	30094	16188	32996	17847
1800	4860	3495	5423	3847

TABLA 5.5: Imágenes Utilizadas en el entrenamiento para un *fold* cualquiera

Podemos observar que los conjuntos están bastante balanceados. En general, de media para los diez entrenamientos el porcentaje de imágenes pertenecientes a la clase control es ligeramente superior, como muestra la tabla 5.6. El conjunto de imágenes sobre el que se realiza la muestra no siempre es el mismo en número, dado que si analizamos el sistema de forma global habrá pacientes que tenga un mayor número de espectrogramas correspondientes a sus ventanas de actividad. No obstante, son bastante similares para cada entrenamiento.

Ventana	Casos día (%)	Controles día (%)	Casos Noche (%)	Controles Noche (%)
60 s	47.76 %	52.24 %	42.35 %	57.65 %
300 s	47.88 %	52.12 %	47.63 %	52.37 %
1800 s	47.57 %	52.43 %	47.59 %	52.41 %

TABLA 5.6: Porcentaje medio de imágenes empleadas en el entrenamiento

En relación a los tiempos de duración de los entrenamientos, los entrenamientos de imágenes de actividad nocturna han tenido una menor duración que los de la actividad diurna, debido a que el número de imágenes era menor. Lo mismo ocurre cuando comparamos los entrenamientos de 60, 300 y 1800 segundos, siendo estos últimos los de menor duración. En la tabla 5.7 se recogen los tiempos medios de duración de los entrenamientos.

Longitud de Ventana	Entrenamiento Día	Entrenamiento Noche
60	21 horas 26 minutos	6 horas 15 minutos
300	4 horas 50 minutos	2 horas 45 minutos
1800	1 hora 45 minutos	1 hora 30 minutos

TABLA 5.7: Duración media de los entrenamientos para cada fold

En la figura se puede ver un ejemplo de los entrenamientos realizados. En este caso, se muestra la evolución de la *accuracy* y la tasa de pérdida para un entrenamiento diurno de las ventanas de 1800s. Concretamente, se muestra el entrenamiento para el *fold* especificado en la tabla 5.5.



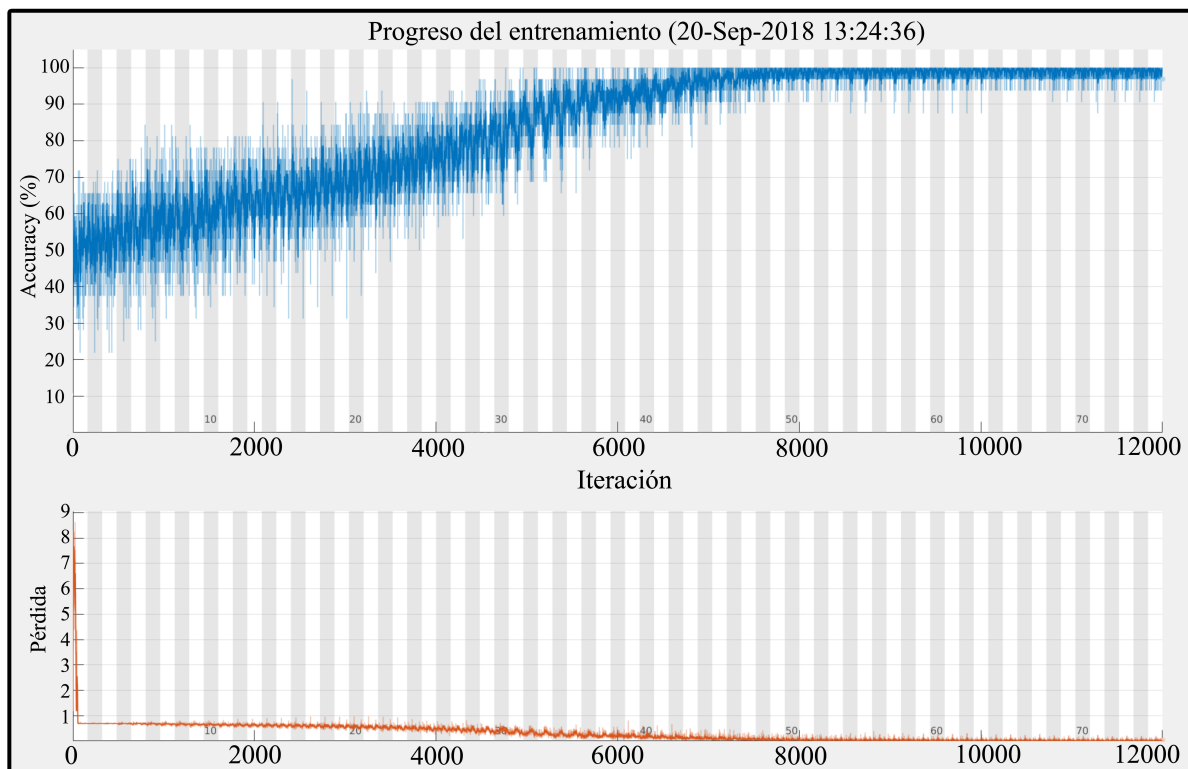


FIGURA 5.14: Entrenamiento para la actividad diurna

## 5.5 OBTENCIÓN DE RESULTADOS: ANÁLISIS COMPLETO DE CADA PACIENTE

Una vez realizados los diferentes entrenamientos, se tiene una red entrenada para diferenciar las ventanas de actividad pertenecientes a pacientes con TDAH y pacientes sanos. No obstante, el objetivo es clasificar cada paciente de forma íntegra. Para ello, se van a emplear diferentes algoritmos de clasificación supervisada que, a partir a los resultados obtenidos para cada paciente en la red neuronal, decidirá si un paciente es un Caso o un Control.

El funcionamiento de los diferentes algoritmos empleados para el clasificador son los que se detallaron en la sección 3.1.2. Para poder utilizar los clasificadores con las predicciones obtenidas en la red convolucional se realiza el siguiente procesado:

- Se realiza un recuento de las ventanas que la red neuronal clasificó como ventanas de Caso y Control y se asocian dichas ventanas a cada paciente, obteniendo así un porcentaje de Caso y Control por cada paciente. Por ejemplo, si un paciente tiene 92 imágenes correspondientes a su actividad y la red neuronal clasificó 78 de ellas como Caso y las otras 14 como Control, esos valores equivaldrían a porcentajes de 0.8478 y 0.1522 respectivamente.
- Hacemos esto para los resultados de la clasificación del día y de la noche.

- Una vez que tenemos los porcentajes asociados a cada paciente, utilizamos como entradas para el clasificador los porcentajes asociados a cantidad de ventanas de **Control** tanto para el día como para la noche y con ello, obtenemos un modelo para cada clasificador que nos servirá para hacer las predicciones correspondientes sobre el conjunto de test.
- Repetiremos el mismo proceso en los 10 *folds* utilizados en el entrenamiento y para los diferentes conjuntos de tamaños de ventanas.

En la figura 5.15 se muestra un esquema de como se realiza este procesado

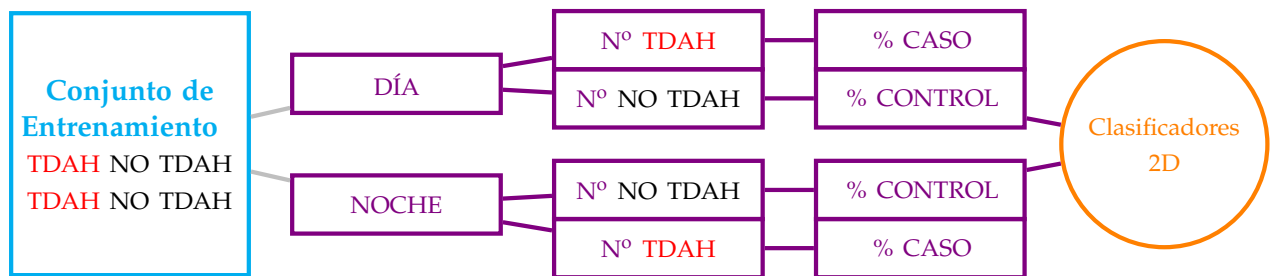


FIGURA 5.15: Proceso de entrenamiento para la clasificación por pacientes

Una vez entrenados los diferentes modelos de los clasificadores, utilizamos estos modelos para hacer las predicciones sobre el conjunto de test, que es el mismo que el utilizado para la red convolucional, adaptando las predicciones de la CNN de forma análoga; es decir, obteniendo los porcentajes correspondientes a ventanas de Caso y Control de cada paciente.

## Capítulo 6

# RESULTADOS Y DISCUSIÓN

En esta sección se discutirán los resultados obtenidos de los diferentes entrenamientos y posterior clasificación de los pacientes. Analizaremos los resultados mediante dos criterios:

- Exactitud (*accuracy*) en la clasificación por ventanas de actividad.
- Resultados obtenidos de la clasificación global de cada paciente.

Para garantizar la independencia de los resultados obtenidos de las particiones de entrenamiento y test, se ha empleado la técnica de **validación cruzada** descrita en la sección 3.2.1, utilizando diez *fold*s diferentes.

### 6.1 PRESTACIONES DE LA RED CONVOLUCIONAL PARA LA CLASIFICACIÓN DE VENTANAS DE ACTIVIDAD

---

Para analizar la eficiencia de la red en cuanto a la clasificación por ventanas de actividad se han utilizado diez *fold*s de entrenamiento. En la tabla 6.1 se muestran los resultados obtenidos para el *accuracy* del entrenamiento.

Longitud de Ventana	Accuracy Día	Accuracy Noche
60	0.5935	0.7137
300	0.6507	0.7045
1800	0.6640	<b>0.7538</b>

TABLA 6.1: Accuracy obtenida del entrenamiento de la CNN para el conjunto de test

La exactitud del modelo en la clasificación por ventanas no es muy elevada, pero se distingue claramente que para las señales de noche es algo mayor. Se muestra la matriz de confusión para el mejor de los *fold*s utilizados para la clasificación.

	Predicción Casos	Predicción Controles	
Casos	1107	352	0.7587
	36.74 %	11.68 %	0.2413
Controles	333	1221	0.7857
	11.05 %	40.53 %	0.2143
	0.7688	0.7762	<b>0.7727</b>
	0.2312	0.2238	<b>0.2273</b>

TABLA 6.2: Matriz de confusión para el mejor *fold* obtenido en el entrenamiento

Observamos que el modelo es razonable en cuanto a la clasificación correcta de aquellas ventanas pertenecientes a las señales de los casos (sensibilidad). Tiene sentido pensar que la clasificación por ventanas tenga unos valores de precisión más bajos, ya que existen periodos de actividad en los que, tanto para una persona diagnosticada con TDAH como un paciente sano, pueden registrarse medidas muy similares en el actígrafo.

Para la finalidad de este trabajo, resulta más interesante el análisis de clasificación por cada paciente, en función de las ventanas pertenecientes a una u otra clase que haya clasificado la red neuronal.

## 6.2 CLASIFICACIÓN DE LOS PACIENTES A PARTIR DE LOS RESULTADOS OBTENIDOS DE LA RED CONVOLUCIONAL

El estudio de la eficiencia de los clasificadores se ha llevado a cabo evaluándolos mediante las métricas presentadas en la sección 3.2.2: **Exactitud, Sensibilidad, Especificidad y Área (AUC) bajo la curva ROC.**

La entrada a los clasificadores se corresponde con un vector de dos componentes, donde cada una de ellas equivale al porcentaje de ventanas que han sido clasificadas como *Control* por la CNN, la primera componente para el tramo de actividad diurna y la segunda para el tramo de actividad nocturna. En las tablas siguientes se ve la relación de probabilidad de los 5 clasificadores utilizados en atención a las métricas mencionadas anteriormente.

Clasificador	Exactitud	Sensibilidad	Especificidad	AUC
KNN	0.9619	0.9333	<b>0.9905</b>	0.9753
Árbol	0.9310	0.8952	0.9667	0.9309
SVM	0.9381	0.8952	0.9810	0.9553
SVM <sub>RBF</sub>	<b>0.9643</b>	0.9424	0.9857	0.9980
SVM q=2	0.9595	<b>0.9952</b>	0.9238	<b>0.9977</b>

TABLA 6.3: Métricas para ventana de 1800s

Clasificador	Exactitud	Sensibilidad	Especificidad	AUC
KNN	0.9833	0.9714	<b>0.9952</b>	0.9857
Árbol	<b>0.9857</b>	0.9810	0.9905	0.9833
SVM	0.9500	0.9286	0.9714	0.9966
SVM <sub>RF</sub>	0.9833	0.9714	<b>0.9952</b>	0.9993
SVM <sub>q=2</sub>	0.9548	<b>1</b>	0.9095	<b>1</b>

TABLA 6.4: Métricas para ventana de 300s

Clasificador	Exactitud	Sensibilidad	Especificidad	AUC
KNN	0.9624	0.9571	0.9667	0.9765
Árbol	0.9524	<b>0.9667</b>	0.9381	0.9524
SVM	0.8905	0.9619	0.8191	0.9846
SVM <sub>RF</sub>	<b>0.9691</b>	0.9524	<b>0.9857</b>	<b>0.9918</b>
SVM <sub>q=2</sub>	0.9405	0.9571	0.9238	0.9909

TABLA 6.5: Métricas para ventana de 60s

En las tablas anteriores se ve claramente que a la hora de clasificar a un sujeto teniendo en cuenta todas sus ventanas de actividad, el sistema es bastante preciso, siendo el peor porcentaje para la tasa de acierto del 89 %, un porcentaje bastante mayor que el que podíamos obtener clasificando las ventanas de forma individual. A la hora de elegir cuál de los clasificadores funciona mejor, hay que tener en cuenta cuál de las tres métricas nos interesa que sea la mejor ajustada.

En el diagnóstico de enfermedades, siempre es preferible asegurar que aquellos pacientes enfermos sean diagnosticados correctamente, frente al perjuicio que suponga que un paciente sano pueda ser inicialmente diagnosticado erróneamente. En términos de las métricas, esto se traduce en que interesan sistemas de diagnóstico de alta sensibilidad. Como criterio, hemos tomado que el mejor clasificador para una métrica concreta será aquel que presente el mejor resultado en al menos dos de los conjuntos de señales estudiados.

Por tanto, atendiendo a la **exactitud**, el mejor clasificador sería el SVM<sub>rbf</sub>, en términos de **sensibilidad** nos quedaríamos con el clasificador SVM<sub>q</sub> y por último en relación con la **especificidad** tomaríamos el clasificador KNN.

En términos de la **AUC** para todos los clasificadores hemos obtenido valores medios por encima del 90 % lo que indica que el sistema es capaz de diferenciar con bastante precisión entre los pacientes de ambas clases.

Otro de los métodos estadísticos más utilizados es el conocido como **diagrama de caja o boxplot**. El boxplot nos permite representar gráficamente una serie de datos numéricos a través de sus cuartiles. En un boxplot podemos distinguir varias zonas:

- **Mediana**: Esta representada como una línea en el interior de la caja y nos indica la posición central del conjunto de los datos.
- **Caja de Rango Intercuartil** : Muestra la distancia entre el primer y el tercer cuartil (Q3-Q1).

- **Bigotes:** Representan los rangos del 25 % superior e inferior de los cuartiles.

Nos centraremos en el análisis de la sensibilidad, la especificidad y la exactitud en base a los boxplots obtenidos para los grupos de entrenamiento empleados.

En las figura 6.1, se puede observar que en términos de exactitud, los valores obtenidos en los diez *folde*s de entrenamiento se ajustan bastante a la distribución para cada clasificador, registrando muy pocos valores atípicos.

En relación a la sensibilidad, podemos observar que se confirma que el mejor clasificador obtenido es el  $SVM_q$ .

Por último, en términos de especificidad es también el clasificador KNN el que nos proporciona las mejores distribuciones para los tres grupos.

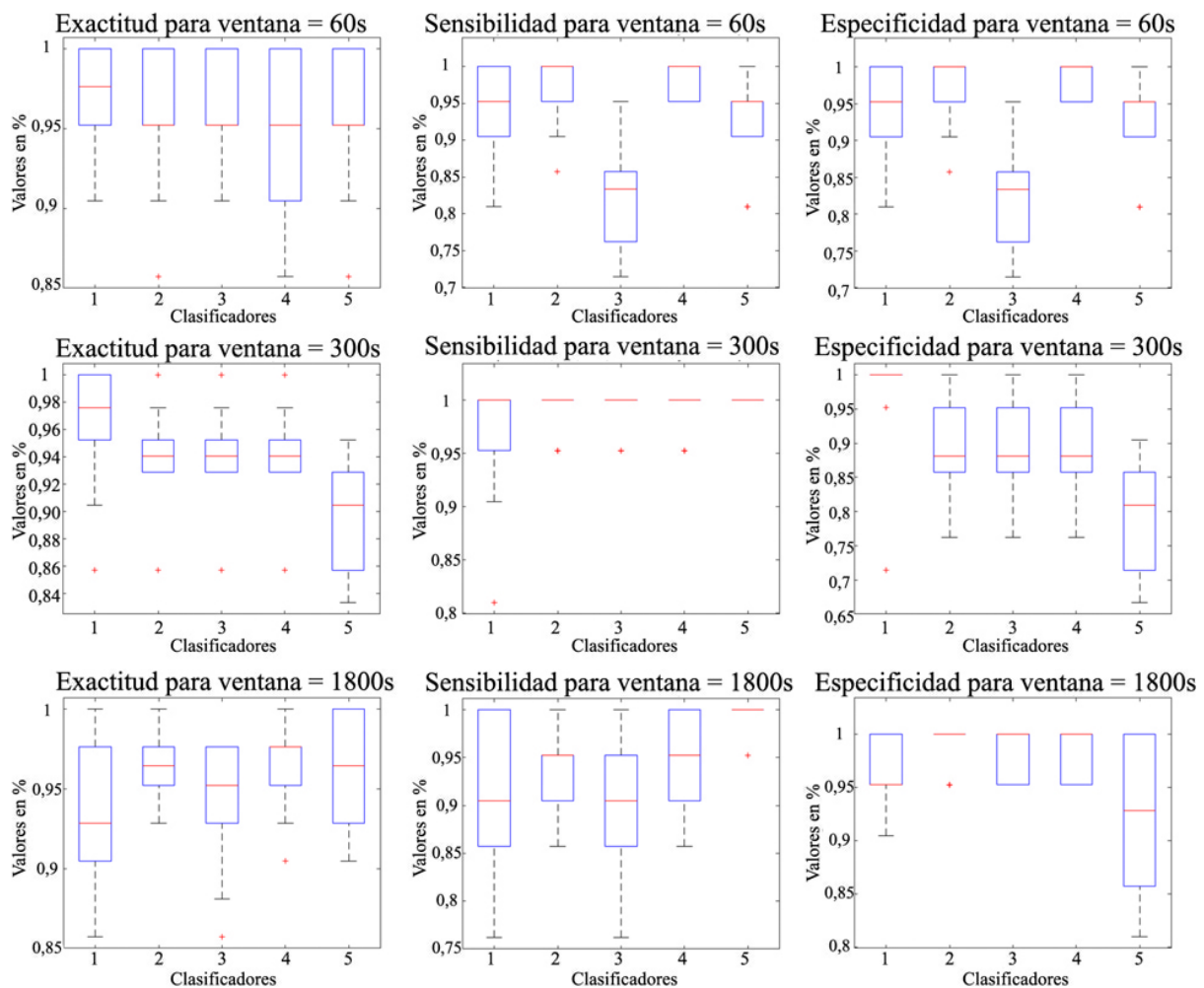


FIGURA 6.1: Boxplots de Exactitud, Sensibilidad y Especificidad para los cinco clasificadores utilizados (KNN, Árbol, SVM, SVM con RBF y SVM con  $q=2$ )

### 6.2.1 ANÁLISIS DE LA CURVA ROC PARA CASOS CONCRETOS

Vamos a evaluar los resultados obtenidos para la curva ROC mediante dos análisis:

- Comparativa de los diferentes clasificadores para un *fold* concreto respecto a la AUC y la curva ROC obtenida.
- Análisis de los mejores en sensibilidad y especificidad respecto a su curva ROC.

Con ello, se pretende reafirmar las conclusiones extraídas a lo largo de la sección.

En la figura 6.2 vemos representada la curva ROC para el peor de los *folds* empleados en el conjunto de test y en la tabla siguiente el valor del AUC obtenido.

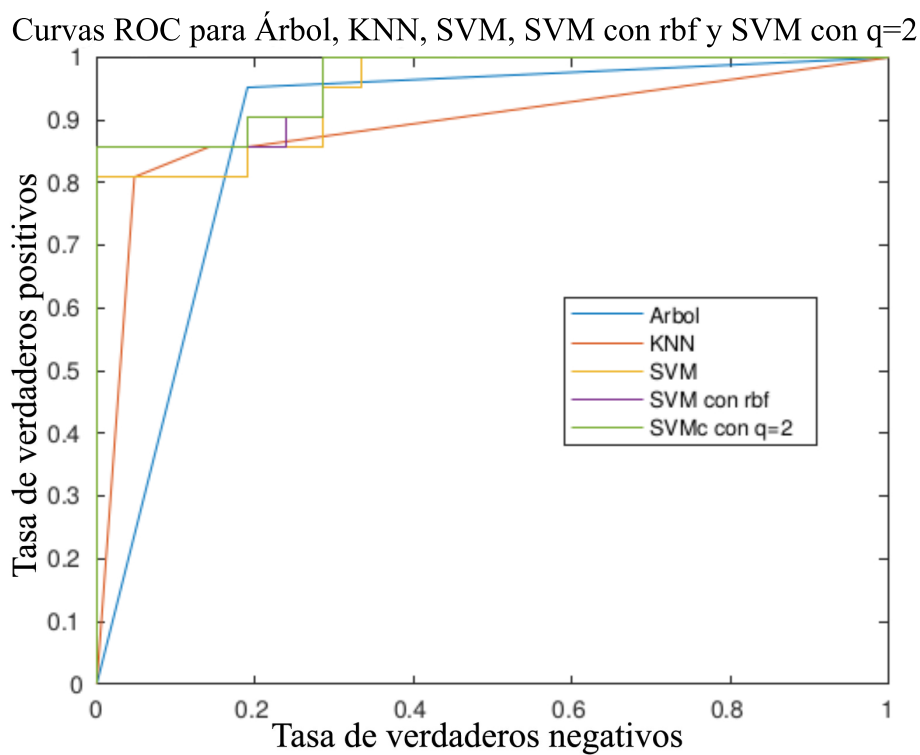


FIGURA 6.2: CURVAS ROC para el peor *fold* en ventanas de 60 segundos

Árbol	KNN	SVM	SVM <sub>RBF</sub>	SVM q=2
0.8809	0.8912	0.9478	0.9615	<b>0.9637</b>

TABLA 6.6: AUCs de los clasificadores ventana de 60s

Observamos que todos los clasificadores proporcionan AUCs superiores a 0.88 , por lo que podemos considerar que, incluso en el peor de los casos obtenidos, el test realizado conserva esa capacidad elevada de discriminación entre clases. Podemos ver en la matriz de confusión que, para la peor curva ROC obtenida, esta clasificando correctamente 18 de los 21 casos.

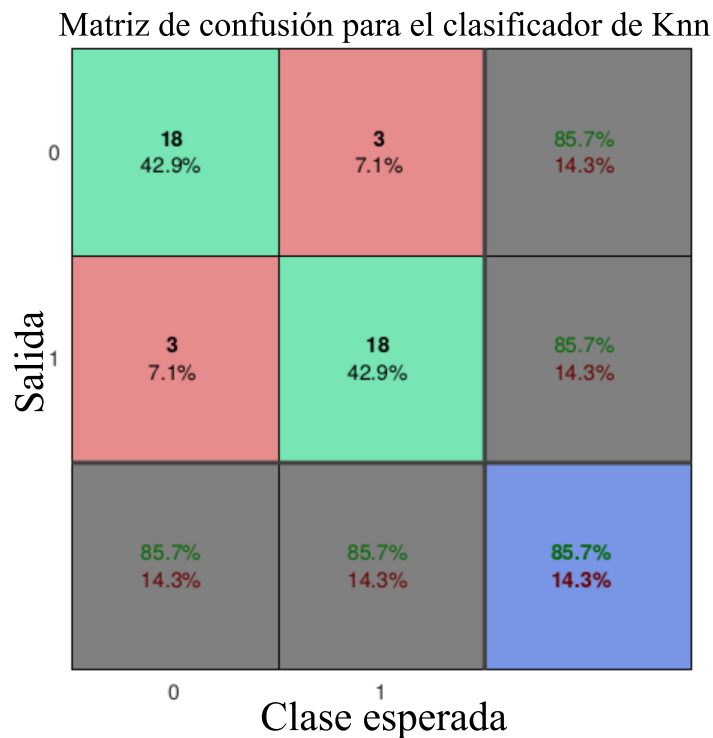
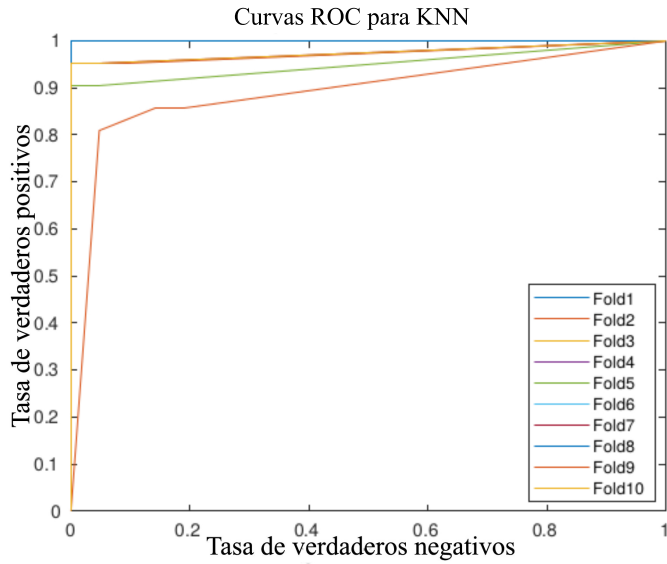


FIGURA 6.3: Matriz de confusión para el caso de peor AUC

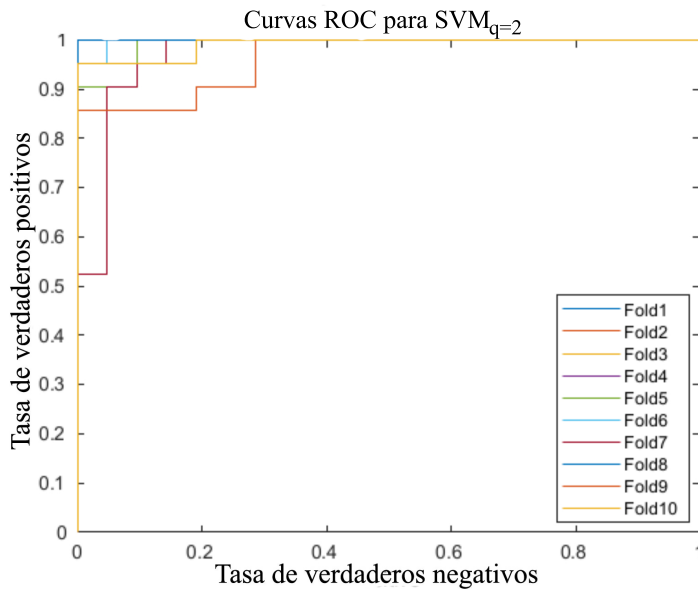
Respecto a la eficiencia de los clasificadores de forma global a lo largo de los *folds*, anteriormente se dedujo que el clasificador que mejor se comportaba en cuanto a sensibilidad era el clasificador **SVM<sub>q</sub>** y en relación con la **especificidad** el clasificador **KNN**. Vamos a observar ahora las Curvas ROC obtenidas para ambos clasificadores en el estudio de ventanas de 60s.





Fold	AUC
Fold 1	1
Fold 2	0.9739
Fold 3	1
Fold 4	1
Fold 5	0.9501
Fold 6	1
Fold 7	0.9750
Fold 8	1
Fold 9	0.8911
Fold 10	0.9750

TABLA 6.7: AUCs de los clasificadores ventana de 60s



Fold	AUC
Fold 1	1
Fold 2	0.99546
Fold 3	1
Fold 4	1
Fold 5	0.99093
Fold 6	0.99773
Fold 7	0.97052
Fold 8	1
Fold 9	0.96372
Fold 10	0.99093

TABLA 6.8: AUCs de los clasificadores ventana de 60s

En las tablas anteriores podemos observar que las AUC de los clasificadores son bastante buenas

en todos los *folders*, por lo que se pueden considerar que los test realizados son fiables. Observando el clasificador que presentaba una mayor especificidad, vemos que el peor de los *folders* entrenados presenta una AUC de 0.8911, lo que quiere decir que en el peor de los casos, el sistema está diferenciando los Casos de los Controles con el 89 % de probabilidad de hacerlo correctamente. En el caso del clasificador que presentaba mayores valores de sensibilidad, nos encontramos ante una probabilidad de acierto en el diagnóstico de como mínimo un 96 % en los *folders* entrenados.

### 6.3 PRESTACIONES DE LA CNN PARA DIFERENTES PROPORCIONES DE LOS CONJUNTOS DE ENTRENAMIENTO Y TEST

Cuando el conjunto de entrenamiento es muy grande, se corre el riesgo de que la red sea capaz de distinguir solo casos muy concretos y se produzcan fallos graves a la hora de establecer una generalización. Para comprobar si un sistema es eficiente, será capaz de realizar una clasificación óptima aunque el conjunto de datos de entrenamiento sea más pequeño. En nuestro caso, hemos evaluado el comportamiento de la red neuronal y los clasificadores para dos conjuntos diferentes de *train-test*: el primero con una relación de 10-90 % y el otro 20-80 %.

Al igual que en la sección anterior, vamos a analizar el comportamiento del sistema desde el punto de vista de la clasificación por ventanas y de la clasificación global. En la tabla 6.9 se muestran los resultados de clasificación por ventanas obtenidos en el entrenamiento para ambas agrupaciones de *train-test*.

Conjunto Train-Test	Longitud de Ventana	Accuracy Día	Accuracy Noche
10-90	60	0.5440	0.6058
	300	0.5571	0.6424
	1800	0.5492	0.7095
20-80	60	0.5547	0.6674
	300	0.5973	0.6551
	1800	0.6059	0.7131

TABLA 6.9: Accuracy obtenida del entrenamiento de la CNN para el conjunto de test

En cuanto a las métricas de ventana, vemos que no hay mucha variación en la precisión. Es lógico que cuanto más ejemplos vea la red neuronal, mejor será la clasificación. No obstante, existe una diferencia muy pequeña entre el *fold* en el que hemos empleado el 10 % de los pacientes para el entrenamiento y la media obtenida para nuestro conjunto inicial; del 11.48 % en el peor caso, lo que se puede traducir en que el sistema es bastante robusto.

En relación con la clasificación global de cada paciente, también se ha realizado el estudio atendiendo a las métricas de exactitud, sensibilidad y especificidad. En las siguientes tablas se ve la relación de estos parámetros en los dos *folders* de entrenamiento que hemos utilizado.

Conjunto Train-Test	Clasificador	Precisión	Sensibilidad	Especificidad
10-90	KNN	<b>0.9538</b>	0.9692	0.9385
	Árbol	0.9000	0.8769	0.9231
	SVM	0.6615	0.3231	<b>1</b>
	SVM <sub>RBF</sub>	<b>0.9538</b>	0.9692	0.9385
	SVM q=2	0.9231	<b>1</b>	0.8462
20-80	KNN	0.8276	0.6724	<b>0.9828</b>
	Árbol	0.8362	0.7069	0.9655
	SVM	0.8190	0.6724	0.9655
	SVM <sub>RBF</sub>	0.8362	0.7069	0.9655
	SVM q=2	<b>0.9310</b>	<b>0.9483</b>	0.9138

TABLA 6.10: Métricas para ventana de 1800s

Conjunto Train-Test	Clasificador	Precisión	Sensibilidad	Especificidad
10-90	KNN	0.8846	0.7692	<b>1</b>
	Árbol	0.8538	0.7231	0.9846
	SVM	0.5000	0	<b>1</b>
	SVM <sub>RBF</sub>	0.8923	0.7846	<b>1</b>
	SVM q=2	<b>0.9538</b>	<b>0.9692</b>	0.9385
20-80	KNN	<b>0.9397</b>	0.8966	<b>0.9828</b>
	Árbol	0.9224	0.8966	0.9483
	SVM	0.8190	0.6724	0.9655
	SVM <sub>RBF</sub>	<b>0.9397</b>	0.8966	<b>0.9828</b>
	SVM q=2	<b>0.9397</b>	<b>1</b>	0.8793

TABLA 6.11: Métricas para ventana de 300s

Conjunto Train-Test	Clasificador	Precisión	Sensibilidad	Especificidad
10-90	KNN	0.5308	0.0769	0.9846
	Árbol	0.9077	0.8308	0.9846
	SVM	0.5077	0.0154	<b>1</b>
	SVM <sub>RBF</sub>	0.5308	0.0769	0.9846
	SVM q=2	<b>0.9231</b>	<b>0.9231</b>	0.9231
20-80	KNN	0.8534	0.9138	0.7931
	Árbol	0.8362	0.7241	<b>0.9483</b>
	SVM	0.8534	0.9138	0.7931
	SVM <sub>RBF</sub>	0.8534	0.9138	0.7931
	SVM q=2	<b>0.8707</b>	<b>1</b>	0.7414

TABLA 6.12: Métricas para ventana de 60s

Se observa que, a pesar de tener un conjunto de entrenamiento muy pequeño, hay algunos casos en los que se obtienen valores de precisión, sensibilidad y especificidad bastante elevados.

Por ejemplo, en el caso del clasificador SVM<sub>q</sub>, para el segundo *fold* hemos obtenido una sensibilidad del 100 %. Cabe destacar, que estos resultados no son extrapolables si no realizamos una validación cruzada para estos tamaños de conjunto, pero nos muestran claramente que el sistema puede llegar a ser capaz de diferenciar un Caso de un Control incluso cuando tiene pocas muestras de ambos, lo que reafirma la idea de robustez que observábamos con las accuracy de ventana. En cuanto a los clasificadores, el que mejores prestaciones presenta es el SVM<sub>q</sub>, ya que es el que parece realizar una clasificación con mayor independencia del tamaño del conjunto de train.

## 6.4 ESTADO DEL ARTE: COMPARATIVA CON OTROS MODELOS

En esta última sección, se busca hacer una comparativa entre los resultados obtenidos por publicaciones previas y los presentados a lo largo de esta sección. Realizaremos la comparativa basándonos en tres criterios:

- Ver la evolución frente a los estudios previos sobre actimetría que se tomaron como referente para iniciar este trabajo.
- Comparar nuestros resultados con los obtenidos por otros estudios que emplean igualmente actimetría y *Deep Learning*.
- Comparar nuestros resultados con el empleo de otras técnicas de adquisición de señales.

### 6.4.1 COMPARATIVA CON ESTUDIOS ANTERIORES

En los estudios tomados como punto de partida para este trabajo [2, 21], se realizó un análisis estadístico clásico para la clasificación. En la tabla 6.13 podemos ver un ejemplo de los mejores valores obtenidos en dicho estudio frente a las mejores métricas que se han conseguido con la red neuronal que hemos planteado.

Modelo	Exactitud	Sensibilidad	Especificidad
CNN + clasificador árbol	0.9857	0.9810	0.9905
Martín et al.	0.8571	0.9500	0.7727

TABLA 6.13: Métricas mejores obtenidas para ambos estudios

En la tabla se muestra el mejor de los casos, con el que se pone de manifiesto que se ha conseguido mejorar ampliamente los tres resultados; en particular la especificidad y la exactitud de la clasificación que han aumentado un 13 % y un 22 % respectivamente sobre el valor anterior.

### 6.4.2 COMPARATIVA CON OTROS TRABAJOS QUE EMPLEAN LA ACTIMETRÍA Y APRENDIZAJE PROFUNDO PARA EL DIAGNÓSTICO

#### 6.4.3 ESTUDIO DE MUÑOZ ET AL

La primera diferencia con dicho estudio es que, en nuestro caso se evalúan los registros tomados a lo largo de la vida cotidiana de los niños, frente al entorno restringido que se plantea en el estudio. Además, en nuestro caso contamos con 148 pacientes, frente a los 22 analizados por Muñoz et al. En la tabla se muestran los resultados obtenidos por este estudio frente a los que se han obtenido en el presente trabajo.

Modelo	Exactitud	Sensibilidad	Especificidad
CNN + clasificador árbol	0.9857	0.9810	0.9905
Muñoz et al. (muñeca)	0.8750	0.6	1
Muñoz et al. (tobillo)	0.9375	0.8	1

TABLA 6.14: Métricas mejores obtenidas para ambos estudios

Podemos observar que, en nuestro caso, los resultados en cuanto a exactitud y sensibilidad para pacientes no utilizados en el entrenamiento es mejor que la obtenida por Muñoz et al; especialmente la sensibilidad. En cuanto a la especificidad, en nuestro caso es mínimamente inferior.

#### 6.4.4 OTROS ESTUDIOS CON ACTIMETRÍA

En el estudio de Mahony et al, que obtenía datos de actimetría a partir de IMUs también se utilizó un conjunto pequeño de pacientes (24 diagnosticados y 19 controles) y la extracción de características se realizó de forma manual; frente a la CNN planteada en el presente estudio. En la tabla 6.15 se ven los resultados obtenidos en dicho estudio.

Por último, cabe mencionar que se realizó otro estudio haciendo uso de CNN de 1D en el que también se emplearon los estudios de Martín et al. como punto de partida.

En este caso, los resultados obtenidos no fueron satisfactorios y han sido mejorados sustancialmente con el presente estudio.

Modelo	Exactitud	Sensibilidad	Especificidad
CNN + clasificador árbol	0.9857	0.9810	0.9905
CNN 1D	0.88	0.92	0.84
Mahony et al.	0.9512	0.9444	0.9565

TABLA 6.15: Métricas mejores obtenidas para ambos estudios

### 6.4.5 COMPARATIVA CON OTROS TRABAJOS QUE EMPLEAN APRENDIZAJE PROFUNDO PARA EL DIAGNÓSTICO

La principal ventaja de la actimetría frente a otras técnicas se encuentra en la facilidad de adquisición de los datos, siendo un método mucho más cómodo para el paciente. En la tabla 6.16 se muestran los resultados obtenidos por los estudios mencionados en el capítulo 2.6 frente a los obtenidos en este trabajo.

Modelo	Exactitud	Sensibilidad	Especificidad
CNN + clasificador árbol	0.9857	0.9810	0.9905
Riaz et al.	0.818	1	0.75
Öztoprak et al.	1	1	1

TABLA 6.16: Métricas mejores obtenidas para ambos estudios

En el estudio de Öztoprak et al. el conjunto de test era solo sobre 10 pacientes, frente a nuestro estudio, con 42 pacientes en el conjunto de test.

## CONCLUSIONES Y LÍNEAS FUTURAS

### 7.1 CONCLUSIONES

---

Al inicio del presente documento, se especificó que el objetivo principal de este trabajo consistía en plantear un sistema de diagnóstico para el TDAH objetivo, rápido, fiable y no intrusivo, empleando la actimetría junto con las técnicas de aprendizaje automático profundo. Para ello, se tomaron como punto de partida las señales procedentes del actímetro de muñeca *ActiGraph GT3x* y a partir de ellas, se inició el procesado para adaptar la señal a lo que se correspondería con una entrada óptima para una red de *Deep Learning*.

Tras analizar diversas redes de aprendizaje automático, se planteó una red neuronal convolucional sencilla que nos sirviera como estructura básica de clasificación. Con el fin de aprovechar al máximo la versatilidad de este tipo de redes, muy eficientes en la clasificación de imágenes, se dividió la señal de entrada en diferentes ventanas temporales, para tener señales más pequeñas, y cada una de ellas se transformó mediante el espectrograma en una señal 2D que sirviera de entrada a la CNN.

Una vez que la red realizó las tareas de clasificación y extracción de las características de las ventanas de actividad de cada sujeto, se planteó una clasificación global que proporcionara un diagnóstico para cada sujeto, determinando así si padecía o no TDAH.

Teniendo en cuenta los resultados obtenidos al finalizar todo el proceso de clasificación se han podido extraer las siguientes conclusiones:

- Las redes neuronales convolucionales pueden resultar una herramienta útil a la hora de encontrar patrones que marquen la diferencia entre un sujeto previamente diagnosticado con TDAH y un sujeto sano.
- La transformación de las señales 1D del actímetro en señales 2D hace que el trabajo con ellas sea mucho más versátil y permite realizar sobre ellas análisis que desde el punto de vista de la señal 1D convencional no parece ser tan efectivo.
- La combinación de las técnicas de aprendizaje profundo junto con la actimetría, se muestran como un sistema robusto y eficiente como ayuda al diagnóstico del TDAH.

## 7.2 LIMITACIONES Y LÍNEAS FUTURAS

---

En el desarrollo de este Trabajo Fin de Grado, hemos encontrado ciertas limitaciones, principalmente a la hora del tratamiento de las señales. Aunque en principio las señales de actividad parecían tener una cantidad elevada de datos, a la hora de realizar la transformación mediante el espectrograma se encontró que muchas de las muestras no aportaban información, ya que las mediciones eran tan pequeñas que provocaban un *overfitting* demasiado elevado en la CNN, haciendo que esta solo pudiera distinguir entre señales con algo de actividad y aquellas cuya potencia era 0. No obstante, se logra subsanar este problema seleccionando solo aquellos datos que mostrarán un mínimo razonable de actividad y estudiando detalladamente lo que queríamos reflejar en los espectrogramas.

Otro de los puntos a destacar es que, aunque la cantidad de imágenes que analiza la red es elevada, en realidad el número global de pacientes es limitado. Sería interesante probar con pacientes nuevos para ver si el comportamiento eficiente del sistema se sigue manteniendo. También podría resultar de interés acotar los grupos de edades de los pacientes, ya que la actividad de los niños varía sustancialmente en función de la edad, por lo que comparar las ventanas de actividad de niños en edades cercanas podría ayudar a un mejor entendimiento de la patología.

Las redes convolucionales realizan la extracción de características de forma automática y oculta, por lo que resultaría interesante ver qué filtros se están activando en cada capa de la red para las diferentes señales de Casos y Controles y utilizar estos datos para intentar dar a los resultados una interpretación médica; es decir, buscar la explicación biológica de dónde pueden estar dichas diferencias. Además, podríamos ajustar los parámetros de la red para su mejor funcionamiento en la detección de la patología, creando una red más compleja que nos proporcionará valores de *accuracy* mayores en la clasificación por ventanas de actividad. Conseguir mayores valores de *accuracy* nos permitiría crear un sistema no solo de diagnóstico, sino de seguimiento de la enfermedad, que permitiera tratar a los pacientes de forma más personalizada, viendo en qué momentos podría estar presentando anomalías y ajustar el tratamiento del TDAH a como se manifiesta la patología en cada persona.

En definitiva, los resultados obtenidos en este TFG han resultado prometedores para el objetivo que se perseguía, pero aún queda trabajo por hacer para conseguir que el diagnóstico objetivo del TDAH mediante la técnica planteada sea una realidad.



## BIBLIOGRAFÍA

- [1] American Psychiatric Association. Diagnostic And Statistical Manual Of Mental Disorders, (DSM-V). *American Journal of Psychiatry*, 5ª edición, 2013.
- [2] D Martín Martínez, P Casaseca de la Higuera, S Alberola López, J Andrés de Llano, JA López Villalobos, J Ardura Fernández, and C Alberola López. Nonlinear Analysis Of Actigraphic Signals For The Assessment Of The Attention-Deficit/Hyperactivity Disorder (ADHD). *Medical Engineering & Physics*, 34(9):1317–1329, 2012.
- [3] MathWorks MATLAB for Deep Learning. <https://es.mathworks.com/>, Último acceso Diciembre 2018.
- [4] Laura Igual, Joan Carles Soliva, Antonio Hernández-Vela, Sergio Escalera, Xavier Jiménez, Oscar Vilarroya, and Petia Radeva. A fully-automatic caudate nucleus segmentation of brain mri: Application in volumetric analysis of pediatric attention-deficit/hyperactivity disorder. *BioMedical Engineering OnLine*, 10(1):105, Dec 2011.
- [5] Samuele Cortese, Robin Azoulay, Francisco Castellanos, François Chalard, Michel Lecendreux, David Chechin, Richard Delorme, Guy Sebag, Andrea Sbarbati, Marie-Christine Mouren, Bernardo Dalla Bernardina, and Eric Konofal. Brain iron levels in attention-deficit/hyperactivity disorder: A pilot mri study. *the world journal of biological psychiatry. The official journal of the World Federation of Societies of Biological Psychiatry.*, 13:223–31, 2011.
- [6] Dongchuan Yu. Additional brain functional network in adults with attention-deficit/hyperactivity disorder: A phase synchrony analysis. *PLOS ONE*, 8, 01 2013.
- [7] Robert J. Gillies, Paul E. Kinahan, and Hedvig Hricak. Radiomics: Images are more than pictures, they are data. *Radiology*, 278(2):563–577, Feb 2016.
- [8] Huaiqiang Sun, Ying Chen, Qiang Huang, Su Lui, Xiaoqi Huang, Yan Shi, Xin Xu, John A. Sweeney, and Qiyong Gong. Psychoradiologic utility of mr imaging for diagnosis of attention deficit hyperactivity disorder: A radiomics analysis. *Radiology*, 287(2):620–630, May 2018.
- [9] Alba Guzmán, Ernesto Pereda, Soledad Mañas, Leopoldo D Méndez, Almudena González, and Julián J González. Electroencephalography signatures of attention-deficit/hyperactivity disorder: clinical utility. *Clinical Neurophysiology*, 2015.

- [10] Mehran Ahmadlou, Reza Rostami, and Vahid Sadeghi. Which attention-deficit/hyperactivity disorder children will be improved through neurofeedback therapy? a graph theoretical approach to neocortex neuronal network of adhd. *Neuroscience Letters*, 516(1):156–160, Dec 2012.
- [11] Tian Liu, Yanni Chen, Pan Lin, and Jue Wang. Small-World Brain Functional Networks in Children With Attention-Deficit/Hyperactivity Disorder Revealed by EEG Synchrony. *Clinical EEG and neuroscience*, 2014.
- [12] Hüseyin Öztoprak, Mehmet Toycan, Yasar Kemal Alp, Orhan Arıkan, Elvin Dogutepe, and Sirel Karakas. Machine-based classification of adhd and nonadhd participants using time/frequency features of event-related neuroelectric activity. *Neuropsychiatric Disease and Treatment*, 128(12):2400–2410, 2017.
- [13] Agatha Lenartowicz and Sandra K. Loo. Use of eeg to diagnose adhd. *Current Psychiatry Reports*, 16(11):498, Sep 2014.
- [14] Celestino Rodríguez, Paloma González-Castro, Luís Álvarez, Lorena Vicente, José C. Núñez, Julio A. González-Pienda, David Álvarez, Ana Bernardo, and Rebeca Cerezo. Nuevas técnicas de evaluación en el Trastorno por Déficit de Atención e Hiperactividad (TDAH). *European Journal of Education and Psychology*, 4(1):63–73, 2011.
- [15] Celestino Rodríguez, Marisol Fernández-Cueli, María Paloma González-Castro, Luis Álvarez, and David Álvarez García. Diferencias en la fluidez sanguínea cortical en los subtipos de TDAH. Un estudio piloto. *Aula Abierta*, 39(1):25–36, 2011.
- [16] Paloma González-Castro, Celestino Rodríguez, Ángel López, Marisol Cueli, and Luis Álvarez. Attention Deficit Hyperactivity Disorder, differential diagnosis with blood oxygenation, beta/theta ratio, and attention measures. *International Journal of Clinical and Health Psychology*, 13:101–109, March 2013.
- [17] Arthur S. Walters, Rosalia Silvestri, Marco Zucconi, Ranju Chandrashekariah, and Eric Konofal. Review of the possible relationship and hypothetical links between attention deficit hyperactivity disorder (adhd) and the simple sleep related movement disorders, parasomnias, hypersomnias, and circadian rhythm disorders. *Journal of Clinical Sleep Medicine*, 4(6):591–600, 2008.
- [18] Allan Hvolby. Associations of sleep disturbance with adhd: implications for treatment. *ADHD Attention Deficit and Hyperactivity Disorders*, 7(1):1–18, Mar 2015.
- [19] Aarti Sathyanarayana, Shafiq Joty, Luis Fernandez-Luque, Ferda Ofli, Jaideep Srivastava, Ahmed Elmagarmid, Shahrhad Taheri, and Teresa Arora. Impact of physical activity on sleep: a deep learning based exploration. *arXiv preprint arXiv:1607.07034*, 2016.
- [20] Lena Granovsky, Gabi Shalev, Nancy Yacovzada, Yotam Frank, and Shai Fine. Actigraphy-based sleep/wake pattern detection using convolutional neural networks. *arXiv preprint arXiv:1802.07945*, 2018.

- [21] P. Casaseca-de-la-Higuera, D. Martín-Martínez, S. Alberola-López, J. María Andrés-de-Llano, J. Antonio López-Villalobos, J. Ramón-Garmendia Leiza, and C. Alberola-López. Automatic diagnosis of adhd based on multichannel nonlinear analysis of actimetry registries. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 4204–4207, Aug 2012. doi: 10.1109/EMBC.2012.6346894.
- [22] Mario Muñoz-Organero, Lauren Powell, Ben Heller, Val Harpin, and Jack Parker. Automatic extraction and detection of characteristic movement patterns in children with adhd based on a convolutional neural network (cnn) and acceleration images. *Sensors*, 18(11):3924, 2018.
- [23] Niamh O’Mahony, Blanca Florentino-Liano, Juan J Carballo, Enrique Baca-García, and Antonio Artés Rodríguez. Objective diagnosis of adhd using imus. *Medical engineering & physics*, 36(7):922–926, 2014.
- [24] Atif Riaz, Muhammad Asad, Eduardo Alonso, and Greg Slabaugh. Fusion of fmri and non-imaging data for adhd classification. *Computerized Medical Imaging and Graphics*, 65: 115–128, 2018.
- [25] Hüseyin Öztoprak, Mehmet Toycan, Yaşar Kemal Alp, Orhan Arıkan, Elvin Doğutepe, and Sirel Karakaş. Machine-based classification of adhd and nonadhd participants using time/frequency features of event-related neuroelectric activity. *Clinical Neurophysiology*, 128(12):2400–2410, 2017.
- [26] S. Cortese, S. Faraone, E. Konofal, and M. Lecendreux. Sleep in Children With Attention-Deficit/Hyperactivity Disorder: Meta-Analysis of Subjective and Objective Studies. *J. Am. Acad. Child. Adolesc. Psychiatry*, 48:894–908, 2009.
- [27] R.W. Tuthill. Hair Lead Levels Related to Children’s Classroom Attention-deficit Behavior. *Arch. Environ. Health*, 52:214–220, 1996.
- [28] R.E. Weiss, M.A. Stein, B. Trommer, and S. Refetoff. Attention-Deficit Hyperactivity Disorder and Thyroid Function. *J. Pediatr.*, 123:539–545, 1993.
- [29] M. El-Faddagh, M. Laucht, A. Maras, L. Vöhringer, and M. H. Schmidt. Association of Dopamine D4 Receptor (DRD4) Gene with Attention-Deficit/Hyperactivity Disorder (ADHD) in a High-Risk Community Sample: a Longitudinal Study from Birth to 11 Years of Age. *J. Neural. Transm.*, 111:883–889, 2004.
- [30] M.M Van-Veen, J.J.S. Kooij, M.M. Boonstra, M.C.M.Gordijn, and E.J.W. Van-Someren. Delayed Circadian Rhythm in Adults with Attention-Deficit/Hyperactivity Disorder and Chronic Sleep-Onset Insomnia. *Biol. Psychiatry*, 67:1091 – 1096, 2010.
- [31] S. Kurtek, E. Klassen, Zhaohua Ding, S.W. Jacobson, J.L. Jacobson, M.J. Avison, and A. Srivastava. Parameterization-Invariant Shape Comparisons of Anatomical Surfaces. *IEEE Trans. Med. Imag.*, 30:849 –858, 2011.
- [32] M. Ashtari, S. Kumra, S.L. Bhaskar, T. Clarke, E. Thaden, K.L. Cervellione, J. Rhinewine, J.M. Kane, A. Adesman, R. Milanaik, J. Maytal, A. Diamond, P. Szeszko, and B.A. Ardekani.

- Attention-Deficit/Hyperactivity Disorder: A Preliminary Diffusion Tensor Imaging Study. *Biol. Psychiatry*, 57:448–455, 2005.
- [33] T.J. Silk, N. Vance, A. Rinehart, J.L. Bradshaw, and R. Cunningtin. White-Matter Abnormalities in Attention Deficit Hyperactivity Disorder: A Diffusion Tensor Imaging Study. *Hum. Brain. Map*, 30:2757–2765, 2009.
- [34] L. Zimmer. Positron Emission Tomography Neuroimaging for a Better Understanding of the Biology of ADHD. *Neuropharmacology*, 57:601–607, 2009.
- [35] M. Modarres-Zadeh, Xueyan Xu, A. Jaromczyk, and S. Senders. A Neuro-Behavioral Test and Algorithms for Screening and Evaluating Therapy in ADHD. In *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*, pages 2486–2489, 2005.
- [36] J.A. Gutierrez Gnechi, J.C. Herrera Garcia, and J. de Dios Ortiz Alvarado. Auxiliary Neurofeedback System for Diagnostic of Attention Deficit Hyperactivity Disorder. In *Electronics, Robotics and Automotive Mechanics Conference, 2007. CERMA 2007*, pages 135–138, 2007.
- [37] S. Hyun-Lee, B. Abibullaev, Won-Seok Kang, Yunhee Shin, and Jinung An. Analysis of Attention Deficit Hyperactivity Disorder in EEG Using Wavelet Transform and Self Organizing Maps. In *Control Automation and Systems (ICCAS), 2010 International Conference on*, pages 2439–2442, 2010.
- [38] D. Ming, M. Zhang, Y. Xi, H. Qi, Y. Hu, and K.D.K. Luk. Multiscale entropy analysis of attention related EEG based on motor imaginary potential. In *Computational Intelligence for Measurement Systems and Applications, 2009. CIMSA '09. IEEE International Conference on*, pages 24–27, 2009.
- [39] Y. Nan, J. Wang, S.A. Xue, H. Sheng, Y. Jiao, and J. Wang. Analysis of Propagation of Multi-Channel EEG in the Test of Sustained Attention. In *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pages 1666–1669, 2010.
- [40] T.S. Hale, S.L. Smalley, P.D. Walshaw, G. Hanada, J. Macion, J.T. McCracken, J.J. McGough, and S.K. Loo. Atypical EEG Beta Asymmetry in Adults with ADHD. *Neuropsychologia*, 48: 3532–3539, 2010.
- [41] F. Ghassemi, M.H. Moradi, M. Tehrani-Doost, and V. Abootalebi. Classification of ADHD/Normal Participants Using Frequency Features of ERP's Independent Components. In *Biomedical Engineering (ICBME), 2010 17th Iranian Conference of*, pages 1–4, 2010.
- [42] P. Castro-Cabrera, J. Gómez-García, F. Restrepo, O. Moscoso, and G. Castellanos-Dominguez. Evaluation Of Feature Extraction Techniques On Event-Related Potentials For Detection Of Attention-Deficit/Hyperactivity Disorder. In *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pages 851–854, 2010.

- [43] C. Demanuele, C.J. James, E.J.S. Sonuga-Barke, and A. Capilla. Low Frequency Phase Synchronisation Analysis of MEG Recordings from Children with ADHD and Controls Using Single Channel ICA. In *Advances in Medical, Signal and Information Processing, 2008. MEDSIP 2008. 4th IET International Conference on*, pages 1–4, 2008.
- [44] R. Gruber, A. Sadeh, and A. Raviv. Instability of Sleep Patterns in Children With Attention-Deficit/Hyperactivity Disorder. *J. Am. Acad. Child. Adolesc. Psychiatry*, 39:495–501, 2000.
- [45] R. Silvestri, A. Gagliano, I. Aricò, T. Calarese, C. Cedro, O. Bruni, R. Conduro, E. Germanó, G. Gervasi, R. Siracusano, G. Vita, and P. Bramanti. Sleep Disorders in Children with Attention-Deficit/Hyperactivity Disorder (ADHD) Recorded Overnight by Video-Polysomnography. *Sleep Med.*, 10:1132–1138, 2009.
- [46] M.H. Teicher, Y. Ito, C.A. Glod, and P. Wallace. Quantitative Assessment of Fidgeting in Attention-Deficit Hyperactivity Disorder (ADHD). *Am. Acad. Child Adolescence Psychiatry*, 9:42–43, 1993.
- [47] M.H. Teicher, Y. Ito, C.A. Glod, and N.I. Barber. Objective Measurement of Hyperactivity and Attentional Problems in ADHD. *J. Am. Acad. Child. Adolesc. Psychiatry*, 35:334–342, 1996.
- [48] A.V. Dane, R.J. Schachar, and R. Tannok. Does Actigraphy Differentiate ADHD Subtypes in a Clinical Research Setting? *J. Am. Acad. Child. Adolesc. Psychiatry*, 39:752–760, 2000.
- [49] M.H. Teicher, C. A. Glod, K. Pahlavan, D. Harper, E. Magnus, and F. Wren. Locomotor Activity and the Diagnosis of ADHD. *Am. Psychiatr Assoc. Abstr.*, 141:98, 1991.
- [50] H.J. Kam, K. Lee, S.M. Cho, Y.M. Shin, and R.W. Park. High-Resolution Actigraphic Analysis of ADHD: A Wide Range of Movement Variability Observation in Three School Courses – A Pilot Study. *Healthc. Inform. Res.*, 17(19–37), 2011.
- [51] S. Y. Kung. *Support vector machines*. Cambridge University Press, 2014. doi: 10.1017/CBO9781139176224.016.
- [52] Enrique J Carmona Suárez. Tutorial sobre máquinas de vectores soporte (svm). *Tutorial sobre Máquinas de Vectores Soporte (SVM)*, 2014.
- [53] Clasificación: Roc y auc <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc?hl=es-419>, Último acceso Marzo 2019.
- [54] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.
- [55] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61: 85–117, 2015.
- [56] Pedro Isasi Viñuela and IM Galván León. Redes de neuronas artificiales. *Un Enfoque Práctico*, Editorial Pearson Educación SA Madrid España, 2004.
- [57] Rodrigo Salas. Redes neuronales artificiales. *Universidad de Valparaíso. Departamento de Computación*, 1, 2004.

- [58] Fernando Izaurieta and Carlos Saavedra. Redes neuronales artificiales. *Departamento de Física, Universidad de Concepción Chile*, 2000.
- [59] Francisco José Palacios Burgos. Redes neuronales con gnu/linux. *Herramientas en GNU/Linux para estudiantes universitarios*, 2003.
- [60] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.
- [61] Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*, 2014.
- [62] Dimitri P Bertsekas and John N Tsitsiklis. Neuro-dynamic programming: an overview. In *Proceedings of the 34th IEEE Conference on Decision and Control*, volume 1, pages 560–564. IEEE Publ. Piscataway, NJ, 1995.
- [63] S Kevin Zhou, Hayit Greenspan, and Dinggang Shen. *Deep learning for medical image analysis*. Academic Press, 2017.
- [64] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, et al. Recent advances in convolutional neural networks. *Pattern Recognition*, 77:354–377, 2018.
- [65] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. URL <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neuralnetworks.pdf>.
- [66] Diego Calvo. [://www.diegocalvo.es/red-neuronal-convolucional/](http://www.diegocalvo.es/red-neuronal-convolucional/), Último acceso diciembre 2018.
- [67] D. Yu and L. Deng. *Automatic Speech Recognition: A Deep Learning Approach*. Signals and Communication Technology. Springer London, 2014.
- [68] Orestis Tsinalis, Paul M Matthews, Yike Guo, and Stefanos Zafeiriou. Automatic sleep stage scoring with single-channel eeg using convolutional neural networks. *arXiv preprint arXiv:1610.01683*, 2016.
- [69] Daniel Wagner, Kathrin Kalischewski, Jorg Velten, and Anton Kummert. Activity recognition using inertial sensors and a 2-d convolutional neural network. In *Multidimensional (nD) Systems (nDS), 2017 10th International Workshop on*, pages 1–6. IEEE, 2017.
- [70] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.
- [71] Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht. The marginal value of adaptive gradient methods in machine learning. In *Advances in Neural Information Processing Systems*, pages 4148–4158, 2017.

## Apéndice A

# ADAPTACIÓN PARA OTROS ENTORNOS DE PROGRAMACIÓN

A la hora de crear la red convolucional que ha servido de base para el sistema de diagnóstico propuesto en este TFG, se ha utilizado Matlab como entorno de programación, debido a la versatilidad que proporciona a la hora del tratamiento de señal. No obstante; con el auge actual en el desarrollo del *Deep Learning*, han surgido otros entornos que proporcionan un gran rendimiento con redes de aprendizaje profundo. En este apéndice, se pretende introducir unas nociones básicas sobre como trasladaríamos nuestro sistema al entorno que mejor esta demostrando comportarse para este tipo de redes: **Tensorflow**. En concreto, dividiremos este apéndice en tres secciones:

- Introducir el entorno y sus posibles ventajas.
- Pautas para crear una red neuronal convolucional.

### A.1 TENSORFLOW

---

TensorFlow consiste en una biblioteca de código abierto desarrollada por Google con la finalidad de realizar aprendizaje automático a través de un rango de tareas. Permite la creación de sistemas capaces de construir y entrenar redes neuronales para la detección de patrones. Aunque proporciona APIs en varios lenguajes, la más utilizada es la existente para Python, debido a que permite utilizar **Keras**; biblioteca de redes neuronales diseñada para Python. En nuestro caso, nos basaremos en esta API debido a que nos proporciona ciertas ventajas:

- Tiene una interfaz simple y consistente, optimizada para el usos de casos comunes.
- Los modelos Keras se hacen conectando bloques de construcción configurables entre sí, con pocas restricciones.
- Presenta facilidades a la hora de extender el modelo, permitiendo al usuario escribir bloques de construcción personalizados.

- Al estar construida sobre Python, proporciona versatilidad para poder transformar nuevas funcionalidades relacionadas con el tratamiento matemático, por lo que podríamos transformar gran parte de nuestras funciones al entorno nuevo de trabajo con cierta sencillez.

## A.2 CREACIÓN DE UNA CNN SOBRE TENSORFLOW

---

El proceso de creación de un modelo de clasificación puede dividirse en dos fases: la primera, el diseño del modelo propiamente dicho y la segunda, la comprobación mediante el entrenamiento del funcionamiento y eficiencia del mismo. En esta sección, se presentará el código para realizar ambas fases haciendo uso de Keras.

### A.2.1 CÓDIGO PARA LA CREACIÓN DE UNA CNN SOBRE TENSORFLOW

---

En esta sección definiremos lo que sería una red equivalente a la que hemos utilizado en nuestro trabajo, construida sobre Tensorflow + Keras.

- Primero, cargamos las librerías necesarias para poder crear la red neuronal

```
import numpy as np
from scipy import misc
from PIL import Image
import glob
import matplotlib.pyplot as plt
import scipy.misc
from matplotlib.pyplot import imshow
%matplotlib inline
from IPython.display import SVG
import cv2
import seaborn as sn
import pandas as pd
import pickle
import keras
from keras import layers
from keras.layers import Flatten, Input, Add, Dense, Activation,
ZeroPadding2D, BatchNormalization, Flatten, Conv2D,
AveragePooling2D, MaxPooling2D, GlobalMaxPooling2D, Dropout
from keras.models import Sequential, Model, load_model
from keras.preprocessing import image
from keras.preprocessing.image import load_img
from keras.preprocessing.image import img_to_array
from keras.applications.imagenet_utils import decode_predictions
from keras.utils import layer_utils, np_utils
from keras.utils.data_utils import get_file
```



```

from keras.applications.imagenet_utils import preprocess_input
from keras.utils.vis_utils import model_to_dot
from keras.utils import plot_model
from keras.initializers import glorot_uniform
from keras import losses
import keras.backend as K
from keras.callbacks import ModelCheckpoint
from sklearn.metrics import confusion_matrix, classification_report,
roc_curve, auc_score, auc
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import RobustScaler, MinMaxScaler,
\StandardScaler

import tensorflow as tf

```

- Una vez cargadas las librerías necesarias, definiríamos el modelo

```

def definir_capas():
    model = Sequential()
    model.add(Conv2D(32, kernel_size=(10, 10), input_shape=(129, 155, 1),
    activation='relu'))
    model.add(BatchNormalization())
    model.add(MaxPooling2D(pool_size=(2, 1)), strides=(2,1))

    model.add(Conv2D(64, kernel_size=(10, 10), activation='relu'))
    model.add(BatchNormalization())
    model.add(MaxPooling2D(pool_size=(2, 1)), strides=(2,1))

    model.add(Flatten())
    model.add(Dense(2, activation='softmax'))

    SGD=keras.optimizers.SGD(learningRate=0.01, momentum = 0.9,
    decay =0.0)
    model.compile(loss='binary_crossentropy', optimizer=SGD,
    metrics=['accuracy'])

    return model

```

- Entrenamos el modelo y obtenemos las métricas para la clasificación por ventanas

```

model.fit(X_train, train_label, epochs=75, batch_size=32,
    verbose=1, validation_data=(X_test, Y_test), callbacks=[history])

test_scores=model.evaluate(X_test, Y_test, verbose=1)
predictedLabels1=model.predict_classes(X_train, batch_size = 32,
    verbose = 0, steps = NULL)

```

```
predictedLabels2=model.predict_classes(X_test ,batch_size = 32,  
verbose = 0, steps = NULL)  
predictedProb1=model.predict_proba(X_train ,batch_size = 32,  
verbose = 0, steps = NULL)  
predictedProb2=model.predict_proba(X_test ,batch_size = 32,  
verbose = 0, steps = NULL)
```

Con estas instrucciones, realizaríamos el entrenamiento por ventana haciendo uso de Keras.

## Apéndice B

# CLASIFICACIÓN GLOBAL DE PACIENTES

En el presente apéndice se explicará el procesado realizado; con la ayuda del código, para el empleo de los clasificadores a partir de los resultados obtenidos por la CNN. Principalmente, los pasos que se realizan para adaptar dichos resultados son:

- Agrupación de las ventanas pertenecientes a cada paciente.
- Obtención de los porcentajes de ventanas de caso o control.
- Selección de etiquetas y creación de conjuntos de clasificadores.

## B.1 AGRUPACIÓN DE LAS VENTANAS DE CADA PACIENTE

---

La red convolucional está diseñada para realizar la clasificación por ventanas, por lo que necesitamos agrupar todas esas ventanas; etiquetadas como *Caso* o *Control* de cada paciente.

- Buscamos en el conjunto las etiquetas que me indican a que paciente concreto pertenecen los datos obtenidos y eliminamos las etiquetas duplicadas:

```
labelClasesTrain=categorical(LabelsTrain(:,1));  
ClasesTrain=unique(labelClasesTrain);
```

- Una vez que tenemos los pacientes concretos nos centramos en el número de ventanas que se han analizado para cada paciente:

```
auxTrain=zeros(1,length(ClasesTrain))';  
  
for i=1:length(ClasesTrain)  
    auxTrain(i)=length(find(labelClasesTrain==ClasesTrain(i)));  
end
```

En el ejemplo de código se ve este proceso para el conjunto de *train*. Se repite del mismo modo para los pacientes que hemos empleado para testear.

## B.2 OBTENCIÓN DE LOS PORCENTAJES DE VENTANAS DE CASO O CONTROL

---

La red convolucional determina si una ventana pertenece a una de las dos clases, pero como se ha mencionado en los capítulos anteriores, no clasifica todas las ventanas de la misma clase para un paciente concreto. Por tanto, es necesario saber el porcentaje de ventanas de Caso y Control que la red asocia a cada paciente para llevar a cabo la clasificación.

```

for ( i =1:length ( ClasesTrain ))

    clase=ClasesTrain ( i );
    posicionSujeto=find ( labelClasesTrain==clase );
    posicionInicioSujeto=posicionSujeto ( 1 );
    posicionFinalSujeto=posicionSujeto ( length ( posicionSujeto ));
    predictedAux=predictedLabels1 ( posicionInicioSujeto : posicionFinalSujeto );
    numCasos=length ( find ( predictedAux=='Caso ' ));
    numControles=length ( find ( predictedAux=='Control ' ));
    Total=length ( predictedAux );
    porcentajeCaso=numCasos/Total ;
    porcentajeControl=numControles/Total ;
    porcentajesTrain ( i ,1)=porcentajeCaso ;
    porcentajesTrain ( i ,2)=porcentajeControl ;

end

```

## B.3 SELECCIÓN DE ETIQUETAS Y CREACIÓN DE CONJUNTOS DE CLASIFICADORES

---

En el proceso de entrenamiento de la red convolucional se ha trabajado con conjuntos exactos; a nivel de pacientes, pero de forma independiente para las señales registradas en el día y la noche. Por tanto, los clasificadores tomarán en cuenta los porcentajes correspondientes a ventanas clasificadas como *Control* (también puede tomarse el porcentaje de *Caso*) para ambos periodos de actividad, interpretando cada uno como un *feature* diferente.

- Se adaptan las etiquetas de decisión pertenecientes a las dos clases posibles con los valores de los porcentajes obtenidos para cada paciente.

```

Xt=porcentajesTrain ;
Lt = LabelsDecisionTrain ;

Xa = porcentajesTest ;
La = LabelsDecisionTest ;
LaNum=La ;
casos=find ( LaNum=='Caso ' );

```

```

controles=find (LaNum=='Control ');
clear LaNum;
LaNum( casos ,1)=1;
LaNum( controles ,1)=0;

```

- Se almacenan de forma independiente para cada paciente en función de si corresponden con el periodo diurno o nocturno.

```

if (p==1)
    XtDia=Xt ;
    LtDia=Lt ;
    XaDia=Xa ;
    LaDia=La ;
    LaNumDia=LaNum ;
elseif (p==0)
    XtNoche=Xt ;
    LtNoche=Lt ;
    XaNoche=Xa ;
    LaNoche=La ;
    LaNumNoche=LaNum ;

end

```

- Por último, se decide que porcentaje queremos que evalúe el clasificador y generamos los conjuntos definitivos para la clasificación

```

if (Caso==1)
    Xt (: ,1)= XtDia (: ,1);
    Xt (: ,2)= XtNoche (: ,1);

    Lt=LtDia ;

    Xa (: ,1)= XaDia (: ,1);
    Xa (: ,2)= XaNoche (: ,1);

    La=LaDia ;

    LaNum (: ,1)= LaNumDia (: ,1);

elseif (Control==1)
    Xt (: ,1)= XtDia (: ,2);
    Xt (: ,2)= XtNoche (: ,2);

    Lt=LtDia ;

    Xa (: ,1)= XaDia (: ,2);
    Xa (: ,2)= XaNoche (: ,2);

```

```
La=LaDia ;  
LaNum(:,1)=LaNumDia(:,1);  
end
```