

UNIVERSIDAD NACIONAL DEL LITORAL



DOCTORADO EN INGENIERÍA

# Procesamiento de señales relacionadas con patologías del sueño

Ramiro Casal

FICH

FACULTAD DE INGENIERÍA Y CIENCIAS HÍDRICAS

INTEC

INSTITUTO DE DESARROLLO TECNOLÓGICO PARA LA INDUSTRIA QUÍMICA

CIMEC

CENTRO DE INVESTIGACIÓN DE MÉTODOS COMPUTACIONALES

*sinc(i)*

INSTITUTO DE INVESTIGACIÓN EN SEÑALES, SISTEMAS E INTELIGENCIA  
COMPUTACIONAL



UNIVERSIDAD NACIONAL DEL LITORAL  
Facultad de Ingeniería y Ciencias Hídricas  
Instituto de Desarrollo Tecnológico para la Industria Química  
Centro de Investigación de Métodos Computacionales  
Instituto de Investigación en Señales, Sistemas e Inteligencia Computacional

# PROCESAMIENTO DE SEÑALES RELACIONADAS CON PATOLOGÍAS DEL SUEÑO

**Ramiro Casal**

Tesis remitida al Comité Académico del Doctorado  
como parte de los requisitos para la obtención  
del grado de  
DOCTOR EN INGENIERIA  
Mención Inteligencia Computacional, Señales y Sistemas  
de la  
UNIVERSIDAD NACIONAL DEL LITORAL

**2020**

Comisión de Posgrado, Facultad de Ingeniería y Ciencias Hídricas, Ciudad Universitaria,  
Paraje "El Pozo", S3000, Santa Fe, Argentina.



UNIVERSIDAD NACIONAL DEL LITORAL  
Facultad de Ingeniería y Ciencias Hídricas  
Instituto de Desarrollo Tecnológico para la Industria Química  
Centro de Investigación de Métodos Computacionales  
Instituto de Investigaciones en Señales, Sistemas e Inteligencia Computacional

## **PROCESAMIENTO DE SEÑALES RELACIONADAS CON PATOLOGÍAS DEL SUEÑO**

**Ramiro Casal**

**Lugar de Trabajo:**

LSyDNL  
Laboratorio de Señales y Dinámicas No Lineales  
Facultad de Ingeniería  
Universidad Nacional de Entre Ríos

**Director:**

Gastón Schlotthauer

IBB-FI-UNER-CONICET

**Co-director:**

Leandro E. Di Persia

sinc(i)-FICH-UNL-CONICET

**Jurado Evaluador:**

Dr. Marcelo Risk  
Dr. Franco Pesanna  
Dr. Pablo Granitto

IMTIB- CONICET  
UTN.Buenos Aires  
CIFASIS, UNR

**2020**

# Índice general

Índice general	I
Índice de figuras	III
Índice de tablas	IV
Resumen	VII
<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	1
1.2. Objetivos . . . . .	5
1.3. Justificación del encuadre de la tesis . . . . .	5
1.4. Organización del documento . . . . .	5
<b>2. Marco Teórico</b>	<b>7</b>
2.1. Síndrome de apnea obstructiva del sueño . . . . .	7
2.2. Métodos de diagnóstico . . . . .	9
2.3. Señales de oximetría de pulso . . . . .	11
2.4. Etapas de sueño según AASM . . . . .	11
2.4.1. Automatización en la clasificación de las fases del sueño . . . . .	13
2.5. Eventos respiratorios según AASM . . . . .	15
2.6. Antecedentes . . . . .	16
2.6.1. Clasificación automática de fases del sueño . . . . .	16
2.6.2. Cribado de apnea del sueño . . . . .	19
2.7. Base de datos . . . . .	21
<b>3. Métodos clásicos</b>	<b>24</b>
3.1. Metodología . . . . .	25
3.1.1. Preprocesamiento . . . . .	26
3.1.2. Extracción de características . . . . .	26
3.1.3. Selección de características y clasificación . . . . .	33
3.1.4. Algoritmos propuestos . . . . .	36
3.2. Resultados . . . . .	36
3.2.1. Selección de hiperparámetros . . . . .	36
3.2.2. Selección de características . . . . .	37
3.2.3. Desempeño de los algoritmos en la etapa de evaluación . . . . .	42
3.3. Discusión . . . . .	44
3.3.1. Ventajas y dificultades de la oximetría de pulso . . . . .	44
3.3.2. Longitud de segmento $L$ . . . . .	44
3.3.3. Especificaciones técnicas de las señales . . . . .	46
3.3.4. Comparación con estado del arte . . . . .	47

3.3.5. Diseño del método y utilización subsiguiente . . . . .	47
3.4. Conclusiones . . . . .	47
<b>4. Redes neuronales recurrentes</b>	<b>49</b>
4.1. Metodología . . . . .	49
4.1.1. Preprocesamiento . . . . .	51
4.1.2. Redes neuronales recurrentes: LSTM y GRU . . . . .	52
4.1.3. Capa <i>softmax</i> . . . . .	53
4.1.4. Voto mayoritario . . . . .	54
4.2. Resultados . . . . .	54
4.2.1. Medidas de desempeño . . . . .	54
4.2.2. Parámetros de la red y del optimizador . . . . .	54
4.2.3. Entrenamiento de la red . . . . .	55
4.2.4. Desempeño en el subconjunto de testeo . . . . .	56
4.2.5. Clasificación de fases del sueño según AASM . . . . .	57
4.3. Discusión . . . . .	59
4.3.1. Incorporación de SpO <sub>2</sub> . . . . .	61
4.3.2. Comparación con estado del arte . . . . .	61
4.4. Conclusiones . . . . .	63
<b>5. Mecanismos de atención</b>	<b>64</b>
5.1. Metodología . . . . .	65
5.1.1. Arquitectura propuesta . . . . .	65
5.2. Resultados . . . . .	70
5.2.1. Detalles de implementación . . . . .	71
5.2.2. Desempeño en el subconjunto de testeo . . . . .	73
5.2.3. Clasificación de fases del sueño según AASM . . . . .	75
5.3. Discusión . . . . .	76
5.3.1. RNN y redes basadas en mecanismos de atención . . . . .	77
5.3.2. Comparación con el estado del arte . . . . .	78
5.3.3. Clasificación de las fases del sueño según AASM . . . . .	80
5.4. Conclusiones . . . . .	82
<b>6. Acercamiento a trabajos futuros</b>	<b>84</b>
6.1. Incorporación del TTS a un método para diagnóstico de AHI . . . . .	84
6.2. Algoritmo propuesto por Schlotthauer y cols. basado en EMD . . . . .	85
6.3. Resultados . . . . .	86
6.4. Discusión . . . . .	88
6.5. Conclusiones . . . . .	89
<b>7. Conclusiones</b>	<b>90</b>
<b>Bibliografía</b>	<b>93</b>

# Índice de figuras

2.1. Hipnograma y HR . . . . .	14
2.2. Eventos de apnea e hipopnea . . . . .	15
2.3. Gráfico de evento de apnea obstructiva. . . . .	20
3.1. Esquema del algoritmo basado en métodos clásicos . . . . .	25
3.2. Variación del desempeño con los métodos de selección de características . . . . .	38
3.3. Total de tiempo despierto/dormido y longitud del segmento $L$ . . . . .	45
4.1. Esquema del algoritmo basado en RNN . . . . .	50
4.2. Ordenamiento de las secuencias para los <i>mini-batch</i> . . . . .	55
4.3. Entrenamiento del sistema basado en RNN y selección del modelo . . . . .	56
4.4. Hipnogramas simplificados . . . . .	58
4.5. Hipnogramas según AASM . . . . .	60
5.1. Arquitectura para aprendizaje de la representación de la red . . . . .	68
5.2. Arquitectura para modelización de la secuencia de la red . . . . .	70
5.3. Hipnogramas simplificados . . . . .	75
5.4. Hipnogramas según AASM . . . . .	77
6.1. Algoritmo desarrollado por Schlotthauer y cols. . . . .	85

# Índice de tablas

2.1. Índice Kappa: magnitud y significado. . . . .	14
2.2. Estado del arte para clasificación de fases del sueño . . . . .	19
2.3. Estado de arte para clasificación de eventos de apnea/hipopnea. . . . .	21
2.4. Características de la SHHS 1 . . . . .	22
3.1. Características seleccionadas utilizando FS 1. . . . .	39
3.2. Características seleccionadas utilizando FS 2. . . . .	40
3.3. Características seleccionadas utilizando FS 3. . . . .	41
3.4. Matriz de confusión con los resultados obtenidos en la etapa de evaluación . . . . .	42
3.6. Desempeño de los algoritmos para los diferentes valores de $L$ considerados . . . . .	43
3.7. Comparación con estado del arte. . . . .	46
4.1. Desempeño de las redes neuronales recurrentes desarrolladas . . . . .	57
4.2. Desempeño y comparación con la literatura según AASM . . . . .	59
4.3. Desempeño y comparación con la literatura al clasificar en W, liviano, profundo y REM. . . . .	59
4.4. Desempeño y comparación con la literatura al clasificar en W, REM y no REM . . . . .	60
4.5. Comparación con estado del arte . . . . .	62
5.1. Desempeño de las redes basadas en <i>transformers</i> . . . . .	74
5.2. Desempeños y comparación con la literatura según AASM . . . . .	76
5.3. Desempeño y comparación con la literatura al clasificar en W, liviano, profundo y REM . . . . .	76
5.4. Desempeño y comparación con la literatura al clasificar en W, REM y no REM . . . . .	77
5.5. Comparación con el estado del arte . . . . .	79
6.1. Resultados para cribado de SAOS . . . . .	87
6.2. Corrección de resultados para cribado de SAOS . . . . .	87

# Lista de acrónimos

AASM	Academia Americana de Medicina del Sueño ( <i>American Academy of Sleep Medicine</i> )
Acc	Exactitud
AR	Modelo <b>autorregresivo</b> ( <i>autorregressive model</i> )
ApEn	Entropía aproximada ( <i>Approximate entropy</i> )
CNN	Redes neuronales convolucionales ( <i>convolutional neural network</i> )
CV	Validación cruzada ( <i>cross-validation</i> )
CWT	Transformada ondita continua ( <i>continuous wavelet transform</i> )
DispEn	Entropía de dispersión ( <i>dispersion entropy</i> )
ECG	Electrocardiografía
EEG	Electroencefalografía
EMD	Descomposición empírica en modos ( <i>empiric mode decomposition</i> )
EMG	Electromiografía
EOG	Electrooculografía
FFS	Selección de características en avance ( <i>forward feature selection</i> )
<b>FNN</b>	Red neuronal prealimentada ( <i>feed forward network</i> )
FN	Falsos negativos
FP	Falsos positivos
FT	Transformada de Fourier ( <i>Fourier transform</i> )
FuzEn	Entropía difusa ( <i>fuzzy entropy</i> )
GRU	Unidades recurrentes con compuertas ( <i>Gated recurrent units</i> )
HR	Frecuencia cardíaca, <i>Heart rate</i>
IAH	Índice de apnea/hipopnea
ICSD	Clasificación Internacional de Patologías del Sueño ( <i>International Classification of Sleep-Disorders</i> )
IDH	Índice de desaturaciones por hora
$\kappa$	Coficiente Kappa de Cohen
LZ	Complejidad de Lempel-Ziv
LSTM	Memoria de largo y corto plazo ( <i>long-short term memories</i> )
NPV	Valor predictivo negativo ( <i>negative predictive value</i> )
PermEn	Entropía de permutación ( <i>permutation entropy</i> )
PPV	Valor predictivo positivo ( <i>positive predictive value</i> )
PSD	Densidad espectral de potencia ( <i>power <b>espectral</b> density</i> )
PSG	Polisomnografía
ReLU	Unidad lineal rectificadora ( <i>rectified linear unit</i> )
REM	Fase de sueño de movimientos oculares rápidos
RF	Bosques aleatorios ( <i>random forest</i> )
RNN	Redes neuronales recurrentes ( <i>recurrent neural network</i> )
SampEn	Entropía muestral ( <i>sample entropy</i> )
SaO <sub>2</sub>	Saturación de oxígeno en sangre arterial



SAOS	Síndrome de apnea obstructiva del sueño
SE	Entropía espectral ( <i>spectral entropy</i> )
Se	Sensibilidad
SHHS	Base de datos <i>Sleep Heart Health Study</i>
Sp	Especificidad
SpO <sub>2</sub>	Saturación periférica de oxígeno en sangre
SVD	Descomposición de valores singulares ( <i>singular value decomposition</i> )
SVM	Máquinas de soporte vectorial ( <i>support vector machine</i> )
TCN	Redes convolucionales temporales ( <i>Temporal convolutional networks</i> )
TE	Tiempo de entrenamiento en minutos (TE)
TSMRE	Entropía de Renyi multirresolución en el dominio tiempo-escala ( <i>time-scale multiresolution Renyi entropy</i> )
TTR	Tiempo total del registro
TTS	Tiempo total de sueño
VN	Verdaderos negativos
VP	Verdaderos positivos
W	Fase de vigilia

# Resumen

Una gran porción de la población humana adulta sufre de algún trastorno del sueño, siendo el síndrome de apnea obstructiva del sueño (SAOS) uno de los que tiene mayor prevalencia. El procedimiento de diagnóstico estándar para determinar la presencia de patologías del sueño y cuantificar su severidad es la polisomnografía (PSG), un estudio en el que se registran numerosas variables fisiológicas durante el sueño nocturno en un lugar especializado para tal fin, llamado laboratorio del sueño. Este procedimiento resulta costoso, complejo y poco accesible para la población, lo que conduce a un subdiagnóstico generalizado de este tipo de patologías. Por estas razones, existe un creciente interés en reducir la necesidad de estudios de PSG para analizar el sueño del paciente, el cual se ha materializado en el desarrollo de varias propuestas cuyo objetivo es realizar un cribado de las patologías del sueño, teniendo el doble objetivo de permitir obtener un conocimiento general de su prevalencia en la población y de decidir qué porción de los sujetos atendidos necesitan realmente un estudio más complejo para ser debidamente diagnosticados.

Entre las propuestas más prometedoras se encuentra la oximetría de pulso, destacándose por su bajo costo, la simpleza de su instrumentación y su accesibilidad. A partir de este dispositivo se pueden obtener estimaciones de la frecuencia cardíaca y de la saturación periférica de oxígeno en sangre que resultan particularmente útiles para numerosas patologías del sueño, teniendo además la ventaja de ser un equipamiento que interfiere mínimamente con el sueño del sujeto monitoreado. Un indicador muy importantes para evaluar en los estudios de sueño es el tiempo total de sueño (TTS). Ninguno de los métodos de cribado para patologías del sueño tienen capacidad de obtener el TTS, el cual, en estudios más complejos y en el estado del arte en general, es estimado principalmente mediante electroencefalografía. Abordando este inconveniente de los métodos de cribado, el objetivo de esta tesis es desarrollar algoritmos que permitan estimar el TTS utilizando oximetría de pulso. Varios estudios sugieren que los cambios en el sistema nervioso central durante la noche se ven reflejados en modificaciones en la dinámica de varias señales fisiológicas, entre ellas las señales cardíacas. Dado que el oxímetro de pulso provee una estimación de la frecuencia cardíaca, a través de ella se pueden registrar estos cambios y, con un procesamiento adecuado, clasificar el estado del sueño en despierto y dormido. Esto, a su vez, puede complejizarse de forma que los algoritmos ya no sólo sean capaces de diferenciar entre los estados de despierto y dormido, sino que puedan clasificar automáticamente las fases del sueño en su totalidad, las cuales proveen información fundamental en los estudios del sueño. Además, esta técnica presenta una ventaja adicional para el cribado del SAOS, una patología caracterizada por la interrupción de las vías aéreas y, asociado a ello, un descenso en el nivel de oxígeno en sangre que puede ser detectado indirectamente por el oxímetro a través de la estimación de la saturación periférica de oxígeno en sangre. Una de los indicadores más importantes para la severidad del SAOS es el índice de apnea/hipopnea (AHI, por sus siglas en inglés) que mide la frecuencia de estas interrupciones respecto al TTS. Existen varios métodos de cribado capaces de detectar estas desaturaciones, sin embargo ninguno de ellos tiene acceso al TTS.

La primera contribución de esta tesis es un algoritmo basado en métodos clásicos que

permite clasificar el estado del sueño en despierto y dormido utilizando la señal de frecuencia cardíaca provista por un oxímetro de pulso comercial. En este algoritmo se extrajeron 70 características mediante medidas de entropía y de complejidad, métodos para análisis del dominio de la frecuencia y del dominio tiempo-escala, así también como estadísticos clásicos. Sobre el espacio de características obtenido se realizó una reducción dimensional mediante métodos de selección. Luego, las características seleccionadas se utilizaron en dos clasificadores: uno basado en máquinas de soporte vectorial y otro basado en bosques aleatorios. Los resultados obtenidos, aunque alentadores por ser un primer trabajo, pusieron en evidencia la necesidad de desarrollar un método que tome en consideración la información del contexto para clasificar los segmentos de frecuencia cardíaca.

De esta forma, y como segunda contribución, se desarrolló un método que realiza la clasificación en forma secuencial. Es decir, que la clasificación de un segmento dado dependerá de la información actual, pero también se relacionará con la información pasada y futura. La introducción de esta propiedad se logró mediante la creación de un algoritmo basado en unidades recurrentes con compuertas (GRU, por sus siglas en inglés). La arquitectura consiste en dos capas de GRU bidireccionales que clasifican el sueño en despierto y dormido. A diferencia del caso anterior, se utilizó como entrada para este algoritmo tanto la frecuencia cardíaca como la saturación de oxígeno en sangre, ambas provenientes de un oxímetro de pulso. Los resultados obtenidos por la red neuronal propuesta son comparables con algoritmos del estado del arte que utilizan señales muchísimo más informativas, con la sola excepción de los algoritmos basados en electroencefalografía. Para este algoritmo, además, presentamos una variación que permite una mayor discriminación de las fases del sueño, según se establece en los criterios de la Academia Americana de Medicina del Sueño (AASM), ampliamente aceptados en todo el mundo. Con este resultado, superamos el estado del arte de todos los métodos para cribado automático de las fases del sueño.

Dada su naturaleza secuencial, las GRU tienen una serie de desventajas en lo que respecta al cálculo paralelo necesario para entrenar el algoritmo, lo que conduce a un aprendizaje muy lento y, a su vez, genera una gran demanda en la memoria computacional. Por esta razón, la tercera y última contribución de esta tesis es un método basado en mecanismos de atención que permiten superar esta desventaja, obteniendo entrenamientos hasta 10 veces más rápidos y sin una pérdida significativa de desempeño. La arquitectura de este algoritmo se compone de redes convolucionales temporales (del inglés TCN), capaces de extraer en forma automática características de la señal, que luego son procesadas mediante un tipo especial de redes neuronales que únicamente utilizan mecanismos de atención llamados *transformers*. A partir de estos desarrollos observamos que las redes de atención también fueron capaces de modelar las dependencias temporales presentes entre los estados de sueño despierto y dormido.

El logro de una estimación confiable del TTS y de una correcta clasificación de las fases del sueño, tanto en su versión simplificada (despierto y dormido) como en su versión completa (fases del sueño según AASM), constituyen el aporte principal de esta tesis doctoral. Como cierre de esta tesis y para establecer los futuros lineamientos que continuarán estos desarrollos, realizamos un análisis preliminar de la variación del desempeño de un algoritmo de cribado del SAOS al incorporar el TTS estimado por nuestros métodos para calcular la frecuencia de los eventos respiratorios. Los resultados sugieren una leve mejora respecto a utilizar una aproximación al TTS mediante el tiempo total del registro y alientan a seguir trabajando en esta línea.

# Capítulo 1

## Introducción

### 1.1. Motivación

El sueño cumple un rol fundamental en los humanos, influenciando numerosos procesos vitales como la función del sistema inmune y del sistema endócrino, la restauración del sistema osteoartromuscular, la consolidación de mecanismos de memoria, la preservación de la función cognitiva, entre otras [1]. Los trastornos del sueño incluyen más de ochenta patologías muy frecuentes tanto en adultos como en niños, afectando a una importante porción de la población mundial. La prevalencia de este tipo de patologías reportada por diferentes estudios epidemiológicos varía significativamente, principalmente debido a las diferentes definiciones de insomnio consideradas [2]. Además, a pesar de las discrepancias respecto a la magnitud, todos los estudios consultados coinciden en que existe una subestimación general de las patologías del sueño [3–5]. Se estima que en Estados Unidos se ve afectado entre el 35 y el 40 % de la población adulta, mientras que en Argentina –si bien no existe una cifra oficial– especialistas de la Asociación Argentina de Medicina del Sueño calculan que aproximadamente un 40 % de la población urbana sufre diversos trastornos del sueño.

Como resultado de su alta prevalencia, los trastornos del sueño se constituyen como una causa importante de morbilidad y mortalidad. Además, debido a las complicaciones severas y las enfermedades concomitantes en los casos no tratados, los costos implicados son enormes tanto para el individuo que lo padece como para la sociedad [6]. Por todo esto, el interés de la comunidad científica en estas patologías ha crecido en gran medida en los últimos años [7].

Entre las patologías del sueño con mayor prevalencia se destaca el síndrome de apnea obstructiva del sueño (SAOS)<sup>1</sup> [9]. Esta condición se caracteriza por pausas repetidas en la respiración durante el sueño causadas por la obstrucción intermitente de las vías aéreas producto de un colapso faríngeo. Estas obstrucciones de las vías aéreas superiores pueden producir una reducción parcial –denominadas hipopneas– o una interrupción total –apneas– del flujo de aire inspirado, ocasionando un incremento en la frecuencia de los despertares y los microdespertares<sup>2</sup>, fragmentación del sueño y, consecuentemente, excesiva somnolencia diurna [10–12]. Además, este síndrome está asociado a un incremento del riesgo de sufrir

---

<sup>1</sup>Existe cierta confusión dado que términos muy similares muchas veces son utilizados como sinónimos en la academia. Por ejemplo «síndrome de apnea obstructiva del sueño», «síndrome de apnea/hipopnea obstructiva del sueño», «síndrome de apnea del sueño». En este trabajo vamos a utilizar el primer término, el cual está ampliamente utilizado en la bibliografía más importante de referencia confeccionada por la Academia Americana de Medicina del Sueño (*American Academy of Sleep Medicine*, AASM) [8]. Creemos que explicitar el término «obstrutivo» evita la confusión respecto a si se incluyen o no las apneas centrales, que poseen una etiología diferente.

<sup>2</sup>Son cambios abruptos en el encefalograma que indicarían un despertar. Sin embargo, el sujeto no suele ser conciente de esto.

hipertensión sistémica y pulmonar, arritmias nocturnas, falla ventricular e infarto, problemas cognitivos y muerte súbita [13]. Las potenciales consecuencias sociales y económicas de este síndrome –accidentes, incremento de la morbilidad e improductividad, entre otras tantas– lo convierten en uno de los principales problemas de salud pública del mundo y, por esto, son la razón principal del constante crecimiento de interés para abordar esta temática, no sólo en el área clínica sino también desde la investigación académica. Para dar un ejemplo de un país que cuenta con datos al respecto, en Australia se estima un costo anual de 8 mil millones de dólares en gastos médicos debido a la falta de tratamiento [14]. Esta situación se agrava si consideramos que el número de pacientes diagnosticados y tratados por este síndrome se ha incrementado drásticamente en los últimos años [14, 15]. Una vez diagnosticado, el SAOS responde en forma satisfactoria al tratamiento, presentando un rápido alivio de los síntomas [9].

El procedimiento de diagnóstico estándar<sup>3</sup> para determinar la presencia de patologías del sueño en general –y también para el diagnóstico de SAOS en particular– es la polisomnografía (PSG), la cual consiste en el registro simultáneo de varias señales fisiológicas tales como electroencefalografía (EEG), electrocardiografía (ECG), electromiografía (EMG), esfuerzo respiratorio, flujo aéreo, saturación de oxígeno en sangre, electrooculografía (EOG), entre otras. La realización del estudio es supervisada por un técnico y es llevada a cabo en un laboratorio de sueño acondicionado especialmente para ese fin. Luego, los registros obtenidos son analizados y etiquetados por especialistas médicos con el objetivo de determinar la severidad de la patología [8]. En general, este análisis se realiza manualmente, aunque se utiliza un *software* para asistir la labor del técnico. Las señales adquiridas mediante la PSG son etiquetadas de acuerdo con dos reconocidos manuales. El primero fue confeccionado en 1968 por Rechtschaffen y Kales [16], por lo que es ampliamente conocido como (R&K), mientras que el segundo pertenece a la AASM [8] y tuvo origen en 2007 con el objetivo de introducir algunas modificaciones y actualizaciones en los criterios tradicionales del R&K. Ambos manuales representan una guía para identificar y anotar una serie de eventos que permiten caracterizar el sueño, tales como los despertares, movimientos musculares, etapas del sueño, eventos cardíacos y eventos respiratorios.

Estas características hacen de la PSG un estudio complejo y con altos costos asociados, por lo que presenta severas limitaciones en su accesibilidad, generando largas listas de espera. Además, muchos pacientes se muestran renuentes a la realización del estudio por tener que pasar la noche monitoreados en un centro de salud. Asimismo, los pacientes que acceden al estudio muchas veces tienen dificultades para conciliar el sueño [17]. Por estas razones, existe un interés creciente en reducir la necesidad de recurrir a los estudios de PSG en laboratorios del sueño para estudiar al paciente. Este creciente interés ha sido materializado en varias propuestas que tienen por objetivo realizar un cribado de las patologías del sueño que permita obtener un conocimiento general de la prevalencia de estas enfermedades en la población, así como también decidir qué porción de los sujetos atendidos necesitan de un estudio más complejo para ser debidamente diagnosticados. En esta dirección, se han propuesto métodos alternativos de diagnóstico que utilizan sonidos cardíacos y respiratorios [18], oximetría de pulso [19–23], ECG [24], presión aérea nasal [25] y combinaciones de varias señales [26]. La oximetría de pulso destaca entre las alternativas por su bajo costo y alta accesibilidad, así como por la simpleza de su instrumentación y uso, constituyéndose como una opción ideal para el cribado de patologías del sueño.

Además de estas ventajas, la oximetría de pulso posee una utilidad inherente a su funcionamiento que es esencial para el diagnóstico del SAOS. Como ya hemos comentado, este síndrome se caracteriza por presentar eventos repetidos de obstrucciones totales o parciales

---

<sup>3</sup>Utilizamos la expresión procedimiento de diagnóstico estándar como traducción del inglés del muy usado término *gold-standard*.

de las vías aéreas superiores durante el sueño. Permítasenos en este capítulo introductorio dejar de lado las definiciones técnicas, tanto de los eventos de apnea/hipopnea como de otros indicadores que serán comentados en el próximo capítulo. Con el objetivo de introducir el problema abordado en esta tesis diremos, por ahora, que estas obstrucciones tienen un correlato con el nivel de saturación de oxígeno en sangre registrado por el oxímetro de pulso. Los eventos de apnea e hipopnea generan una desaturación característica del oxígeno en sangre, presentando un decremento lento de la saturación mientras el evento de obstrucción transcurre, para luego realizar una recuperación más rápida una vez que el evento finaliza. De esta manera, si logramos identificar y estimar la frecuencia de estas desaturaciones durante el sueño podremos obtener una estimación indirecta de la frecuencia de los eventos de apnea/hipopnea del paciente. Este indicador, llamado índice de apnea/hipopnea (IAH), es la medida más importante que permite cuantificar la severidad del SAOS. Para calcular el IAH necesitaremos además calcular el tiempo total de sueño (TTS) del sujeto en estudio.

Se han llevado a cabo varios trabajos con el objetivo de detectar las desaturaciones en la señal de oximetría [19–23]. En estos trabajos se busca estimar el índice de desaturaciones por hora (IDH) como una aproximación al IAH, prescindiendo de señales de EEG para evaluar si el paciente está o no dormido. No obstante, en algunos de ellos se reporta el IDH dividiendo el número de desaturaciones detectadas por el TTS calculado a través del EEG, al que supuestamente no se tendría acceso. En otros casos, en cambio, se utiliza en su lugar el tiempo total del registro (TTR), lo que introduce un sesgo importante.

Respecto a la estimación del TTS, puede obtenerse una estimación de éste como resultado derivado de clasificar automáticamente la señal en dos clases: despierto y dormido. Esta problemática puede verse como una versión simplificada de una aplicación que tiene una larga historia tanto en la comunidad científica como en su correspondiente aplicación clínica. Hablamos de la clasificación automática de las etapas del sueño. Las etapas del sueño están tipificadas en los manuales de R&K [16] y AASM [8] y permiten caracterizar el sueño. El manual R&K define las siguientes etapas del sueño: «despierto» (W), dos etapas de «sueño liviano» (N1 y N2), dos etapas de «sueño profundo» (N3 y N4) y una etapa conocida como «movimientos oculares rápidos» (por sus siglas en inglés, REM). Estas etapas se diferencian entre sí en base a formas de onda características que pueden encontrarse principalmente en las señales de EEG, EMG y EOG. Recientemente, con la publicación de la AASM, las etapas N3 y N4 han sido unificadas en una única etapa correspondiente al sueño profundo, que simplemente llamaremos N3<sup>4</sup>.

De acuerdo con ambos manuales, los registros obtenidos mediante PSG son segmentados en tramos de 30 segundos de longitud para luego ser clasificados manualmente por expertos que identifican patrones característicos que se encuentran principalmente en las señales de EEG, EMG y EOG. Dado que este proceso es muy susceptible a errores humanos, por lo que muestra una enorme variabilidad interprofesional [27] y, además, resulta sumamente tedioso, se han realizado muchos esfuerzos en la comunidad científica con el objetivo de automatizar la clasificación de las etapas del sueño. La clasificación automática de las etapas del sueño es realizada principalmente utilizando la señal de EEG, aunque muchas veces se extrae información complementaria de otras señales [28–30]. Varios estudios han mostrado que la señal de EEG es suficientemente confiable para realizar la clasificación [31]. Sin embargo, debido a razones análogas a las mencionadas anteriormente que motivaron el desarrollo de trabajos para obtener métodos alternativos de diagnóstico a la PSG, existe una creciente demanda de dispositivos ambulatorios que permitan obtener una clasificación de las etapas del sueño sin la utilización de EEG, cuya instrumentación es más compleja y se ve muy afectada por el ruido. Esto puede ser de gran utilidad para la obtención de cribados masivos que permitan evaluar la prevalencia de patologías del sueño en grandes poblaciones, evitando la sobrecar-

---

<sup>4</sup>En inglés esta etapa es conocida como *slow wave sleep*.

ga del sistema de salud y, consecuentemente, las largas listas de espera y el subdiagnóstico generalizado que presentan estas patologías [4].

Las propuestas más extendidas para cribado automático de las etapas del sueño utilizan señales cardíacas. Varios estudios han encontrado que la frecuencia cardíaca promedio es mayor durante la vigilia, decreciendo continuamente desde las etapas de sueño liviano hacia las de sueño profundo, para luego incrementar nuevamente en la etapa de sueño REM. Paralelamente, la variabilidad de la frecuencia cardíaca es mayor durante las etapas de vigilia y sueño REM que durante las etapas de sueño liviano y profundo [32, 33]. Estos hallazgos han establecido las bases para el desarrollo de algoritmos para clasificar las etapas del sueño utilizando señales cardíacas, principalmente ECG [34–36]. Sin embargo, recientemente se han publicado artículos que, en lugar del ECG, utilizan señales de fotopletimografía (por sus siglas en inglés, PPG) obtenidas mediante un oxímetro de pulso, que como ya se ha mencionado, son señales más portables, simples y cuyo costo asociado es significativamente menor que el ECG [37, 38]. En estos trabajos, las etapas del sueño usualmente son agrupadas con el objetivo de obtener un resultado primario que permita estimar rápidamente la condición del paciente para luego ser transferido a un estudio completo si es necesario. La situación más común es clasificar las etapas en W, REM y no-REM (agrupando N1, N2 y N3) o directamente en W y dormido (agrupando etapas REM y no-REM). Estos trabajos serán abordados en detalle en los capítulos siguientes como parte del estado del arte.

Con esta presentación, el lector ya podrá notar que es factible obtener una clasificación simple de las etapas del sueño a partir de la cual lograr una estimación del TTS y, de esta manera, complementar las estimaciones del IAH obtenidas mediante oximetría de pulso, abordando desde una nueva perspectiva el cribado del SAOS.

En esta tesis se desarrollarán técnicas alternativas a la PSG que permitan un apoyo al diagnóstico rápido, eficiente y económico de las patologías del sueño. Estas herramientas estarán basadas en procesamiento de señales, reconocimiento de patrones y aprendizaje maquina, poniendo especial énfasis en el carácter ambulatorio del método.

En sintonía con lo que ha sido presentado hasta ahora en esta introducción, el abordaje propuesto para el cribado de patologías del sueño en general –y del SAOS en particular– puede dividirse en dos líneas de trabajo complementarias: la estimación del TTS y la estimación del IAH, utilizando señales de oximetría de pulso.

La primera de ellas está motivada por las desventajas comentadas en los párrafos anteriores presentes en los dispositivos de cribado de los trastornos del sueño –principalmente en el caso del SAOS–. En este caso se propondrá realizar una clasificación del estado del sueño en despierto y dormido utilizando únicamente señales obtenidas a partir de un oxímetro de pulso comercial, es decir, la frecuencia cardíaca y la saturación de oxígeno en sangre. Una buena clasificación en despierto y dormido nos permitirá obtener una estimación precisa del TTS.

Como segunda propuesta se desarrollarán algoritmos que permitan detectar los eventos de apnea e hipopnea indirectamente. La identificación y conteo de estos eventos, en conjunto con el TTS obtenido como resultado anterior, permitirá la estimación automática de un índice que se comporte de manera similar al IAH, pero utilizando únicamente un oxímetro de pulso comercial. El abordaje de esta última aplicación se encuentra en un estadio inicial para este equipo de trabajo, por lo que únicamente se presentarán algunos resultados preliminares, quedando como objeto principal para futuros desarrollos que servirán de continuación a esta tesis.

Con el objetivo de extender la aplicabilidad de los métodos que se desarrollarán para la clasificación en despierto y dormido, se realizarán modificaciones en los algoritmos para lograr discriminar las etapas del sueño según los criterios estándares publicados por la AASM. La clasificación automática y confiable de las etapas del sueño utilizando únicamente señales

obtenidas mediante oximetría de pulso será útil no sólo en el ámbito clínico, sino que estas tecnologías pueden también ser adaptadas fácilmente para dispositivos portátiles y ser utilizadas para monitoreo personal de la salud. Actualmente, existen dispositivos comerciales que proveen medidas relacionadas con el sueño, aunque pocos de ellos han sido validados [39]. Es una gran ventaja lograr desembarcar en el ámbito de los dispositivos de consumo con un algoritmo que originalmente fue pensado y diseñado con los parámetros de confiabilidad necesarios en el ámbito clínico.

## 1.2. Objetivos

Ya expuestos los antecedentes que sirven de motivación a este trabajo, resta explicitar los objetivos que serán abordados en esta tesis, enmarcada en el Doctorado en Ingeniería dictado por la Facultad de Ingeniería y Ciencias Hídricas de la Universidad Nacional del Litoral (FICH-UNL), en su mención Inteligencia Computacional, Señales y Sistemas.

Los *objetivos generales* de este proyecto son:

- Aportar al desarrollo de técnicas de apoyo al diagnóstico y de toma de decisiones en problemas biomédicos utilizando herramientas de procesamiento de señales y de aprendizaje maquina.
- Contribuir a la formación de grupos de trabajo multidisciplinarios en la temática de este plan de trabajo.

A su vez, los *objetivos específicos* contemplados en esta tesis son:

- Contribuir al desarrollo de métodos que permitan la estimación de indicadores para evaluar patologías del sueño sin la necesidad de realizar estudios de polisomnografía.
- Aplicar técnicas de procesamiento de señales y aprendizaje maquina al desarrollo de algoritmos para clasificación de las etapas de sueño utilizando señales obtenidas mediante oximetría de pulso.
- Evaluar el desempeño de los métodos desarrollados en bases de datos.

## 1.3. Justificación del encuadre de la tesis en la mención «Inteligencia Computacional, Señales y Sistemas»

En la presente propuesta de tesis se plantea ampliar y mejorar los métodos de cribado para patologías del sueño. Para ello se utilizarán señales obtenidas a partir de oximetría de pulso. Estas señales serán procesadas mediante herramientas de aprendizaje maquina y procesamiento de señales. Los enfoques abordados contemplarán la utilización de herramientas clásicas y no clásicas, permitiendo comparar los diferentes desempeños obtenidos. Todas estas temáticas se encuadran en las previstas en la mención de Inteligencia Computacional, Señales y Sistemas del Doctorado en Ingeniería de la FICH-UNL.

## 1.4. Organización del documento

Esta tesis de doctorado está estructurada en 7 capítulos. Este capítulo ha servido como una introducción de la problemática abordada, permitiéndonos realizar una explicación de los motivos que nos alentaron para que nuestro trabajo tomase esta dirección.



En el próximo capítulo se proveerá al lector de las herramientas teóricas, las definiciones técnicas y los antecedentes en el área que creemos necesarios para lograr una comprensión íntegra de los desarrollos realizados en esta tesis, que serán motivo de los tres capítulos siguientes. En ese capítulo, entonces, presentaremos una descripción general del problema abordado, considerando su relación con la fisiología. Se explicará la técnica de PSG y cómo se diagnostican las patologías del sueño, poniendo especial énfasis en el SAOS y en la clasificación de las fases del sueño. Se expondrán las principales razones por las que creemos necesario una mayor inversión en la investigación y desarrollo de técnicas de cribado para las patologías. Finalmente expondremos los principales trabajos que conforman el estado del arte.

En el capítulo 3 detallaremos nuestro primer algoritmo, basado en métodos clásicos, para clasificar las fases del sueño en despierto o dormido utilizando la señal de frecuencia cardíaca provista por un oxímetro de pulso comercial. En este algoritmo se extrajeron varias características utilizando medidas de entropía y de complejidad, métodos del dominio de la frecuencia y del dominio tiempo-escala, así como también estadísticos clásicos. Sobre el espacio de características obtenido, se realizó una reducción dimensional y clasificación utilizando diferentes esquemas, todos ellos basados en técnicas clásicas de aprendizaje maquina.

En el capítulo 4 expondremos el desarrollo de un algoritmo basado en redes neuronales recurrentes (*recurrent neural network*, RNN) que tendrá el mismo objetivo que el anterior, es decir, clasificar las fases de sueño en despierto y dormido. En este algoritmo aplicamos una red neuronal recurrente a la señal de frecuencia cardíaca (*heart rate*, HR) en conjunto con la señal de saturación periférica de oxígeno. para clasificar cada 30 segundos el estado del paciente. La arquitectura diseñada consiste en dos capas apiladas de unidades recurrentes con compuertas (*gated recurrent units*, GRU) junto con una capa *softmax* para efectuar la clasificación. En este capítulo se presenta además una versión modificada del algoritmo para obtener una clasificación que logre una discriminación completa en las fases del sueño, de acuerdo con la AASM.

Dado que las redes neuronales recurrentes presentan cierta dificultad para ser paralelizadas, siendo su entrenamiento muy costoso en términos de operaciones computacionales, en el capítulo 5 presentaremos el último algoritmo desarrollado con el mismo objetivo que los dos anteriores, pero significativamente más rápido de entrenar. Este algoritmo está basado en dos partes fundamentales: la primera es para obtener una representación del espacio de características en forma automática a partir de los datos, mientras que la segunda tiene por objetivo el modelado de la secuencia. La primera parte está basada en redes convolucionales temporales (*temporal convolutional networks*, TCN) y fue diseñada para obtener una representación de la señal de frecuencia cardíaca para ser luego utilizada como entrada de la segunda parte. Esta segunda parte está basada principalmente en un tipo de arquitectura que utiliza únicamente mecanismos de atención –denominada *transformer*– que fue desarrollada para aprender las reglas de transición entre las diferentes etapas de sueño.

En el capítulo 6 abordaremos unos experimentos preliminares para evaluar la variación en el desempeño al introducir el TTS estimado a partir de los algoritmos previos en un método desarrollado por otros investigadores que permiten obtener una estimación del AHI. En este capítulo, además, se planifican los principales trabajos que continuarán esta tesis y los principales desafíos a resolver.

Finalmente, en el capítulo 7 presentaremos las conclusiones a las que se arribó en la realización del trabajo, además de discutir posibles mejoras y trabajos futuros.

# Capítulo 2

## Marco Teórico

En este capítulo revisaremos los conceptos técnicos que consideramos necesarios para emprender la lectura de los próximos capítulos, en los que se presentarán los métodos propuestos. Comenzaremos por presentar una descripción de las patologías del sueño, en general, y del SAOS, en particular. Luego explicaremos las técnicas de PSG utilizadas para diagnosticar las patologías del sueño, así como también se describirán un conjunto de técnicas más simples para realizar un cribado de estas patologías. Se detallarán los procedimientos estándares para el análisis y etiquetado de los estudios, poniendo especial énfasis en la clasificación de las fases del sueño. Finalmente, se comentarán los antecedentes en el área en el que se insertarán los métodos propuestos en este trabajo, tanto para la clasificación de las fases del sueño sin la utilización de EEG como para el cribado del SAOS.

### 2.1. Síndrome de apnea obstructiva del sueño

En la sección 1.1 hemos realizado una breve descripción del problema con la intención de lograr transmitir la motivación de esta tesis doctoral. En esta sección, en cambio, el problema se abordará desde aspectos más técnicos. Inevitablemente se reiterarán algunos conceptos ya desarrollados con el objetivo de que el presente capítulo sea lo más autocontenido posible. Sin embargo, para no ir en desmedro de la legibilidad del documento, resta mencionar que estas reiteraciones se limitarán a lo indispensable.

Hemos mencionado en el capítulo anterior que una de las principales patologías del sueño es el SAOS. Como siempre sucede en los estudios epidemiológicos, existen discrepancias entre diferentes autores respecto a la prevalencia de esta patología en la población general. Una de las principales dificultades que presentan estos estudios epidemiológicos es la obtención de datos válidos a partir de muestras de una población, dado que para ello es necesario la realización de un estudio completo de PSG. Además, las diferencias en la definición de los umbrales para el IAH a partir del cual considerar la patología han contribuido a la variabilidad entre diferentes estudios. Uno de los estudios epidemiológicos con mayor impacto en la comunidad científica es el realizado por Young y cols. [40], que estima que existe un 9% de las mujeres y un 24% de los hombres adultos en edad laboral que poseen un IAH mayor o igual a 5. Además, este estudio ha estimado que un 2% de las mujeres y un 4% de los hombres superan los requisitos mínimos para ser diagnosticados con el SAOS (un IAH mayor o igual a 5 y somnolencia diurna) [40]. Otro estudio desarrollado por Durán y colaboradores ha calculado en una muestra de población de entre 30 y 70 años que un 26% de hombres y un 28% de mujeres poseen un IAH superior o igual a 5, mientras que este porcentaje baja a un 19% y un 15% cuando se considera como punto de corte un AHI de 10 [41]. Además, la prevalencia de esta patología tiende a incrementarse con la edad [42]. El factor de riesgo más importante para el SAOS –y para las patologías del sueño relacionadas

con la respiración en general— es la obesidad. Dado que existe un aumento continuo en la prevalencia de la obesidad, el número de pacientes diagnosticados con este síndrome se ha incrementado drásticamente en los últimos años [15].

El término «síndrome de apneas del sueño» fue introducido en el año 1976 por Guilleminault y colaboradores para definir a aquellos pacientes que presentaban apneas durante el sueño además de excesiva somnolencia diurna [43]. Desde entonces su estudio ha mantenido un creciente interés tanto en la clínica como en la academia.

En este trabajo presentaremos y utilizaremos la definición establecida en la tercera edición de la Clasificación Internacional de Patologías del Sueño (*International Classification of Sleep-Disorders, ICSD*), confeccionada por la AASM en colaboración con varias sociedades internacionales del sueño en el año 2014 [9, 44]. Existe un consenso general respecto a que este texto es la principal referencia para el diagnóstico de las patologías del sueño. Sin embargo, la mayoría de los documentos de distintas sociedades de medicina respiratoria presentan algunas diferencias y, muchas veces, sus propias variantes de esta definición, lo que dificulta la tarea de establecer una única definición aceptada globalmente.

La ICSD considera la apnea del sueño dentro de la categoría de patologías del sueño relacionadas con la respiración<sup>1</sup>. En este manuscrito se diferencian dos tipos diferentes de apnea: la «apnea obstructiva del sueño»<sup>2</sup> y la «apnea central del sueño».

Los principales criterios para el diagnóstico del SAOS se han mantenido invariables durante los últimos años [9]. Para ser considerado SAOS deben incluirse de los siguientes el criterio 1 o 2, además del criterio 3:

1. Conjunto de signos y síntomas como somnolencia diurna, fatiga, insomnio, ronquidos, problemas respiratorios nocturnos subjetivos o apnea observada.
2. Patologías asociadas, ya sean psiquiátricas o médicas, como hipertensión, enfermedades de las arterias coronarias, fibrilación auricular, falla cardíaca congestiva, diabetes, disfunción cognitiva o trastornos de humor.
3. Una frecuencia de 5 o más eventos respiratorios predominantemente obstructivos por hora de sueño durante la PSG como apneas obstructivas y mixtas, hipopneas, esfuerzos respiratorios relacionados con despertares, los cuales son definidos por el manual de la AASM y serán abordados más adelante.

En forma alternativa, la presencia de 15 o más eventos respiratorios obstructivos por hora de sueño es un criterio suficiente, incluso en ausencia de síntomas u otras condiciones asociadas [9].

En esta tercera edición se introduce un cambio respecto a ediciones anteriores que es relevante para la realización de este trabajo: el índice de eventos respiratorios puede ser obtenido ahora por fuera de laboratorios especializados del sueño. En este caso se continúan utilizando los mismos criterios de frecuencias, incluso cuando los autores conocen y aclaran que es común una subestimación al no realizar los estudios en un laboratorio de sueño dado que en lugar de utilizar el TTS se utiliza el TTR como denominador para el cálculo del índice. Vemos que las propuestas desarrolladas en este trabajo permitirán abordar este inconveniente planteado por uno de los organismos referentes en patologías del sueño a nivel mundial [9].

El síndrome de apnea central del sueño es menos frecuente que el obstructivo. A diferencia de este último, que se produce por un estrechamiento de las vías respiratorias durante el sueño, el cual impide el paso normal de aire a los pulmones a pesar del esfuerzo espiratorio

---

<sup>1</sup>El término original utilizado es *sleep-related breathing disorders*.

<sup>2</sup>En este documento, siempre hemos utilizado el término acompañado de la palabra síndrome para describir el conjunto de síntomas que se presentan con las apneas obstructivas, pero hacemos referencia a la misma patología.

realizado, en las apneas centrales el sistema nervioso interrumpe temporalmente el impulso de respirar durante el sueño [45]. Es decir, no existe un esfuerzo respiratorio. Por ello, para diferenciar entre ambos eventos es necesaria la utilización de sensores que permitan registrar la existencia o no del esfuerzo respiratorio. Asimismo, los tratamientos recomendados para las apneas centrales son muy diferentes a los tratamientos para apneas obstructivas, dado que unos atacan causas neurológicas, mientras los otros atacan un colapso faríngeo predominantemente físico [12, 45]. Dado que en este trabajo se plantean estudios de cribado simplificados, no será posible la discriminación entre ambos tipos de apnea. Sin embargo, los eventos de apnea obstructiva son muchísimo más comunes que los centrales.

En el criterio 2 para definir SAOS se mencionan algunas patologías asociadas. Merece enfatizarse la relación del SAOS con las enfermedades cardíacas. Si bien ambas patologías comparten muchos de los factores de riesgo como son la obesidad, una edad avanzada y género masculino, existe una asociación entre ambas enfermedades independientemente de los factores de riesgo en común [10]. Tal es así que ante la presencia de SAOS se incrementa en 2.38 veces la probabilidad de una falla cardíaca [13], en 4 veces la de fibrilación auricular [46] y entre 3 y 4 veces la de hipertensión arterial sistémica [47]. Además existen muchos estudios que la asocian con hipertensión sistémica y pulmonar, enfermedades en las arterias coronarias, diferentes tipos de arritmias y accidentes cerebrovasculares. Por último, también se encuentra asociado a secuelas neurocognitivas tales como vigilancia reducida, deterioro de la coordinación motora fina y depresión [48].

## 2.2. Métodos de diagnóstico

En esta sección abordaremos los métodos de diagnóstico existentes utilizados para evaluar las patologías del sueño. Diferentes sociedades de medicina del sueño han realizado diferentes agrupaciones de estos métodos, aunque todas poseen criterios similares. Todas ellas coinciden en establecer a la PSG como el procedimiento de diagnóstico estándar para la mayoría de las patologías del sueño –y para las patologías relacionadas con la respiración como el SAOS–. Luego, dadas las restricciones mencionadas en el capítulo anterior en lo que respecta a costos y disponibilidad de ese estudio, en términos generales todas las clasificaciones presentan variantes simplificadas de la PSG. En este trabajo seguiremos la presentación de los métodos de diagnóstico realizada por el «Consenso argentino de trastornos respiratorios vinculados al sueño» [47] y por las «Guías prácticas de diagnóstico y tratamiento del síndrome de apneas e hipopneas obstructivas del sueño: Actualización 2019» confeccionada por la Asociación Argentina de Medicina Respiratoria [49]. En estos documentos se dividen los métodos de diagnóstico en cuatro niveles, ordenados de mayor a menor complejidad.

**Nivel 1: Polisomnografía.** El estudio de PSG es el principal método de diagnóstico para las patologías del sueño. Consiste en un estudio nocturno que se realiza en un laboratorio del sueño, en una habitación tranquila y oscura en la que se registran en forma continua y simultánea múltiples variables fisiológicas en presencia de un técnico especializado [47]. Para caracterizar las diferentes fases del sueño se registran:

- derivaciones de EEG, las recomendadas son F4-M1, C4-M1 y O2-M1 cuyas posiciones de los electrodos están determinadas por el sistema internacional 10-20;
- señales de EOG en ambos ojos;
- señales de EMG obtenidas en la pera y en miembros inferiores.

Además, para monitorear las variables cardiorespiratorias es necesario registrar:

- flujo aéreo, en la mayoría de los laboratorios es registrado mediante un termistor ubicado en las fosas nasales y la boca, registrando cambios de temperatura entre el aire inspirado y expirado;
- movimientos de esfuerzo respiratorio detectado mediante transductores de movimiento-señal eléctrica en el tórax y el abdomen;
- saturación periférica de oxígeno registrada mediante un oxímetro de pulso;
- derivaciones de ECG, habitualmente dos derivaciones precordiales;
- ronquidos captados por un micrófono;
- posición corporal registrada manualmente o a través de un sensor.

Las señales presentadas registran en su totalidad los parámetros que deben ser reportados en forma obligatoria en una PSG según el «Manual para etiquetado del sueño y eventos asociados», presentado por la AASM [8].

Todos los eventos asociados al sueño, cuyas definiciones técnicas serán vistas un poco más adelante, deben luego ser identificados y etiquetados en forma manual por un médico especialista, en base a un conjunto de patrones y formas de onda características presentadas en el manual de la AASM [8]. Muchas veces esta tarea es realizada con la ayuda de un *software* para clasificación automática. Sin embargo, las comunidades médicas recomiendan fuertemente a los médicos verificar cada una de las etiquetas generadas en forma automática.

**Nivel 2: Polisomnografía no vigilada.** Es un estudio de PSG normal realizado en el domicilio, en hospitales o en laboratorios de sueño con la diferencia de que no es necesaria la presencia de un técnico para monitorear el registro durante la noche [49].

**Nivel 3: Poligrafía Respiratoria.** Son estudios que han surgido por las mismas razones que motivan esta tesis: el alto costo asociado a los estudio de PSG y su baja disponibilidad. Los equipos de este nivel registran una menor cantidad de señales que en la PSG [47]. En términos generales, podemos decir que la principal diferencia entre este nivel y el anterior es que no se registra el EEG. Por esta razón no se pueden estimar el TTS, el cual es luego aproximado mediante el TTR, conduciendo a un subdiagnóstico del SAOS.

Si bien en el consenso argentino seguido en esta clasificación no se utiliza el término, estos equipos suelen ser conocidos como poligrafías respiratorias<sup>3</sup>.

Estos estudios muchas veces se utilizan para agilizar el diagnóstico en situaciones en las que no hay acceso a PSG y su interpretación es realizada por especialistas y depende de la clínica del paciente considerado. De esta forma, ante un estudio normal en un paciente con alta sospecha de una patología del sueño es necesario completar el diagnóstico con una PSG. Si bien el empleo de poligrafías permite el ahorro de camas hospitalarias, existe un mayor consumo de insumos y el deterioro de los equipos es mayor. Adicionalmente, muchas veces se descartan los estudios por fallas en el registro, siendo necesario repetirlos. Por estos motivos, existen trabajos que sugieren que la PSG continúa siendo un estudio superior en términos costo-efectivos [50].

Abordando estas críticas, algunos grupos del sueño han desarrollado algoritmos para establecer qué estudios realizar a los pacientes según sus síntomas. En forma simplificada, diremos que estos estudios que poseen una menor exactitud son preferidos en casos donde existe una alta o una baja sospecha de presencia de una patología. Por el contrario, los casos más dudosos son estudiados directamente mediante PSG [51].

**Nivel 4: Oximetría de pulso.** En situaciones en las que no se disponga de estudios más complejos y ante la sospecha clínica de SAOS los registros de oximetría pueden ser útiles

<sup>3</sup>En el manual de la AASM existe un apartado con especificaciones técnicas para equipos de estas características, que son denominados *Home Sleep Apnea Testing*.

para diagnosticar y decidir tratamientos de esta patología [52]. Dado que este nivel es particularmente importante para nuestro trabajo, a continuación aprovecharemos a desarrollar brevemente los aspectos técnicos y las señales obtenidas mediante los oxímetros de pulso.

## 2.3. Señales de oximetría de pulso

Los oxímetros de pulso son dispositivos médicos cuyo principio de funcionamiento está basado en una técnica óptica ampliamente aceptada y difundida llamada fotoplethysmografía (*photoplethysmography*, *PPG*). La PPG consiste en una fuente y un detector de luz a partir de los cuales pueden estimarse varios parámetros fisiológicos. La señal de PPG es resultante de la interacción de la luz con los tejidos biológicos, es decir, del balance entre dispersión, absorción, reflexión, transmisión y fluorescencia de la señal. [53]. El uso más difundido de la PPG en los oxímetros de pulso es para estimar en forma indirecta la saturación de oxígeno en sangre arterial ( $\text{SaO}_2$ ) y HR.

La  $\text{SaO}_2$  es la fracción de oxihemoglobina relativa a la hemoglobina total. Los oxímetros de pulso permiten una medición no invasiva e indirecta de la  $\text{SaO}_2$ , comúnmente referida como saturación periférica de oxígeno en sangre ( $\text{SpO}_2$ ). Para ello, se utilizan dos fuentes de luz de diferente longitud de onda –roja e infrarroja– que presentan diferencias en su absorción en función del contenido de hemoglobina en sangre [54].

La estimación de la HR provista por los oxímetros de pulso es obtenida a partir de la componente pulsátil de la PPG, que es proporcional al cambio de volumen sanguíneo en el lecho microvascular. Este cambio de volumen se encuentra asociado al pulso periférico –sincronizado con el evento cardíaco– y es la característica más reconocible de la señal. Los algoritmos más comunes para la estimación de la HR consisten en filtros digitales y detectores de cruce por cero [53], aunque existen muchos trabajos con el objetivo de desarrollar algoritmos para reducir los artefactos de movimiento que afectan significativamente la calidad de la señal [55].

Adelantamos que en este trabajo la señal de HR tendrá preponderancia para determinar el TTS debido a la relación existente entre las fases del sueño y la dinámica de la HR, mientras que la señal de  $\text{SpO}_2$  tendrá una utilidad fundamental para la detección de los eventos de apnea e hipopnea –ya que éstos pueden ser detectados a partir de esta variable fisiológica, como se verá a continuación–. En esta tesis se abordará principalmente la aplicación relacionada con las fases del sueño. Respecto a la detección de eventos de apnea e hipopnea, se presentarán resultados preliminares, quedando los desarrollos fundamentales como una continuación mediante trabajos futuros.

## 2.4. Etapas de sueño según AASM

Una de las ya citadas principales desventajas de los métodos de cribado de SAOS más simples (nivel 3 y 4) es la imposibilidad de medir el TTS. Incluso, en los apartados del Manual de la AASM correspondientes al *home sleep apnea testing* se menciona que el TTS sólo puede ser obtenido a partir de las señales de EEG, EMG y EOG. En caso contrario, el tiempo considerado para calcular la frecuencia de eventos de apnea e hipopnea es el TTR o el tiempo de monitoreo –es el TTR descontando el tiempo en que el paciente estaba claramente despierto según registros actigráficos y descontando además los tiempos de desconexión del equipo en el que no se pudo obtener un registro confiable– [8]. En ambos casos la frecuencia de los eventos tenderá a ser subestimada.

Este inconveniente será uno de los problemas principales a abordar en este trabajo. Por ello, ahora brindaremos una breve descripción de la discriminación del sueño en diferentes

etapas, así también como su relación con la dinámica de la variabilidad de la frecuencia cardíaca. De esta forma obtendremos los conceptos fundamentales que luego nos permitirán comprender su clasificación automática mediante los algoritmos desarrollados y el posterior cálculo del TTS.

El sueño no es un estado uniforme en el que existe ausencia de vigilia, sino que tiene una estructura interna ordenada que es descripta en base a formas de onda características presentes principalmente en las señales de EEG, EMG y EOG. La técnica para registrar el sueño y la definición de las fases del sueño fueron estandarizados por primera vez en un manual compilado por Rechtschaffen y Kales en el año 1968 [16]. Estas primeras definiciones aún continúan formando parte esencial de la medicina del sueño. Sin embargo, en 2007 la AASM ha introducido algunas modificaciones a las etapas planteadas por R&K, principalmente con la intención de reducir las discrepancias entre profesionales en la práctica clínica. De esta manera, las fases de sueño consideradas y sus principales características según el estándar de la AASM son [8]:

- Fase W: representa el estado de vigilia, comprendido desde estados de total alerta hasta estados de somnolencia. Durante esta etapa se ve en la señal de EEG una predominancia de las ondas *alfa* –patrones de ondas sinusoidales de entre 8 y 13 Hz– y *beta* –mayores a 13 Hz–. Además, a partir del EOG pueden detectarse parpadeos y algunos movimientos oculares característicos de la vigilia.
- Fase N1: representa un estado de adormecimiento. Las ondas *alfa* decrecen en magnitud, prevaleciendo una actividad en la señal de EEG de ondas *theta* –frecuencia entre 4 y 7.99 Hz– y movimientos oculares lentos. Esta etapa ocupa aproximadamente el 10 % del sueño.
- Fase N2: representa un estado de sueño ligero. Durante esta etapa comienzan a aparecer ondas *theta* y *delta* –frecuencia entre 0 y 3.99 Hz–. Además son característicos los complejos K –ondas agudas que aparecen de forma abrupta y actúan como mecanismo de inhibición para que el sujeto no se despierte– y husos del sueño –conjuntos de ondas de entre 12 y 14 Hz que suceden varias veces por minuto–. Esta etapa ocupa la mayor parte del sueño, llegando a alcanzar hasta un 50 % del TTS.
- Fase N3: representa los estados de sueño profundo, que en R&K comprendían dos etapas diferentes. Durante esta etapa domina la actividad de onda lenta en la señal de EEG –frecuencia comprendida entre 0.5 y 2 Hz con una amplitud mayor a  $75 \mu\text{V}$  medido en la región frontal–, mientras que la señal de EMG presenta una baja amplitud. No suelen observarse movimientos oculares. Ocupa aproximadamente el 20 % del TTS. La razón entre el tiempo en esta etapa y el TTS es un indicador de la calidad del sueño.
- Fase REM: esta etapa está caracterizada por movimientos oculares rápidos. La señal de EEG es muy similar a la que se presenta durante la vigilia. El cerebro presenta una alta actividad, sin embargo el tronco encefálico bloquea las neuronas motoras por lo que no es posible moverse. El tiempo de sueño REM suele ser aproximadamente el 25 % del TTS.

Estas fases del sueño son evaluadas etiquetando las señales cada 30 segundos. Cada segmento de esta duración es etiquetado con una fase de sueño correspondiente. En caso de presentarse una mezcla de patrones característicos de diferentes etapas, el especialista deberá asignar la etiqueta correspondiente a la clase mayoritaria [8].

La representación gráfica de la sucesión de fases de sueño a lo largo del estudio se denomina **hipnograma**. En casos de sueño normal, las fases del sueño siguen una secuencia ordenada iniciando con W, luego N1 y N2, seguido de N3 para finalizar con sueño REM. Esta

secuencia se denomina ciclo del sueño, cuya duración típica es de entre 90 y 110 minutos. Una noche normal consiste en aproximadamente 6 ciclos de sueño donde la proporción de N3 decrece con el paso de los ciclos, incrementándose gradualmente la proporción de sueño REM [32].

### 2.4.1. Automatización en la clasificación de las fases del sueño

Todos estos manuales que han establecido reglas, definiciones y especificaciones técnicas con el objetivo de sistematizar el procedimiento de etiquetado del sueño y sus eventos asociados estipulan realizar la clasificación en forma manual [8]. Aún más, sociedades como la Asociación Argentina de Medicina Respiratoria desaconsejan utilizar modelos de predicción clínica como método de diagnóstico y adoptar medidas terapéuticas en función de éstos [49]. A pesar de esto, la comunidad científica viene realizando enormes esfuerzos desde hace años para automatizar la clasificación de las fases del sueño y, de esta manera, superar las desventajas del etiquetado manual. Desventajas que no sólo contemplan como factor lo tedioso y costoso –en términos de tiempo y gastos económicos– que es realizar esta actividad en forma manual, sino también los resultados de varios estudios que muestran una significativa variabilidad inter-profesional y, peor aún, intraprofesional [27]. Es decir, el etiquetado manual es una actividad muy susceptible a errores humanos. Además, las definiciones de las fases del sueño no son del todo claras y diferentes «escuelas» incorporan diferentes criterios adicionales, contribuyendo a incrementar las discrepancias.

Para medir el grado de concordancia entre dos clasificadores diferentes es habitual utilizar el índice *kappa* de Cohen  $\kappa$ . Esta medida es más robusta y conservadora que el simple porcentaje de acuerdos entre clasificadores, ya que tiene en cuenta la probabilidad de acuerdo por azar entre los clasificadores. Landis y Koch propusieron en 1977 una escala de valoración que continúa siendo altamente utilizada para caracterizar el  $\kappa$ , considerando que valores menores o iguales a 0 indican que no existe acuerdo, valores en el intervalo  $(0, 0.2]$  se corresponden a un acuerdo insignificante, valores en el intervalo  $(0.2, 0.4]$  se corresponden a un acuerdo discreto, valores en el intervalo  $(0.4, 0.6]$  corresponden a un acuerdo moderado, valores en el intervalo  $(0.6, 0.8]$  se corresponden a un acuerdo sustancial y valores en el intervalo  $(0.8, 1]$  se corresponden a un acuerdo casi perfecto [56]. Esta escala de valoración se resume en la tabla 2.1. Un estudio realizado por Danker-Hopfe y cols. en el 2009 calculó los índices  $\kappa$  para diferentes observadores con el objetivo de evaluar si existía una mejora en el grado de acuerdo por los cambios introducidos por la AASM respecto al estándar anterior, R&K. Para el estándar de la AASM este estudio encontró que  $\kappa = 0.76$ , mientras que para el R&K el valor obtenido fue  $\kappa = 0.68$  –ambos corresponden a un acuerdo moderado– [57]. Por otro lado, un estudio realizado por Whitney y cols. [58] evaluó la variabilidad intraprofesional, analizando el grado de concordancia entre clasificaciones de registros realizadas por el mismo profesional con una diferencia entre sí de aproximadamente 6 meses. En este caso el valor obtenido para el índice  $\kappa$  fue entre 0.79 y 0.87 [58]. Estas estimaciones permiten tener una referencia para luego valorar la factibilidad de utilizar algoritmos automáticos, que en muchos casos presentan índices de concordancia mucho más altos respecto a una referencia bien clasificada.

Con esto en consideración vemos que la clasificación automática de las fases del sueño aparece como una aproximación eficiente para superar las desventajas del etiquetado visual en los estudios de sueño. La mayoría de los algoritmos para clasificación automática consisten en la aplicación de técnicas de procesamiento de señales en conjunto con algoritmos de aprendizaje maquina que permiten extraer información útil de las señales biológicas. Los métodos para clasificación automática de las fases del sueño pueden ser agrupados en dos categorías diferentes: los métodos multicanal y los métodos de un único canal [31].



Índice $\kappa$ de Cohen $\kappa$	Nivel de acuerdo
(0.0, 0.2]	Insignificante
(0.2, 0.4]	Discreto
(0.4, 0.6]	Moderado
(0.6, 0.8]	Sustancial
(0.8, 1.0]	Casi perfecto

Tabla 2.1: Índice Kappa: magnitud y significado.

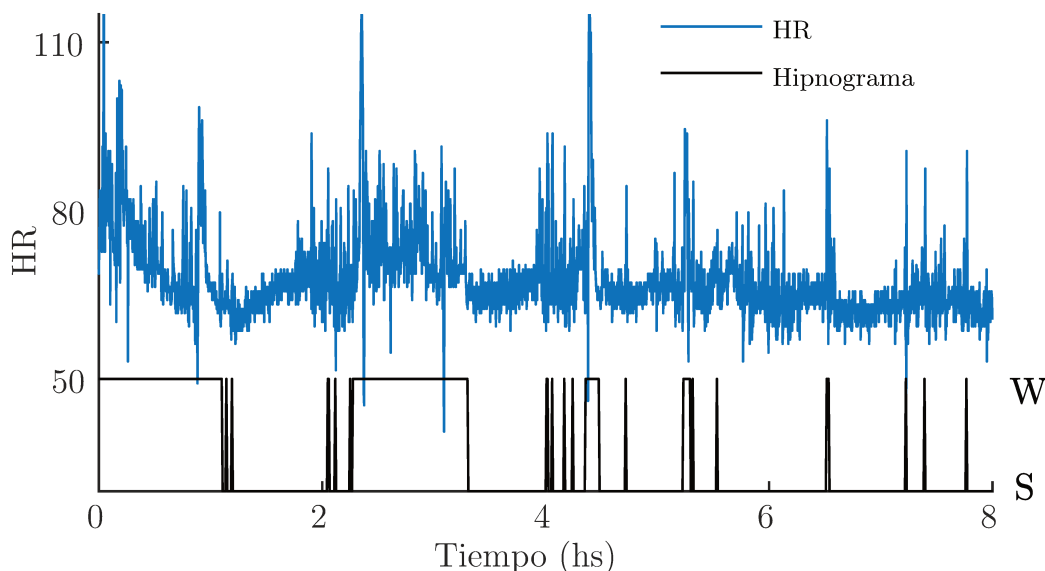


Figura 2.1: Hipnograma y HR. Pueden notarse los cambios en la dinámica de la señal de HR durante las fases correspondientes a sueño y a despierto.

Los métodos multicanal utilizan como entrada una combinación de señales biológicas que consiste en varios canales de EEG, a los que muchas veces también se les añade información extraída de otras señales como EMG y/o EOG. Si bien estos métodos en general obtienen un mejor desempeño, poseen algunas restricciones para el diagnóstico domiciliario por ser más costosos y complejos. Además, como ya describimos en los métodos de diagnóstico, el número excesivo de conexiones puede dificultar la conciliación normal del sueño. Algunos ejemplos de trabajos que pueden ser agrupados en los métodos multicanal son los realizados por Phan y cols. [29], por Özşen y cols. [59] y Shi y cols. [60].

Una forma de abordar estos inconvenientes ha sido utilizar métodos menos costosos y más simples basados en un único canal de EEG. En este sentido, algunos estudios han demostrado que un único canal de EEG es suficiente para obtener una clasificación confiable [61]. Los trabajos realizados por Supratak y cols. [28], Fraiwan y cols. [30], Phan y cols. [62] y Hsu y cols. [63], entre muchos otros, son algunos ejemplos de algoritmos basados en un único canal de EEG.

**HR y fases del sueño.** A pesar de que los métodos de un único canal son más simples, la señal de EEG tiene una instrumentación bastante más compleja en comparación con otras señales. Por este motivo, acompañando el incremento en la demanda de métodos de diagnóstico domiciliarios han comenzado a surgir varios estudios que clasifican las fases del sueño sin utilizar EEG. Esto puede ser muy útil para cribados masivos que permitan evaluar la prevalencia de patologías del sueño en grandes poblaciones, además que permitirán evitar la saturación de los sistemas de salud y, consecuentemente, las largas listas de espera y el subdiagnóstico tan común en la medicina del sueño [4].

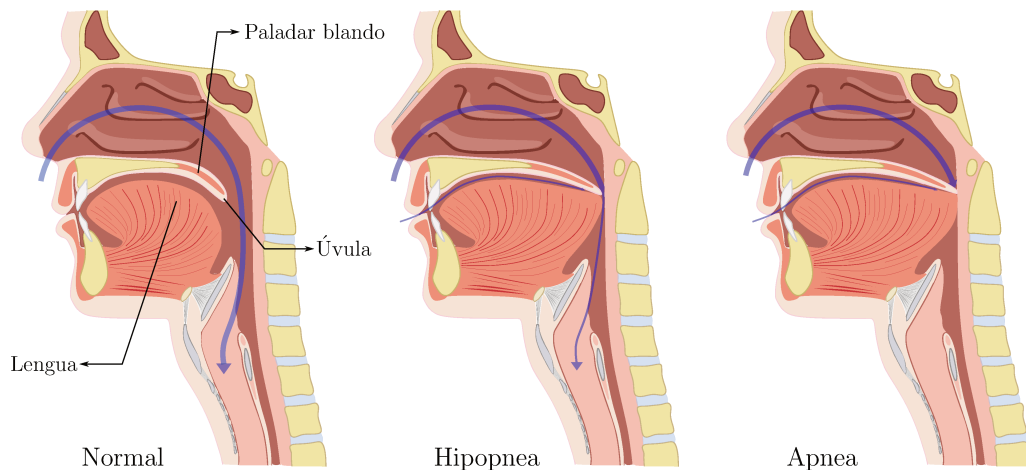


Figura 2.2: Ilustración de los eventos de apnea e hipopnea.

En este sentido, las propuestas más extendidas –que no están basadas en EEG– para clasificar las fases del sueño utilizan señales cardíacas. Estos trabajos se basan en explotar la relación que existe entre las fases del sueño y la regulación de la HR. La regulación del sistema nervioso autónomo cambia con las etapas del sueño y, consecuentemente, se producen cambios apreciables en la HR. Existen estudios que han encontrado evidencia de que la HR promedio es mayor durante la vigilia, decreciendo en forma continua desde el sueño liviano al sueño profundo, para luego incrementarse nuevamente durante el sueño REM [32, 33]. Sumado a esto, la variabilidad de la HR es mayor durante las fases W y REM que durante N1, N2 y N3 [32]. Esta situación se ilustra en la figura 2.1.

Estos estudios han establecido las bases para el desarrollo de múltiples algoritmos basados en señales cardíacas, principalmente ECG [34–36]. Sin embargo, artículos muy recientes han sido publicados utilizando señales de PPG obtenida mediante oxímetros de pulso, siempre con el mismo argumento de ser una señal más portable, simple y cuya tecnología tiene un costo menor que el ECG [37, 38, 64]. En estos trabajos, las fases del sueño son usualmente agrupadas con el objetivo de obtener un resultado primario que permita estimar rápidamente el estado del paciente, para luego ser transferidos a un estudio completo en caso de ser necesario. Es habitual clasificar las fases del sueño en W, REM y no-REM –agrupando N1, N2 y N3– o directamente en W y dormido –al que denominaremos S y es obtenido agrupando REM y no-REM–.

## 2.5. Eventos respiratorios según AASM

Ya hemos mencionado en el capítulo anterior que las obstrucciones faríngeas características del SAOS pueden ser parciales o totales, diferenciándose ambos eventos como hipopneas y apneas, respectivamente. Estos casos se ilustran en la figura 2.2. En este apartado se presentarán los criterios estándares para clasificar los diferentes eventos respiratorios.

Según la AASM, un evento respiratorio es considerado una **apnea** si cumple con los siguientes dos criterios [8]:

- Existe una caída en la magnitud de alguna de las señales relacionadas con la respiración mayor o igual al 90 % respecto al registro basal.
- La duración de la caída mayor o igual al 90 % se mantiene por un tiempo mayor o igual a 10 segundos.

La señal recomendada para identificar los eventos de apnea durante un estudio diagnóstico es aquella obtenida mediante un sensor térmico de flujo de aire oronasal (termistor oronasal). En caso de que este sensor no se encuentre funcionando o su registro no sea confiable, se pueden utilizar como alternativas un transductor de presión nasal, suma de pletismografía de inductancia respiratoria o flujo de pletismografía de inductancia respiratoria[8].

Además, los eventos de apnea pueden clasificarse como **obstructivos** –asociados a un esfuerzo inspiratorio continuo o incrementado durante el período de ausencia de flujo de aire–, **centrales** –ausencia del esfuerzo inspiratorio– o **mixtos** –ausencia de esfuerzo inspiratorio durante la primera porción del evento, seguido luego de una restauración del esfuerzo en la segunda porción– [8].

Asimismo, un evento es considerado una **hipopnea** si cumple con los siguientes tres criterios:

- Existe una caída en la magnitud de alguna de las señales relacionadas con la respiración mayor o igual al 30 % respecto al registro basal.
- La duración de la caída mayor o igual al 30 % se mantiene por un tiempo mayor o igual a 10 segundos.
- Existe una desaturación de oxígeno en sangre mayor o igual al 3 % respecto al registro basal o el evento está asociado con un despertar.

En el caso de las hipopneas, la señal recomendada para su medición es la presión nasal y los sensores alternativos considerados en caso de falla son el termistor oronasal, la suma de pletismografía de inductancia respiratoria y el flujo de pletismografía respiratoria [8].

Las hipopneas serán **obstructivas** si durante el evento se producen ronquidos, se observa un aplanamiento de la presión nasal o está asociado a una paradoja toracoabdominal<sup>4</sup>. Por otro lado, la hipopnea será **central** si no satisface ninguno de los criterios previos [8].

## 2.6. Antecedentes

En esta sección se presentarán los principales antecedentes para la clasificación de fases del sueño utilizando señales cardíacas, así como también los antecedentes para el cribado del SAOS utilizando las señales de saturación periférica de oxígeno en sangre. Para el caso de las fases del sueño no se presentarán únicamente los trabajos que usen oximetría de pulso, ya que existen muy pocos algoritmos. Por esto, incorporamos también algunos trabajos que utilizan la señal de ECG.

### 2.6.1. Clasificación automática de fases del sueño

Dado el incentivo para incrementar los estudios simplificados para patologías del sueño, se hizo evidente la necesidad de algoritmos que permitan obtener medidas equivalentes a las obtenidas con PSG a partir de los nuevos métodos de diagnóstico. Ya hemos mencionado que los desarrollos propuestos en esta tesis estarán basados en los métodos de diagnóstico Nivel 4 que utilizan oximetría de pulso [47]. El oxímetro de pulso es una elección ideal para el cribado de patologías del sueño debido a su bajo costo, accesibilidad y la simpleza de su instrumentación [17].

Existe una cantidad enorme de algoritmos basados en EEG que clasifican en forma automática las fases del sueño. Dado que en este trabajo utilizaremos señales obtenidas mediante

<sup>4</sup>Hace referencia a una desincronización entre los movimientos respiratorios toraco-abdominales. En una respiración normal el diafragma se mueve hacia abajo durante la inspiración y hacia arriba durante la espiración. En la paradoja toraco-abdominal estos movimientos se invierten.

oximetría de pulso como entradas, sólo se presentarán algunos de ellos con el objetivo de tenerlos como una referencia ideal de desempeño. Luego de esto, se presentarán en forma más exhaustiva los algoritmos basados en señales cardíacas.

Fraiwan y cols. [30] desarrollaron un método para clasificar las etapas de sueño de acuerdo a los estándares establecidos por la AASM. Para ello utilizaron características extraídas en forma manual de un único canal de EEG basadas en distribuciones tiempo-frecuencia. Estas características fueron clasificadas mediante el algoritmo de bosques aleatorios (*random forest, RF*) [30]. Este método fue el que obtuvo el mejor desempeño en un artículo en el que se revisaron en forma exhaustiva métodos para clasificación automática de fases del sueño, el cual fue publicado en el año 2017 por Boostani y cols. [31].

Apenas cuatro meses más tarde Supratak y cols. [28] publicaron un artículo en el que describían un método basado en redes neuronales profundas en el que superaron el desempeño obtenido por Fraiwan. Tal y como viene sucediendo en la gran mayoría de las aplicaciones, métodos específicos diseñados para aplicaciones particulares (que contaban con años de estudios de la temática y cuyo diseño era abordado partiendo de una comprensión profunda de los fenómenos subyacentes involucrados) están siendo superados en su desempeño por algoritmos basados en arquitecturas bastante genéricas de redes neuronales. De esta forma, el método desarrollado por Supratak y cols. utilizó una arquitectura basada en redes neuronales convolucionales (*convolutional neural network, CNN*) –para aprender características en forma automática a partir de los datos de EEG– y RNN –para aprender las reglas de transición entre las fases de sueño.

Durante el 2019, Mousavi y cols. [65] presentaron un método basado en un canal único de EEG para obtener una clasificación de las fases del sueño según la AASM. Este método es similar al presentado por Supratak, por lo que su arquitectura contiene CNN para la extracción de características en forma automática a partir de los datos de EEG que luego son procesadas por RNN. La diferencia se encuentra en que el posterior procesamiento de la secuencia se realiza con una estructura *encoder-decoder*, ambos constituidos por las RNN. Entre el *encoder* y el *decoder* se utilizó una red basada en mecanismos de atención. De esta forma, el *encoder* codifica las características que conforman la secuencia de entrada, mientras que el *decoder* genera las clasificaciones a partir de dicha codificación. Generalmente, el último elemento que sale del *encoder* es el que tiene más influencia sobre el *decoder*. Por este motivo, los autores añadieron un mecanismo de atención que permite evaluar las partes de la secuencia de entrada que tendrían mayor influencia en la secuencia de salida. Hasta donde conocemos, Mousavi y cols. logran con este método el mejor desempeño en lo que respecta a algoritmos para clasificación de fases del sueño que utilizan un único canal de EEG [65].

Es vasta la cantidad de trabajos que resuelven esta aplicación mediante EEG y no será abordada aquí con exhaustividad. Para quienes estén interesados, la ya citada revisión de Boostani es un buen comienzo para tener una idea general de la problemática y el estado del arte utilizando EEG [31]. Antes de pasar a los algoritmos basados en frecuencia cardíaca, citaremos un último trabajo –esta vez correspondiente a las aproximaciones multicanal– realizado por Phan y cols. [29]. En ese trabajo utilizan señales de EEG, EOG y EMG, y las clasifican utilizando una combinación de CNN, RNN y redes de atención. Con este algoritmo obtienen un desempeño ligeramente superior al obtenido por los algoritmos de un único canal.

El surgimiento de los métodos de diagnóstico que no utilizan EEG ha sido un estímulo para el desarrollo de algoritmos para clasificación de fases del sueño simplificados a partir de otras señales. Estudios como el de Penzel y cols. [32] sobre la regulación de la HR por el sistema nervioso autónomo y las modificaciones que producen en las fases del sueño han permitido estos desarrollos. Por ejemplo, Adnane y cols. [34] desarrollaron un método clásico de aprendizaje maquina que utiliza ECG para clasificar las fases de sueño en despierto y dormido. El método está basado en extracción de características, que luego son clasificadas

por máquinas de soporte vectorial (*support vector machine*, SVM) que además seleccionan un subconjunto de características óptimas mediante eliminación recursiva. Este trabajo fue diseñado utilizando un esquema de sujeto específico, esto significa que datos del mismo sujeto fueron utilizados para entrenar el clasificador, y la parte restante para evaluarlo. En general el desempeño disminuye al utilizar diferentes sujetos para entrenar y testear los algoritmos. En una aplicación real, es necesario realizar varios estudios sobre el mismo sujeto para poder «personalizar» el algoritmo a una persona en particular. Además, los estudios que se utilizarán para entrenamiento deben ser etiquetados por un profesional, para lo cual muchas veces necesita realizarse una PSG que sirva de referencia. Esto los hace menos factible de ser aplicados en estudios de cribado, aunque puede ser una buena opción para casos de seguimiento de un paciente crónico. Por otro lado, también utilizando únicamente señales de ECG, Malik y cols. [36] desarrollaron una CNN para clasificar el sueño en despierto y dormido. A partir del ECG se extrajo la HR instantánea, que luego fue utilizada como entrada de la red, cuyo objetivo era aprender características de forma automática. Además, el desempeño de la red fue evaluado usando PPG para calcular la HR. Otra de las fortalezas de este trabajo es que los autores han entrenado y testeado la red en diferentes bases de datos, evaluando el desempeño al utilizar diferentes dispositivos –ECG o PPG– y diferentes bases de datos. Los desempeños que obtuvieron fueron similares en todos los casos [36].

La oximetría de pulso es un dispositivo más simple que el ECG a partir del cual también puede extraerse información relacionada con el funcionamiento cardíaco. Uçar y cols. [37] desarrollaron un método clásico que utiliza PPG, variabilidad de HR y una combinación de ambas entradas para clasificar las fases del sueño en despierto y dormido utilizando dos clasificadores diferentes: SVM y *k-nearest neighbour* [37]. Continuando con el mismo orden en el que presentamos los algoritmos de ECG, mencionaremos ahora el trabajo de Malik y cols. [36] que aplicó su método basado en CNN y desarrollado para ECG en señales de PPG, que ya fue discutido en el párrafo anterior [36].

Los trabajos listados hasta ahora utilizan únicamente una señal cardíaca, ECG o PPG. Ahora presentaremos algunos trabajos que añaden a la señal cardíaca alguna otra señal extra para incrementar la información respecto a la fase del sueño. Beattie y cols. utilizaron señales de PPG –completa, no sólo la HR extraída de ella– y de acelerómetro para clasificar las fases del sueño en dormido, sueño liviano, sueño profundo y REM. La señal de PPG permite añadir información adicional que estaría relacionada de forma indirecta con la actividad muscular. Como ya hemos mencionado, la actividad muscular, medida por el EMG, era una de las señales con mayores variaciones ante las diferentes fases del sueño [38]. El agregado de otros sensores hacen que el dispositivo incremente su complejidad. Sin embargo, en el caso de los acelerómetros, el dispositivo sigue siendo útil para confeccionar sistemas portables de diagnóstico y en tecnología «vestible»<sup>5</sup>. Aunque este trabajo utiliza un enfoque clásico de aprendizaje maquina, utilizando extracción de características, el desempeño obtenido es notable.

Finalmente, Fonseca y cols. [66] utilizaron señales de ECG y de acelerómetro para clasificar las fases del sueño de acuerdo con las reglas establecidas por la AASM. Estos autores utilizaron características extraídas en forma manual, las cuales fueron combinadas con arquitecturas basadas en RNN, con las que obtuvieron resultados significativos. Además, en ese trabajo se entrenó y testeó la red desarrollada en diferentes bases de datos [66].

La comparación entre algoritmos tan disímiles es un proceso bastante complicado, ya que fueron diseñados y testeados utilizando diferentes bases de datos y diferentes tipos de señales. No existen trabajos que sean directamente comparables, con alguna excepción de diferentes algoritmos desarrollados por el mismo grupo. A pesar de esto, en este trabajo se

---

<sup>5</sup>El término utilizado en inglés para este tipo de tecnología es *wearable*. Otras traducciones al castellano son «tecnología vestible» y «tecnología corporal».

expondrá el desempeño obtenido y las características principales de los algoritmos utilizados que sirvan como una referencia de desempeño en esta aplicación.

En la tabla 2.2 se presenta un resumen de los resultados obtenidos para los trabajos que fueron presentados. A medida que se describan nuestras propuestas, incluiremos en esta tabla nuestros resultados.

Método	Señales	C	N	Acc	Se	Sp	PPV	NPV	$\kappa$
CWT feat.+RF [30]	EEG	5	16	93.4	95.9	89.6	93.3	93.5	0.86
CNN + RNN [28]	EEG	5	20	94.3	96.8	83.4	86.0	96.2	0.81
CNN + RNN [65]	EEG	5	61	95.7	96.9	90.6	87.8	97.7	0.86
Feat. + SVM + RFE [34]	ECG	2	18	80.0	84.5	69.1	87.0	64.5	0.52
CNN [36]	ECG (CGMH-val)	2	27	83.1	89.4	52.4	90.1	50.5	0.41
CNN [36]	ECG (DREAMS)	2	20	81.4	87.1	53.1	90.2	45.2	0.38
CNN [36]	ECG (UCDSADB)	2	25	73.7	81.9	43.4	84.3	39.2	0.24
Feat. + SVM [37]	HRV	2	10	73.1	74.0	72.0	–	–	0.46
Feat. + SVM [37]	PPG	2	10	76.8	76.0	77.0	–	–	0.53
Feat. + kNN [37]	PPG + HRV	2	10	79.4	77.0	81.0	–	–	0.59
CNN [36]	PPG (CGMH-val)	2	27	84.2	90.9	51.5	90.1	53.6	0.43
Feat. + RNN [66]	ECG + acc	5	778	90.1	94.0	72.9	–	75.1	0.65
Feat. + LDA [38]	PPG + acc	4	60	90.6	94.6	69.3	94.3	70.5	0.64

Tabla 2.2: Desempeños y aspectos destacados obtenidos en trabajos similares. C representa la cantidad de clases en las que se realiza la clasificación, 2 corresponde a despierto y dormido, 4 a despierto, sueño liviano, sueño profundo y REM y 5 a la clasificación según AASM. N es la cantidad de sujetos en la base de datos. En el trabajo de Malik y cols. [36] se presentan los resultados al evaluar un algoritmo entrenado utilizando un subconjunto de la base de datos CGMH que fue testado en diferentes bases de datos.

### 2.6.2. Cribado de apnea del sueño

Es muy común que durante los eventos de apnea obstructiva se produzca una desaturación de oxígeno en sangre, que es registrada indirectamente en la señal de  $SpO_2$ . Durante estos eventos, las desaturaciones tienen un patrón característico similar a una onda diente de sierra, es decir que existe una caída suave y pronunciada durante el evento de apnea a la que le sucede un incremento rápido en la  $SpO_2$  cuando el sujeto se despierta. Esto se ilustra en la figura 2.3. Esta forma de onda no es tan clara durante los eventos de hipopnea, pudiendo incluso estar ausente. También es común no observar el patrón de diente de sierra en las apneas centrales [67].

A partir de las desaturaciones identificadas en la señal de  $SpO_2$  puede construirse un índice de desaturaciones de oxígeno (*oxygen desaturation index*, ODI), que representa la frecuencia en la que la saturación de oxígeno cae debajo de cierto nivel. Los niveles más utilizados para contar las desaturaciones son el 2%, el 3% y el 4%, construyendo el ODI2, ODI3 y ODI4, respectivamente [68]. Como ya dijimos que no todo evento de apnea o hipopnea produce una desaturación, nos podemos imaginar que el AHI y el ODI no son directamente intercambiables. La mayoría de los estudios de cribado que usan oximetría de pulso sólo pueden estimar el ODI. La pregunta que surge es si una vez estimado el ODI será posible inferir el AHI. No hay una respuesta determinística en este sentido, ya que no existe una relación directa entre ambos índices. Sin embargo, algunos trabajos han estimado el comportamiento de la relación entre ambos índices en función del índice de masa corporal. Las relaciones ODI/AHI –considerando los tres ODIs posible– se incrementan a medida que

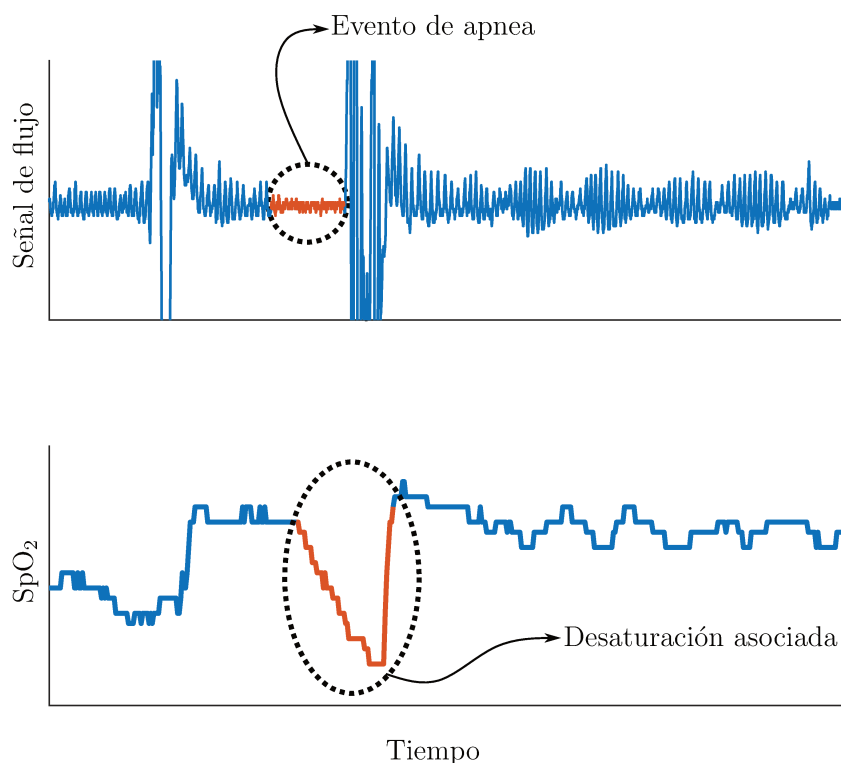


Figura 2.3: Señal de flujo aéreo y señal de  $SpO_2$  en un paciente que sufre un evento de apnea obstructiva. Se señala en rojo el evento de apnea obstructiva, durante el cual puede verse una importante disminución del flujo aéreo. En la señal de  $SpO_2$ , en cambio, se observa una desaturación con forma de diente de sierra. Puede observarse, además, que entre las manifestaciones que se presentan en ambas señales debido al evento existe un retardo de tiempo.

aumenta la masa corporal. Por ejemplo, para un índice de masa corporal de  $30 \text{ kg/m}^2$ , el  $ODI4/AHI \approx 0.3$ , el  $ODI3/AHI \approx 0.6$  y el  $ODI2/AHI \approx 1.2$ . Considerando esto, algunos métodos que estimen el ODI podrán clasificar la severidad del SAOS de acuerdo a los criterios utilizados para el AHI –aunque posiblemente introduciendo un error extra debido a que la relación no es unívoca–.

Las reducciones de las desaturaciones son medidas a partir de una línea de base o estado basal que corresponde a una oxigenación normal. Lamentablemente no existe un consenso unificado sobre como estimar esta línea de base [8]. Existen métodos que calculan la línea de base como el valor medio de  $SpO_2$  durante todo el estudio. Otros métodos consideran únicamente los primeros 3 minutos del estudio [69]. Por otro lado, dado que en algunos casos el nivel de  $SpO_2$  cae a un valor estable por debajo de la línea de base durante el sueño, muchas veces es más robusto calcular la línea de base de forma dinámica. Estas características las iremos informando a medida que describamos los antecedentes en el área.

Uno de los primeros trabajos de los que tenemos conocimiento es el realizado por Chiner y cols. en el año 1999 [70]. En ese trabajo se consideró como desaturación toda caída por debajo del 4% en la señal de  $SpO_2$  respecto a una línea de base definida por el valor medio del minuto previo al evento. Luego se establecieron tres umbrales diferentes para clasificar a los pacientes según el valor del ODI:  $ODI \geq 5$ ,  $ODI \geq 10$  y  $ODI \geq 15$ . En otro trabajo realizado por Vázquez y cols. [22] la línea de base fue definida utilizando el quintil superior de los valores de la  $SpO_2$  durante los 5 minutos anteriores que preceden al evento. De esta forma, este método no tiene en cuenta las desaturaciones en la señal para calcular la línea de base.

En el trabajo de Xie y cols. [71] se utiliza un método similar, pero en este caso se considera el 20 % superior de los valores de la  $\text{SpO}_2$  durante el último minuto. Las principales fallas de los algoritmos que realizan este tipo de detección están relacionados con la estimación incorrecta de la línea de base.

Con esta característica en consideración, Schlotthauer y cols. [19] propusieron un método en el que no sea necesario el establecimiento de un umbral a partir de una línea de base estimada. Para ello, aplicaron la técnica de descomposición empírica en modos (*empirical mode decomposition*, EMD) para procesar la señal de  $\text{SpO}_2$ . Luego, los autores generan un conjunto de señales auxiliares mediante la suma de modos consecutivos sobre la que detectan extremos. A partir de los extremos calculan la diferencia en amplitud y en tiempo entre un máximo local y el siguiente mínimo local, y si esta diferencia es superior a dos umbrales –un umbral de tiempo y otro de amplitud–, el evento es considerado una desaturación. Finalmente, los autores calculan el ODI como la relación entre el número de eventos detectados y la duración total de la señal y a partir de este valor se calcula la severidad de la patología del paciente.

Otra serie de trabajos que es importante resaltar son los desarrollados por Rolón y cols. [20], que también encuentran su origen en la motivación de superar las desventajas de los métodos que estiman una línea de base. En estos trabajos se realiza un análisis basado en diccionarios, generando representaciones ralas de las señales de  $\text{SpO}_2$  que luego son clasificadas para identificar los eventos de apnea/hipopnea. El trabajo de Rolón y cols. presentado en 2016 introduce dos métodos para cuantificar y seleccionar los átomos que permiten una mayor discriminación de los eventos de apnea/hipopnea. A partir de estos átomos luego se identifican los eventos de apnea/hipopnea a partir de un clasificador basado en MLP, para finalmente estimar el IAH [20]. En 2020 estos autores presentaron un nuevo método para cuantificar el grado de discriminabilidad de los átomos [72], mejorando los resultados obtenidos anteriormente. Además, modificaron los métodos del artículo anterior para que en lugar de realizar una clasificación binaria –como lo hacía la propuesta original– se realice una clasificación multi-clases, diferenciando los eventos de apnea y los de hipopnea. Estos eventos evidencian una morfología diferente en la señal de  $\text{SpO}_2$ , por lo que una clasificación multi-clase genera un incremento en el desempeño de los clasificadores.

En la tabla 2.3 se presentan los desempeños de los trabajos que fueron presentados, que luego serán comparados con los desarrollos de esta tesis.

Método	N	AUC	Acc	Se	Sp
DAS-KSVD [72]	954	0.957	87.6	88.3	87.9
MDCS-MC [72]	954	0.942	86.1	86.6	86.4
MDAS-MC [72]	954	0.913	82.3	82.8	82.6
MDCS-BC [20]	954	0.937	85.7	85.9	85.8
MDAS-BC [20]	954	0.891	81.0	83.1	82.1
Schlotthauer y cols. [19]	996	0.922	84.1	85.9	85.0
Vázquez y cols. [22]	315	0.909	80.8	87.5	84.2
Chiner y cols. [70]	216	0.795	76.2	78.1	77.2

Tabla 2.3: Desempeños y aspectos destacados obtenidos en trabajos similares para clasificación de eventos de apnea/hipopnea. N es la cantidad de sujetos en la base de datos.

## 2.7. Base de datos

En esta tesis se hará uso de la base de datos *Sleep Heart Health Study* (SHHS) para desarrollar, evaluar y validar los métodos propuestos [73, 74], ya que contiene registros obte-



	SHHS 1 (min, max)
Sujetos totales	5080
Edad	$63.1 \pm 11.2$ (39.0, 90.0)
Porcentaje de mujeres	52.3 %
Escala de somnolencia de Epworth	$7.8 \pm 4.4$ (0.0, 24.0)
Índice de despertares(/hr)	$19.2 \pm 10.7$ (0.0, 110.4)
Índice de apnea/hipopnea (/hr)	$9.6 \pm 12.7$ (0.0, 115.8)
Tiempo total de sueño (min)	$587.7 \pm 107.6$ (35.0, 858.0)
Tiempo dormido / tiempo total	0.742
Tiempo despierto / tiempo dormido	0.351

Tabla 2.4: Características de la población de estudio utilizada para la SHHS 1.

nidos mediante estudios de PSG que fueron adquiridos por dispositivos domiciliarios y bajo la supervisión de un técnico especializado.

La base de datos SHHS forma parte de un estudio cuyo objetivo fue determinar la relación entre los trastornos respiratorios del sueño y las patologías cardiovasculares asociadas, evaluando su asociación con el riesgo de enfermedad coronaria, infartos, hipertensión sistémica y otras patologías. Para ello se realizaron más de 5000 estudios de PSG a pacientes que fueron incluidos en los criterios de admisión. De esta forma, se obtuvo un primer conjunto de estudios de PSG –el cual es identificado como SHHS 1– que representa una referencia o punto de partida. Luego, durante varios años se les realizó un seguimiento a los pacientes participantes, evaluando su estado de salud y registrando todos los tratamientos y eventos asociados. Finalmente, varios años más tarde, se volvió a registrar una PSG sobre estos sujetos para conformar un segundo conjunto de registros –la SHHS 2–. Dado que no todos los sujetos volvieron a repetir el estudio, la SHHS 2 alcanza la suma de 4080 estudios de PSG, esto es, aproximadamente 1000 estudios menos que su antecesora.

La SHHS fue registrada mediante un equipo de PSG *P series Sleep Monitoring System*, fabricado por Compumedics. Mediante este equipo se configuraron 12 canales para adquirir las señales de oximetría de pulso, del movimiento de la pared del pecho y del abdomen, de flujo nasal/oral, de posición del cuerpo, de EEG –dos derivaciones, de las cuales una cumple la función de respaldo en caso de que la otra falle o tenga mala calidad–, de EOG bilateral, de EMG registrado en la pera y de ECG. El oxímetro de pulso utilizado provee las dos señales que ya hemos mencionado, la HR y la SpO<sub>2</sub>, además de brindar una señal asociada a la calidad del registro que provee información respecto al estado de la conexión del sensor, tomando el valor 0 o 1 dependiendo de si la conexión es defectuosa o tiene buena calidad.

Los estudios adquiridos en esta base de datos fueron analizados bajo los estándares publicados en los manuales de etiquetado del sueño y eventos respiratorios asociados, confeccionados por R %K y la AASM [8, 16]. Para ello, en primer lugar se realizó etiquetado automático mediante la utilización de un *software* que luego fue revisado y, de ser necesario, modificado manualmente por expertos. En las referencias de la SHHS puede encontrarse una descripción exhaustiva respecto a este conjunto de señales y al estudio en general [73, 74].

Para los desarrollos que serán descriptos en esta tesis hemos **trabajo** únicamente con 5000 pacientes pertenecientes a la SHHS 1. No se tuvo en cuenta la SHHS 2 dado que utilizar ambas bases de datos introduciría una repetición de pacientes, lo cual podría ser perjudicial para la evaluación de los algoritmos, dado que, en general, éstos se comportan mejor sobre sujetos que ya han sido «vistos» por el algoritmo que sobre pacientes nuevos. A pesar de esto, la SHHS 1 contiene una cantidad de registros suficientemente grande para evaluar correctamente los algoritmos.

Como entrada de los algoritmos desarrollados se han utilizado únicamente las señales

adquiridas mediante el oxímetro de pulso, cuyo modelo es Nonin XPOD Model 3011. Este oxímetro estima la frecuencia cardíaca con una frecuencia de muestreo de 1 Hz y una precisión de  $\pm 3$  latidos por minuto, mientras que la señal de  $\text{SpO}_2$  posee una frecuencia de muestreo también de 1 Hz, resolución de 1 % y exactitud de  $\pm 2$  en el rango comprendido entre 70 % y 100 %.

Para todos los algoritmos desarrollados se realizó un preprocesamiento particular de acuerdo a las necesidades del método. Estos preprocesamientos serán descritos más adelante, al momento de tratar las descripciones de los desarrollos. Sin embargo, para todos los casos se realizó una misma preparación de la señal con el objetivo de eliminar los segmentos inválidos del registro obtenido por el oxímetro. Para ello, en primer lugar se utilizó la señal complementaria de calidad provista por el oxímetro para identificar estos segmentos. Luego, una vez identificados, los datos inválidos fueron reemplazados mediante una interpolación lineal entre las estimaciones confiables previa y posterior.

En la tabla 2.4 se resumen las principales características generales de la población que conforma la SHHS 1.

## Capítulo 3

# Desarrollo de un algoritmo basado en métodos clásicos

En este capítulo abordaremos el primer desarrollo generado en el marco de esta tesis con el objetivo de clasificar las fases del sueño en despierto y dormido a partir de las señales de HR obtenidas mediante un oxímetro de pulso. Al ser nuestro primer trabajo en el área, está basado en un abanico de herramientas clásicas ampliamente aceptadas dentro del ámbito del aprendizaje maquina. Si bien los desempeños obtenidos por este método serán luego superados con creces por los siguientes, es difícil para el autor pensar la existencia de estos últimos sin la experiencia adquirida a partir de este primer ensayo, no sólo en términos de métodos y prácticas recomendadas en el área sino también en relación a la exploración de los procesos fisiológicos y los cambios de dinámica de las señales asociadas. Es decir, al estar basado en métodos clásicos, para implementar este algoritmo ha sido necesario un análisis intensivo de la dinámica de la señal y de los cambios manifestados en las diferentes fases del sueño de forma de extraer las características de la señal que mejor permitan representar esas modificaciones. Para realizar esta extracción de características, se estudió una vasta cantidad de medidas y métodos propios del procesamiento de señales, haciendo una evaluación de su potencial discriminativo. Finalmente se exploraron alternativas para la selección de un subconjunto de características óptimas asociadas a diferentes clasificadores.

Podemos decir que la línea evolutiva entre este algoritmo y los siguientes representa de modo particular los cambios generados en el área del aprendizaje maquina con el advenimiento del aprendizaje profundo. Hasta ese momento era necesario un estudio exhaustivo y una comprensión cabal del problema para su correcto abordaje. De esta forma, los autores de algoritmos de procesamiento de señales y aprendizaje maquina necesitaban adquirir una experiencia importante en el área de la señal a abordar. A esto se le suma el hecho de que muchas de las herramientas clásicas permiten realizar una trazabilidad de la clasificación, pudiendo evaluarse qué tuvo más peso en la decisión final. Todo esto establecía una suerte de círculo virtuoso de conocimiento de la aplicación abordada que, a partir del desarrollo del aprendizaje profundo, entró en declive, aunque siguen demostrando su utilidad en conjuntos de relativamente pocos datos donde los algoritmos de aprendizaje profundo no son aplicables. Lo que no entró en declive, sin embargo, fueron los desempeños obtenidos que no pararon de mejorar con la evolución de los métodos de aprendizaje maquina. Por estas razones, aunque luego nos inclinemos por la mejora del desempeño –que en definitiva es lo que nos importa–, el desarrollo de este algoritmo ha sido un puntapié inicial sumamente beneficioso para el posterior abordaje del problema tratado en esta tesis.

Como hemos mencionado en los capítulos anteriores, este algoritmo encuentra su origen como un intento de superar las desventajas que presentan los dispositivos de cribado de patologías del sueño al no poder estimar el TTS y, por lo tanto, subestimar el IAH –o en su

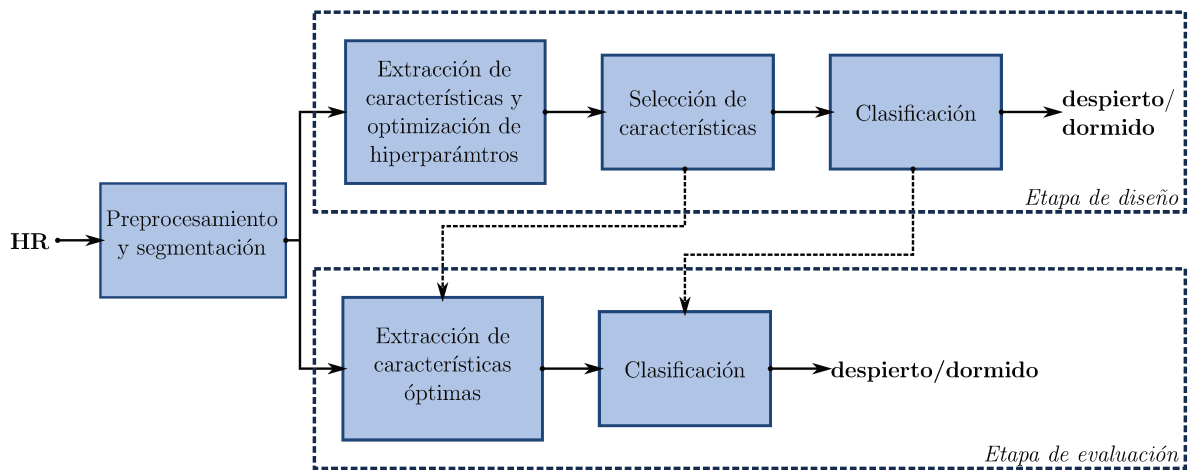


Figura 3.1: Esquema del algoritmo desarrollado. En la etapa de diseño se realiza una extracción y selección de características. En la etapa de evaluación, el sistema diseñado es evaluado con nuevos datos.

defecto el IDH-. Para ello, el algoritmo clasifica segmentos de la señal de HR en despierto y dormido a partir de un conjunto de características extraídas de la señal. Para esta etapa de extracción de características se han utilizado herramientas de la teoría de la información, métodos para análisis de frecuencia y de tiempo-frecuencia, además de algunos estadísticos clásicos. Una vez extraídas estas características, las cuales se supone que tienen potencial para discriminar las diferentes dinámicas presentes en la señal de HR correspondientes a despierto y dormido, hemos aplicado un esquema de selección de características en conjunto con la clasificación de los segmentos. Finalmente, el sistema óptimo –subconjunto de características y clasificador– fue testeado utilizando pacientes diferentes a aquellos con los que se diseñó el sistema.

A continuación describiremos el método desarrollado y las herramientas utilizadas, para luego presentar y discutir los desempeños obtenidos.

### 3.1. Metodología

En la figura 3.1 se muestra un diagrama en bloques del algoritmo. En primer lugar, las señales de HR son preprocesadas y segmentadas en ventanas de longitud  $L$  para los 5000 pacientes que conforman la SHHS 1. Luego, la base de datos se divide en dos subconjuntos de forma aleatoria: 500 registros serán utilizados para optimizar el diseño del algoritmo y realizar la selección de características (parte superior de la figura), y los restantes 4500 sujetos se utilizarán para entrenar y testear el clasificador «óptimo» (parte inferior de la figura).

En la primera etapa, que llamaremos «etapa de diseño», realizamos una extracción de características para cada segmento, optimizando sus hiperparámetros con el objetivo de maximizar el área bajo la curva ROC (AUC) [75]. Además, estas características se estandarizaron para tener media cero y varianza unitaria. Luego, los métodos de selección de características se llevaron a cabo en conjunto con la clasificación –esto implica la selección de un método de selección de características y, además, la selección de un clasificador con el que se utilizará ese método–. Se consideraron dos aproximaciones posibles: (1) selección de características en avance (*forward feature selection*, FFS) y SVM y (2) método de selección de variables (*variable selection*) propio del clasificador RF. Para validar el desempeño de las técnicas propuestas se realizó una validación cruzada (*k-fold cross validation*, *k-fold CV*).

El sistema «óptimo» obtenido en la etapa de diseño se entrenó y validó en la siguiente

etapa, denominada «etapa de evaluación», con los 4500 sujetos restantes que no habían sido utilizados hasta ahora. Nuevamente se utilizó *k-fold* CV para validar el desempeño obtenido.

### Selección de $k$

No existe una regla formal para la selección de  $k$  siempre y cuando el tamaño de la base de datos permita obtener  $k$  particiones con un número suficiente de elementos como para calcular estadísticos confiables. A medida que incrementamos  $k$ , la diferencia entre los diferentes subconjuntos de entrenamiento se vuelve menor, por lo que el sesgo disminuye (el sesgo es menor para  $k = 10$  que para  $k = 5$ ) [76].

Duda y cols. [77] han establecido una serie de reglas heurísticas para seleccionar la porción  $\gamma$  de la base de datos para ser utilizada como conjunto de validación, donde  $0 < \gamma < 1$ . En la mayoría de los casos se utiliza una porción menor de la base de datos para validar los algoritmos ( $\gamma < 0.5$ ) porque este conjunto se utiliza sólo para establecer las propiedades globales del clasificador –cuando dejar de ajustar sus parámetros–, mientras que el conjunto de entrenamiento es utilizado para aprender un gran conjunto de parámetros, por lo que suele necesitar ser mayor. En general, es habitual seleccionar por defecto  $\gamma = 0.1$ , por lo tanto  $k = 10$ , ya que es un valor que ha probado ser efectivo en muchas aplicaciones. En general, cuando el número de grados de libertad de un clasificador es pequeño en relación a la cantidad de elementos de la base de datos utilizados para entrenar el algoritmo, el error de generalización es relativamente insensible a la selección de  $\gamma$  y, por ende, a la de  $k$  [77].

#### 3.1.1. Preprocesamiento

Además del preprocesamiento general descrito en el capítulo anterior –a partir del cual se reemplazaron las porciones de señal defectuosas mediante una interpolación lineal–, la señal de HR se estandarizó con el objetivo de reducir la variabilidad entre sujetos, haciendo su media global igual a 0 y su desvío estándar igual a 1. Luego, las señales fueron partidas en segmentos de longitud  $L$ , variando esta longitud desde  $L = 30$  hasta  $L = 300$  en pasos de 30 y generando, así, 10 bases de datos posibles con diferentes longitudes. Esto se efectuó para evaluar cómo varía el desempeño del sistema al considerar diferentes longitudes de segmentos. Sólo se conservaron los segmentos que se correspondían en su totalidad a un único estado, ya sea despierto o dormido.

#### 3.1.2. Extracción de características

La información relevante contenida en la señal de HR fue resumida en un total de 70 características que fueron extraídas de la señal. En esta sección abordaremos los principales aspectos teóricos y prácticos de cada una de estas características.

### Entropía aproximada

La entropía aproximada (*approximate entropy*, ApEn) fue introducida por Pincus en el año 1991 como una medida de regularidad de los datos [78]. Una mayor irregularidad en los datos –de la señal– produce valores más altos de ApEn, mientras que ante menores irregularidades ocurre lo contrario. Para una serie temporal  $N$ -dimensional, la ApEn depende de tres hiperparámetros: la dimensión de inmersión  $m$ , el retardo de inmersión  $\tau$  y el umbral  $r$ . La ApEn ha sido utilizada ampliamente como una característica no lineal para clasificar diferentes dinámicas.

Para una definición formal, sea  $x[n]$  una serie temporal de longitud  $N$ . Luego, pueden construirse  $M = N(m-1)\tau$  vectores de estado haciendo  $\mathbf{x}_i^m = [x[i], x[i+\tau], \dots, x[i+(m-1)\tau]]$ , donde  $i = 1, 2, \dots, M$  [79]. Finalmente, la ApEn se define como sigue [78]:

$$\text{ApEn}(m, \tau, r, N) = \phi^m(r) - \phi^{m-1}(r), \quad (3.1)$$

donde

$$\phi^m(r) = \frac{1}{M} \sum_{i=1}^M \ln \frac{1}{M} \sum_{j=1}^M \theta(d(\mathbf{x}_i^m, \mathbf{x}_j^m), r), \quad (3.2)$$

siendo  $d(\cdot)$  una medida de distancia entre los vectores de estado y  $\theta(\cdot)$  una función núcleo (*kernel*). Las medidas de distancia más utilizadas son la norma Euclídea y la norma del máximo. Asimismo, las funciones núcleo más usadas son la función escalón de Heaviside [78] y los núcleos Gaussianos [80].

Para este trabajo se utilizó la norma Euclídea junto con un núcleo Gaussiano. Además, se extrajeron tres características relacionadas con la ApEn. La primera de ellas es el valor de ApEn estimado con los hiperparámetros  $r$ ,  $\tau$  y  $m$  que maximizan el AUC. Las otras dos características son medidas propuestas por Restrepo y cols. [81] en el año 2014 para detectar cambios en la regularidad, seleccionando el valor máximo de ApEn y el valor  $r$  en el cual se localiza este máximo. Al igual que en el caso anterior, tanto  $m$  como  $\tau$  fueron seleccionados de forma de maximizar el AUC.

### Entropía muestral

La ApEn es un estimador que presenta un importante sesgo, ya que en la sumatoria de la ecuación 3.2 incluye el término correspondiente a la comparación de distancia del punto con sí mismo, donde siempre existe una coincidencia porque la distancia es cero. Es decir, no excluye de la sumatoria el término cuando  $i = j$ . Por ejemplo, para el caso de una función núcleo de Heaviside se tiene  $\theta(d(\mathbf{x}_i^m, \mathbf{x}_j^m), r) = H((d(\mathbf{x}_i^m, \mathbf{x}_j^m) - r))$ , siendo  $H$  la función escalón de Heaviside. Este sesgo es más evidente cuanto menor es la longitud de la secuencia.

Para superar esta limitación, en los comienzos del siglo Richman y Moorman propusieron la entropía muestral (*sample entropy*, SampEn), cuyo valor es independiente de la longitud de la secuencia y presenta una consistencia mayor que la evidenciada por la ApEn [82]. Siguiendo una notación similar a la utilizada para las ecuaciones de ApEn, la SampEn puede estimarse haciendo:

$$\text{SampEn}(m, \tau, r, N) = -\ln \frac{B^{m+1}(r)}{B^m(r)}, \quad (3.3)$$

con

$$B^m(r) = \frac{1}{M} \sum_{i=1}^M \frac{1}{M-1} \sum_{j=1, i \neq j}^M \theta(d(\mathbf{x}_i^m, \mathbf{x}_j^m), r), \quad (3.4)$$

donde nuevamente  $d(\cdot)$  es una medida de distancia entre los vectores de estado y  $\theta(\cdot)$  una función núcleo. Para este caso se seleccionó la norma Euclídea y la función escalón de Heaviside.

Para valores de  $r$  altos,  $\text{SampEn} \rightarrow \text{ApEn}$ , mientras que para valores pequeños de  $r$  la SampEn presenta una alta varianza. Tal y como han propuesto Richman y Moorman [82], para calcular  $B^m(r)$  hemos considerado únicamente los primeros  $N - m$  vectores de longitud  $M$ , asegurando de esta forma que para  $1 \leq i \leq N - m$  estén definidos  $\mathbf{x}_i^m$  y  $\mathbf{x}_i^{m+1}$ .

**Similaridades y diferencias entre SampEn y ApEn.** Muy a menudo las medidas de ApEn y SampEn son confundidas, entendiéndose que la última es simplemente una versión «mejorada» de la primera. Esta afirmación es correcta, pero en forma parcial, ya que ambas medidas presentan diferencias importantes en su origen. Sobre estas diferencias hablaremos en este apartado con el objetivo de entender por qué hemos planteado la utilización de ambas medidas y no simplemente utilizamos la «más nueva».

La ApEn tuvo su origen en la entropía de Kolmogorov-Sinai, publicada en el año 1958 [83]. Esta entropía está definida como la tasa media de información generada por un proceso. Luego, a comienzos de la década del 90 Pincus presentó la ApEn como un estadístico para evaluar la complejidad a partir de datos cuya longitud estaba limitada y en presencia de ruido. Para ello se basó en el algoritmo de Echmann y Ruelle [84]. Como mencionamos, esta medida introducía un sesgo, dado que contabilizaba el término correspondiente a la distancia de un dato con sí mismo para evitar la posibilidad de que el logaritmo presente en la ecuación tomase el valor cero. Por esta razón, la estimación propuesta por Pincus no cumplía con algunas propiedades esperadas. Según Richman y Moorman, la ApEn era altamente dependiente de la longitud de la secuencia, presentando un valor uniformemente inferior que el esperado cuando la longitud era corta. Por otro lado, la estimación obtenida presentaba una falta de consistencia: si el valor de ApEn para un conjunto de datos es mayor que para otra, éste debería permanecer mayor para todas las condiciones en las que se realicen los testeos. Sin embargo, esto no sucedía. Según estos autores, esta última limitación es particularmente importante dado que la ApEn ha sido recomendada en forma repetida como una medida para comparar conjuntos de datos [82].

Para superar estas limitaciones, aproximadamente una década después de la propuesta realizada por Pincus, Richman y Moorman propusieron la SampEn como una medida de regularidad más consistente basada en Grassberger y Procaccia [85]. De esta forma se lograba una estimación con mayor independencia de la longitud de los datos. Sin embargo, ambas medidas continúan siendo altamente dependientes del conjunto de parámetros seleccionados.

Granero-Belinchon y cols. [86] han explicado esta diferencia en la convergencia de ApEn y SampEn. Mientras que ApEn es un estimador de orden  $m$  de la entropía de Kolmogorov-Sinai  $h_{KS}$  de resolución finita  $\epsilon$ , es decir

$$\lim_{\epsilon \rightarrow 0} \lim_{m \rightarrow \infty} \text{ApEn} = h_{KS} \quad (3.5)$$

SampEn es interpretado como un estimador de orden  $m$  de la entropía de Eckmann-Ruelle  $K_2$  de resolución finita  $\epsilon$ ,

$$\lim_{\epsilon \rightarrow 0} \lim_{m \rightarrow \infty} \text{SampEn} = K_2. \quad (3.6)$$

A pesar de las diferencias y similitudes presentes en estas dos estimaciones de entropía, en este trabajo ambas han sido utilizadas como características potenciales para discriminar las fases de despierto y dormido. En términos prácticos únicamente estamos interesados en su poder discriminativo y es por esta razón que, a pesar de sus similitudes teóricas –aunque se ha visto que una no es simplemente una versión «mejorada» de la otra–, hemos evaluado la utilidad de ambas medidas.

**Parámetros de ApEn y SampEn.** En los trabajos iniciales, tanto Pincus como Richman y Moorman establecieron ciertos valores de los parámetros para calcular las entropías –por ejemplo  $m = 2$ ,  $\tau = 1$  y  $r = 0.1$  veces el desvío estándar de la secuencia– [78, 82]. Con el paso del tiempo, se volvió habitual la utilización de esos mismos valores por defecto, sin tener en cuenta que los parámetros óptimos podían variar para diferentes aplicaciones.

Tal es así, que Chon y col. [87] encontraron que ni la ApEn ni la SampEn eran exactas para medir la complejidad de una señal cuando se realizaban los cálculos con los parámetros

«recomendados». Por otro lado, Restrepo y col. [81] propusieron utilizar como una característica para propósitos de clasificación el valor máximo de ApEn –en función de  $r$ – combinado con el parámetro  $r$  en el cual se obtenía este máximo. En este primer trabajo nosotros hemos utilizado las características propuestas por Restrepo y, además, hemos realizado una búsqueda de parámetros para encontrar el conjunto de parámetros óptimos que mejor discriminaba los dos estados a ser clasificados.

### Entropía difusa

La entropía difusa (*fuzzy entropy*, FuzEn) es una medida análoga a la SampEn, pero en este caso el grado de similitud –es decir la función núcleo  $\theta(\cdot)$ – es calculado mediante una función definida como  $e^{-d(\mathbf{x}_i^m, \mathbf{x}_j^m)^q/r}$ . Según Chen y col. el estimador FuzEn presenta mayor consistencia y es menos dependiente de la longitud de los datos que SampEn [88].

### Entropía de dispersión

La familia de entropías que han sido presentadas hasta ahora son herramientas muy poderosas para evaluar las características dinámicas de una serie temporal, pero presentan la complicación de ser muy costosas en términos computacionales. Existe otro tipo de entropía que fue propuesta por Band y Pompe, denominada entropía de permutación (*permutation entropy*, PermEn) que permite cuantificar la irregularidad de una serie temporal basándose en el análisis de los patrones de permutación [89]. Estos patrones dependen de la comparación de un dato con sus vecinos, evaluando su magnitud. Sin embargo, esta comparación no considera la diferencia que existe en las amplitudes, sino que simplemente asigna un patrón de permutación de acuerdo a cómo se ordenan sus magnitudes. Por ejemplo, para esta entropía el patrón de permutación es el mismo para los segmentos [1, 3, 2] que para [1, 300, 2].

Con el objetivo de obtener una medida rápida de calcular, pero que a la vez presente una mayor consistencia en la caracterización de las dinámicas de una serie temporal, Rostaghi y Azami propusieron la entropía de dispersión (*dispersion entropy*, DispEn), cuyo cálculo para una serie temporal  $x[n]$  de longitud  $N$  incluye una secuencia de pasos que se detallarán a continuación.

Primero,  $x[n]$  es asignado a  $c$  clases. Para ello,  $x[n]$  es transformado en la serie  $y[n]$  mediante una función de distribución acumulada. Luego, cada elemento de  $y[n]$  es asignado a un entero entre 1 y  $c$  haciendo  $z[n] = \text{round}(cy[n] + 0.5)$ , con  $n = 1, 2, \dots, N$ . Finalmente, dado una dimensión de inmersión  $m$  y un retardo  $\tau$ , se reconstruyen los vectores de estado haciendo  $\mathbf{z}_i^{m,c} = [z^c[i], z^c[i + \tau], \dots, z^c[i + (m - 1)\tau]]$ .

Segundo, se asigna un patrón de dispersión  $\pi_{\nu_0, \nu_1, \dots, \nu_{m-1}}$  a cada serie temporal  $\mathbf{z}_i^{m,c}$ , donde  $z^c[i] = \nu_0$ ,  $z^c[i + \tau] = \nu_1$ ,  $\dots$ ,  $z^c[i + (m - 1)\tau] = \nu_{m-1}$ . El número de patrones de dispersión posibles que puede ser asignado a cada serie temporal  $\mathbf{z}_i^{m,c}$  es igual a  $c^m$ , donde cada señal tiene  $m$  elementos y cada uno de ellos puede tomar valores enteros desde 1 hasta  $c$ .

Tercero, se calcula la frecuencia relativa de cada uno de los  $c^m$  posibles patrones de dispersión haciendo

$$p(\pi_{\nu_0, \nu_1, \dots, \nu_{m-1}}) = \frac{\#\{i | i \leq N - (m - 1)\tau, \mathbf{z}_i^{m,c} \text{ es de tipo } \pi_{\nu_0, \nu_1, \dots, \nu_{m-1}}\}}{N - (m - 1)\tau} \quad (3.7)$$

es decir que  $p(\pi_{\nu_0, \nu_1, \dots, \nu_{m-1}})$  representa el número de patrones de dispersión  $\pi_{\nu_0, \nu_1, \dots, \nu_{m-1}}$  que son asignados a  $\mathbf{z}_i^{m,c}$  dividido por el número total de vectores de estado con dimensión de inmersión  $m$ .

Finalmente, basados en la definición de entropía de Shanon, la DispEn con dimensión de inmersión  $m$ , retardo  $\tau$  y número de clases  $c$  se estima haciendo



$$\text{DispEn}(x, m, c, \tau) = - \sum_{\pi=1}^{c^m} p(\pi_{\nu_0, \nu_1, \dots, \nu_{m-1}}) \ln p(\pi_{\nu_0, \nu_1, \dots, \nu_{m-1}}) \quad (3.8)$$

### Extensión de las medidas de entropía a los dominios tiempo-frecuencia $(t, f)$ o tiempo-escala $(t, s)$

Podemos obtener una medida de la uniformidad de la distribución de energía de un señal en el dominio frecuencial interpretando la densidad espectral de potencia (*power spectral density*, PSD) como una función de distribución de cuasi-probabilidades –modificando la amplitud de la PSD de forma tal que cumpla las propiedades de una PDF– para luego utilizar los conceptos de entropía [90]. De esta manera, un valor de entropía alto en el dominio frecuencial implica una mayor uniformidad del espectro de potencia de la señal –potencia similar para todos los valores de frecuencias–, mientras que un valor bajo implica lo contrario. Estos conceptos pueden extenderse hacia los dominios  $(t, f)$  o  $(t, s)$  con el objetivo de lograr discriminar señales con anchos de banda similares, pero cuya variación en el tiempo difiere [91].

Sea  $x[n]$  una serie real de longitud  $N$  y sea  $z[k]$  su transformada de Fourier (*Fourier transform*, FT) de longitud  $M$ . La entropía espectral (*spectral entropy*, SE) está definida como:

$$\text{SE}_{(f)} = - \sum_{k=1}^M \mathcal{Z}[k] \ln(\mathcal{Z}[k]), \quad (3.9)$$

donde  $\mathcal{Z}[k] = \frac{|z[k]|^2}{\sum_{k=1}^M |z[k]|^2}$ .

La entropía de Shanon en  $(t, f)$  o  $(t, s)$  es una extensión de la SE a estos dominios. Se obtiene reemplazando la FT por una distribución tiempo-frecuencia o tiempo-escala, que llamaremos  $\rho[n, k]$ . Entonces, la entropía  $(t, s)$  está definida como:

$$\text{SE}_{(t,s)} = - \sum_{n=1}^N \sum_{k=1}^M \rho_N[n, k] \ln(\rho_N[n, k]), \quad (3.10)$$

donde  $\rho_N[n, k] = \frac{\rho[n, k]}{\sum_n \sum_k \rho[n, k]}$ .

Estas ideas pueden extenderse también a entropías como la de Renyi haciendo:

$$\text{RE}_{(t,s)} = - \frac{1}{1-q} \ln \sum_{n=1}^N \sum_{k=1}^M \rho_N[n, k]^q. \quad (3.11)$$

En este trabajo, siguiendo las recomendaciones de Boashash, seleccionamos  $q = 3$  [91].

Por otro lado, relacionando estos conceptos con las entropías multiresoluciones [92], nosotros propusimos una medida de entropía a través de las escalas. En este sentido, para cada escala  $k$  de la representación  $\rho[n, k]$  calculamos un valor de entropía haciendo:

$$\text{RE}_{(t,s)}(k) = - \frac{1}{1-q} \ln \sum_{n=1}^N \left( \frac{\rho[n, k]}{\sum_n \rho[n, k]} \right)^q. \quad (3.12)$$

Esta medida de entropía permite estimar la uniformidad de la energía de una señal dentro de una escala dada. Su valor es máximo cuando la energía es constante a lo largo del tiempo.

Con el objetivo de satisfacer las propiedades de una función de densidad de probabilidades fue necesario normalizar  $\rho[n, k]$ . De esta forma, la entropía no varía con la magnitud de la energía, sino que su valor sólo depende de la distribución de energía a lo largo del tiempo. Como en el caso previo, establecimos  $q = 3$ . En lo que resta del documento nombraremos este

conjunto de características como entropía de Renyi multirresolución en el dominio tiempo-escala (*time-scale multiresolution Renyi entropy*, TSMRE).

En este trabajo utilizamos la transformada ondita continua (*continuous wavelet transform*, CWT) para obtener la distribución de tiempo-escala  $\rho[n, k]$ , empleando 32 escalas y una ondita de tipo Haar. Aunque esta familia de onditas es muy simple y, además, tiene la desventaja de ser discontinua y por lo tanto no derivable, para este trabajo fue la que obtuvo el mejor desempeño. Esto puede ser explicado porque la estimación de HR que realiza el oxímetro de pulso tiene una alta cuantización, presentando importantes saltos – discontinuidades – en estimaciones sucesivas. Es por ello que una función constante por partes permite una buena aproximación de la señal.

### Complejidad de Lempel-Ziv

La complejidad de Lempel-Ziv (LZ) es una medida no paramétrica y muy simple de calcular, propuesta en el año 1976 y cuyo uso se ha extendido con el paso del tiempo a señales biológicas con el propósito de reconocer diferencias en la regularidad. Para su obtención, en primer lugar, transformamos una señal discreta  $x[n]$  en una secuencia de símbolos  $\mathbf{P}$ . La secuencia de símbolos<sup>1</sup>  $\mathbf{P}$  habitualmente es obtenida mediante la comparación con uno o más umbrales. Como no se involucran operaciones matemáticas, este símbolo puede ser cualquier cosa. Habitualmente se seleccionan números enteros o letras. En general, en el contexto del análisis de señales biomédicas se utiliza la mediana para binarizar la señal, asignando un símbolo a los elementos de la señal que están por debajo de la mediana y otro símbolo los que están por encima. En nuestro trabajo se obtuvieron mejores resultados utilizando dos umbrales definidos por los percentiles 0.33 y 0.66, generando secuencias que pueden obtener tres símbolos diferentes –por ejemplo 0, 1 y 2 o  $A$ ,  $B$  y  $C$ –, es decir:

$$\mathbf{P} = s_1, s_2, \dots, s_n, \quad (3.13)$$

donde

$$s_i = \begin{cases} 0, & \text{si } x[i] < P_{33} \\ 1, & \text{si } P_{33} < x[i] < P_{66} \\ 2, & \text{si } x[i] \geq P_{66}. \end{cases} \quad (3.14)$$

Luego, la medida de complejidad  $c[n]$  puede obtenerse siguiendo el algoritmo que se detalla en [93].

Todas las medidas descritas hasta ahora permiten cuantificar la regularidad de los datos. Como hemos mencionado en capítulos anteriores, las señales de HR presentan una mayor variabilidad y valor medio durante la vigilia, decreciendo luego desde las fases de sueño liviano hacia las de sueño profundo, para volver a incrementarse durante el sueño REM. Por esto, todas las medidas presentadas hasta ahora fueron propuestas con el objetivo de explotar esta diferencia en la regularidad –o complejidad– de los datos durante las diferentes fases del sueño. Estas medidas, que ya habían probado su utilidad en varios trabajos utilizando señales biomédicas [94], también resultaron apropiadas para la aplicación abordada en esta tesis.

### Características basadas en el dominio frecuencial

Está ampliamente estudiado que el análisis espectral de las señales de variabilidad de HR permite cuantificar y evaluar la influencia del sistema nervioso autónomo en los procesos

<sup>1</sup>Cuando hablamos de símbolos estamos haciendo referencia a cualquier abstracción con la que se representa el conjunto de valores que quedan por encima o por debajo del umbral.

vitales [95]. Teniendo esto en consideración, se extrajo de la señal de HR varias características clásicas utilizadas bajo la idea de que portarán información relativa a las fases del sueño [96]. Por esta razón, se calculó la potencia en los siguientes anchos de banda: muy baja frecuencia (*very low frequency*, VLF) –potencia comprendida en el intervalo (0.003, 0.04] Hz–, baja frecuencia (*low frequency*, LF) –potencia comprendida en el intervalo (0.04, 0.15] Hz– y alta frecuencia (*high frequency*, HF) –potencia comprendida en el intervalo (0.15, 0.4] Hz–. Además, se obtuvieron las relaciones LF/HF, HF y VF normalizadas y la potencia total TP. La estimación espectral necesaria para estas medidas se obtuvo mediante el periodograma de cada señal.

### Características basadas en los dominios $(t, f)$ y $(t, s)$

Se extrajeron 15 características que son discutidas en el artículo de Boashash y col. [97] publicado en el año 2011, propuestas para el análisis de anomalías en los EEG de neonatos. Mediante estas medidas es posible caracterizar la naturaleza no estacionaria de la señal de HR, por lo cual son potencialmente útiles para discriminar las fases del sueño. A continuación se presentará una breve descripción de estas características.

Muchas de las características presentadas por el equipo de Boashash están basadas en la descomposición de valores singulares (*singular value decomposition*, SVD) de la representación  $(t, f)$  o  $(t, s)$ , simbolizada como  $\rho[n, k]$ . El método de SVD factoriza la matriz  $\rho[n, k]$ , de tamaño  $M \times N$  en dos subespacios haciendo  $\rho[n, k] = \mathbf{U}\mathbf{S}\mathbf{V}^H$ , donde  $\mathbf{U}$  y  $\mathbf{V}$  son matrices ortogonales unitarias de dimensión  $N \times N$  y  $M \times M$ , respectivamente.  $\mathbf{S}$  es una matriz diagonal de dimensión  $N \times M$ , cuya diagonal principal toma valores reales no negativos que son conocidos como valores singulares de  $\rho[n, k]$ .

Las primeras dos características consideradas son el valor máximo – $\max(\text{diag}(\mathbf{S}))$ – y la varianza – $\text{var}(\text{diag}(\mathbf{S}))$ – de los valores singulares de  $\rho[n, k]$ . La tercera característica es una medida de complejidad que se calcula haciendo:

$$E_{\text{SVD}} = - \sum_{i=1}^N \bar{S}_i \ln \bar{S}_i \quad (3.15)$$

donde  $\bar{S}_i = \frac{S_i}{\sum_{j=1}^N S_j}$ .

La cuarta característica es una medida de la concentración de energía, definida como:

$$\text{ECM} = \left( \sum_{n=1}^N \sum_{k=1}^M |\rho[n, k]|^{\frac{1}{2}} \right)^2 \quad (3.16)$$

Además se obtuvieron 8 características relacionadas con la energía sub-banda que se obtiene al integrar a lo largo del tiempo de la siguiente manera:

$$\text{SBE}_\delta = \sum_{n=1}^N \sum_{k=\delta L+1}^{(d+1)L} \rho[n, k], \quad (3.17)$$

donde  $L = \frac{M}{8}$  y  $\delta = 0, 1, \dots, 7$ .

Por último, además de todas estas características, se calcularon la media y la desviación estándar en el dominio  $(t, s)$  como una extensión de los estadísticos clásicos, además de la entropía de Renyi en el dominio  $(t, s)$  según la ecuación 3.11.

### Características basadas en la autocorrelación

Al analizar la serie de autocorrelación de los segmentos de la señal de HR, se observó que aquellos que pertenecían a la categoría de despiertos tenían una mayor periodicidad en

la señal y eran más suaves que los pertenecientes a dormido. Por este motivo, se extrajo un conjunto de características que están relacionadas con la serie de autocorrelación con el objetivo de introducir la información que aportaban para diferenciar entre las clases.

En primer lugar, se calculó el primer mínimo y el primer cruce por cero de la serie.

En segundo lugar, se ajustó un modelo autorregresivo (*autorregresive model*, AR) de orden 4 a los segmentos de la señal, utilizando los coeficientes estimados que caracterizaban la serie de autocorrelación como características.

Finalmente, se calculó la complejidad de LZ en la serie de autocorrelación con el objetivo de obtener una medida de las diferencias en la regularidad de las series correspondientes a despierto y dormido.

### Características basadas en estadísticos clásicos

Además de todas las características que hemos mencionado hasta ahora, se calcularon dos estadísticos clásicos: la media y el desvío estándar del segmento de la señal. Estas características –especialmente la media– son particularmente útiles para discriminar las fases del sueño, ya que varía significativamente entre los estados de despierto y dormido.

### Resumen de las características

En resumen, para su posterior selección y clasificación se extrajeron 70 características, entre las que se encuentran:

- 7 variantes de medidas de entropía y complejidad: ApEn, ApEn<sub>máx</sub>, r<sub>máx</sub>, SampEn, FuzEn, DispEn y LZ.
- 32 medidas de entropía en el dominio  $(t, s)$ : TSMRE.
- 7 características del dominio frecuencial: VLF, LF, HF, LF/HF, HF y VF normalizadas, y TP.
- 15 características en el dominio  $(t, s)$ : máx  $\mathbf{S}$ , var(diag( $\mathbf{S}$ )), E<sub>SVM</sub>, ECM,8 SBE <sub>$\delta$</sub> , mean <sub>$(t,s)$</sub> , std <sub>$(t,s)$</sub>  y RE <sub>$(t,s)$</sub> ,
- 7 características basadas en los coeficientes de autocorrelación: primer mínimo, primer cruce por cero, 4 coeficientes de AR y complejidad de LZ de la serie de autocorrelación.
- 2 estadísticos clásicos: media y desvío estándar.

### 3.1.3. Selección de características y clasificación

Es habitual que el desempeño de un clasificador varíe significativamente al considerar diferentes combinaciones de características. En general, añadir características que presentan información redundante –o características que no contienen información relevante– incrementa el costo computacional y hace más arduo el proceso de entrenamiento. Esto se suele conocer como la «maldición de la dimensionalidad»<sup>2</sup>. Muchas veces las características redundantes –o irrelevantes– conducen a un potencial deterioro en su desempeño, mientras que otras veces, dependiendo de las particularidades del clasificador considerado, las características «irrelevantes» pueden ser no tenidas en cuenta para tomar la decisión [98]. Para superar estos problemas es habitual en los sistemas clásicos de aprendizaje maquina introducir una

<sup>2</sup>En términos más estrictos, al tener más características incrementa la dimensión del vector de características. Para poder muestrear un espacio de dimensión cada vez mayor, entonces, la cantidad de datos necesarios se incrementa exponencialmente con la dimensión del vector de características.

rutina de selección de características que encuentre el subconjunto óptimo de características para un clasificador dado.

En el algoritmo presentado en este capítulo utilizamos dos esquemas diferentes para realizar la selección de características. En el primero de ellos utilizamos un «método de envoltura» (*wrapper method*) para la selección de características junto a un clasificador SVM [98]. En el segundo, en cambio, utilizamos un «método incrustado» (*embedding method*) propio del clasificador RF, denominado selección de variables. A continuación describiremos ambas propuestas.

### Selección de características en avance y SVM

Para implementar un método de selección de características, según Kohavi y John [99], es necesario definir cómo buscar el espacio de todos los subconjuntos posibles de características, qué medida de desempeño utilizar para guiar esta búsqueda y qué clasificador emplear. Mediante una búsqueda exhaustiva se puede encontrar el óptimo global, pero el problema es NP-*hard*. Si consideramos que es necesario entrenar el clasificador para cada una de las posibles combinaciones –teniendo en cuenta además que en nuestro caso en particular hay 70 características y 5000 pacientes totales–, podemos notar que es una opción viable sólo para algunos problemas sencillos y rápidos de evaluar. Para evitar este inconveniente, en este trabajo implementamos una solución de tipo «voraz» (*greedy solution*), la ya mencionada selección de características en avance (*forward feature selection*, FFS).

Para la  $n$ -ésima iteración, supongamos que  $\mathcal{S}_f^n$  es el conjunto de características óptimas seleccionadas hasta este momento y  $\mathcal{R}_f^n$  es el conjunto de características restantes. Sea  $N_f$  el número total de características y  $\varepsilon$  una medida del error. Sea, además,  $\mathcal{S}_f^1 = \emptyset$  y sea  $\mathcal{R}_f^1$  el conjunto de todas las características. El procedimiento FFS puede ser resumido de la siguiente manera:

- 1: **for**  $n = 1$  **to**  $N_f$  **do**
- 2:   **for**  $R_f \in \mathcal{R}_f^n$  **do**
- 3:     **Conjunto provisional de características:**  $\mathcal{P}_{R_f} = \mathcal{S}_f^n \cup \{R_f\}$
- 4:     **Validación cruzada:** Dividir en forma aleatoria los datos en  $K$  partes.
- 5:     **for**  $k = 1$  **to**  $K$  **do**
- 6:       Seleccionar el  $k$ -ésimo subconjunto para **testeo** y el resto para **entrenamiento**.
- 7:       **Entrenar** el clasificador con el conjunto de entrenamiento.
- 8:       **Testear** el clasificador con el conjunto de testeo.
- 9:       Calcular el **error**  $\varepsilon(k)$  para todo  $k$ .
- 10:    **end for**
- 11:    Estimar el error medio  $\bar{\varepsilon}(R_f)$  para todo  $\mathcal{P}_{R_f}$ .
- 12:    **end for**
- 13:    Encontrar el minimizador  $R_f^* = \arg \min_{R_f} \bar{\varepsilon}(R_f)$ .
- 14:    Actualizar el **mejor conjunto de características seleccionadas**  $\mathcal{S}_f^{n+1} = \mathcal{S}_f^n \cup \{R_f^*\}$ .
- 15:    Actualizar el **conjunto de características restantes**  $\mathcal{R}_f^{n+1} = \mathcal{R}_f^n - \{R_f^*\}$ .
- 16:    Guardar el mejor resultado  $\varepsilon_{\min}(n) = \bar{\varepsilon}(R_f^*)$  para todo  $n$ .
- 17: **end for**

En este algoritmo las características son incorporadas en forma progresiva en subconjuntos «anidados» de tamaño cada vez mayor. Para cada uno de los subconjuntos se obtiene una medida de error para luego seleccionar el conjunto «óptimo». A pesar de que el costo computacional de este método es menor que el que se obtiene al hacer una búsqueda exhaustiva, llegar a una solución óptima todavía puede ser muy lento.

Para este esquema hemos utilizado una SVM como clasificador, propuesta por Vapnik [100]. Es común afirmar que no existe un clasificador mejor que otro en términos generales, sino que su desempeño dependerá de cada problema. Si bien consideramos que esto es correcto, también tenemos que decir que hasta el surgimiento del aprendizaje profundo, la SVM era uno de los clasificadores más potentes a considerar, obteniendo excelentes resultados en una gran variedad de problemas. Además, al contar con una base teórica sólida y formal, sus resultados son completamente interpretables, lo que lo hace una solución atractiva para luego comprender el porqué de las clasificaciones realizadas.

La SVM involucran la optimización de una función objetivo convexa con restricciones que no es afectada por mínimos locales. Al optimizar esta función se encuentra un hiperplano, cuya distancia de separación con los puntos más cercanos de dos clases linealmente separables –conocido como margen– es la máxima posible. Estos puntos de ambas clases –definidos por los vectores de características– eventualmente pueden ser proyectados a un espacio de alta dimensionalidad. En nuestro trabajo, para proyectar los puntos a un espacio hiperdimensional hemos utilizado un núcleo gaussiano dado por  $K(x_i, x_j) = \exp\{-\gamma\|x_i - x_j\|^2\}$ . Por ser una técnica ampliamente conocida en el área del aprendizaje maquina no entraremos en detalles excesivos al respecto. Puede encontrarse una explicación más detallada en [100, 101].

Por último, la medida de error  $\varepsilon$  que hemos utilizado para guiar la búsqueda se seleccionó con el objetivo de maximizar el AUC, basándonos en la sensibilidad (Se) y especificidad (Sp). De esta forma, en cada iteración se seleccionó la característica que minimizaba la magnitud  $(1 - \text{Se})^2 + (1 - \text{Sp})^2$  –esto es la distancia al punto (0, 1) en la curva ROC [75]–.

Además de este criterio de uso más «general», realizamos algunos experimentos utilizando una medida «particular» relacionada con la aplicación bajo la hipótesis de que esto podría conducir a un mejor desempeño final del clasificador. Por ello, planteamos utilizar como medida de error  $\varepsilon$  la diferencia absoluta entre el TTS estimado y el TTS real para un paciente, y así optimizar en forma directa sobre la medida final que se utilizará para evaluar la bondad de los métodos desarrollados. Sin embargo, los experimentos desarrollados utilizando esta última medida no fueron conducentes, reportando un desempeño inferior. Por este motivo, se descartaron estos resultados.

### Selección de variables y RF

El otro esquema considerado para clasificar y seleccionar un conjunto óptimo de características es la utilización de RF y un método propuesto por Breiman para obtener un *ranking* de características según su importancia para el clasificador [102]. Este método de selección de variables puede ser incorporado en el clasificador RF directamente como parte del proceso de entrenamiento. Por este motivo, es mucho más eficiente dado que evita la necesidad de reentrenar varias veces el algoritmo como en el método anterior.

Para aplicar este método, una vez que el clasificador ha sido entrenado utilizando la totalidad de características, se procede a permutar en forma aleatoria los valores de las características, de a una por vez. Es decir, se desordenan los valores para una de las características de forma que el valor de dicha característica para un dato corresponda, en realidad, a otro dato. Luego de cada permutación se vuelve a efectuar la clasificación sin ningún reentrenamiento y se mide el incremento en el error. Esto se realiza para cada una de las características, de a una por vez, calculando la variación en la medida de error comparada con el resultado obtenido cuando todas las variables se encuentran sin permutar. De esta forma se puede realizar una estimación de la importancia de cada característica para el clasificador –mayor sea el deterioro en el desempeño al desordenar una característica, mayor será su importancia–. Con esto se obtiene un *ranking* de características, por lo que resta determinar cuáles de las características serán descartadas del conjunto «óptimo».

### 3.1.4. Algoritmos propuestos

Hasta ahora hemos abordado los procesos de extracción y selección de las características. La etapa final a describir es el entrenamiento y testeo de los clasificadores que utilizan el subconjunto óptimo de características. Sin embargo, antes resta resolver otro problema: el desbalance de clases.

Como los registros pertenecen a estudios de PSG para evaluar el sueño del paciente, es de esperarse que durante la mayor parte del tiempo el sujeto se encuentre dormido. Es decir, existe una mayor cantidad de segmentos correspondientes a la clase dormido que a la clase despierto. Cuando esto sucede, se corre el riesgo de que el clasificador tenga un sesgo hacia la clase mayoritaria. Existen diferentes estrategias para prevenir esta situación. Entre las más conocidas podemos mencionar el remuestreo de la base de datos o la generación de muestras sintéticas de la clase minoritaria mediante combinaciones convexas.

En este trabajo utilizamos dos métodos diferentes para abordar el desbalance de clases. El primero de éstos es el método de submuestreo de la clase mayoritaria, en el cual simplemente se eliminan muestras de la clase mayoritaria en forma aleatoria hasta que las clases están balanceadas. Este método se aplicó tanto para FFS-SVM como para RF y su selección de características. El segundo de los métodos modifica la función de costo asociada al clasificador SVM, imponiendo un costo adicional al modelo de forma tal que los errores cometidos durante el entrenamiento en datos correspondientes a la clase minoritaria tengan mayor peso [101]. De esta forma podemos sesgar el clasificador para que «preste mayor atención» a la clase minoritaria. El coeficiente utilizado para penalizar los errores en la clasificación toma en cuenta la relación entre la cantidad de datos que pertenecen a las clases despierto y la cantidad de datos que pertenecen a dormido. Como es fácil de imaginar, este método únicamente fue utilizado en SVM, ya que se trata de una modificación sobre su función de costo.

Las estrategias para balancear los datos se aplicaron sólo durante el entrenamiento. Durante las etapas de evaluación, no se realizó ningún procesamiento. De esta manera, el clasificador será evaluado bajo condiciones muy similares a la aplicación real.

En conclusión, si tomamos en cuenta las rutinas de selección de características y clasificación en conjunto con las estrategias propuestas para resolver el desbalance de clases, tenemos tres algoritmos posibles que serán entrenados y testeados: (1) FFS-SVM con penalización en los errores de la clase minoritaria (FS 1); (2) FFS-SVM con balance artificial de las clases (FS 2) y; (3) RF y su método asociado de selección de características (FS 3).

## 3.2. Resultados

En esta sección presentaremos los resultados obtenidos para los tres métodos propuestos, detallando la salida de cada una de las etapas que componen los métodos.

### 3.2.1. Selección de hiperparámetros

Para varias de las características descritas en la sección previa es necesario establecer en forma experimental los hiperparámetros. Para ello, realizamos un conjunto de experimentos con el objetivo de encontrar la combinación óptima de hiperparámetros utilizando 500 pacientes correspondientes a la base de datos para diseñar el sistema.

Para explorar las diferentes combinaciones de hiperparámetros se realizó una grilla de búsqueda. En primer lugar, realizamos una búsqueda abarcando importantes variaciones de los hiperparámetros para tener una idea general del comportamiento de las características al variar los hiperparámetros. Luego, realizamos una búsqueda más fina en torno a las regiones

con mejor desempeño encontradas anteriormente para obtener los hiperparámetros óptimos. Para guiar esta búsqueda utilizamos la medida de AUC como medida objetiva del poder discriminatorio de las características [75].

Como hemos mencionado anteriormente, los experimentos fueron realizados considerando secuencias de longitud  $L$ , con  $L$  variando entre 30 y 300 en pasos de 30. Es decir, se consideraron 10 casos en total. Las características en las que se realizó la búsqueda de hiperparámetros son ApEn, SampEn, FuzEn y DispEn. En las primeras tres características, variamos los hiperparámetros de dimensión de inmersión  $m$  –desde 2 hasta 8 en pasos de 1–, umbral  $r$  –desde  $\exp(-7)$  to  $\exp(4)$  variando el exponente en pasos de 0.55– y retardo de inmersión  $\tau$  –desde 1 hasta 4 en pasos de 1–. Para el caso de FuzEn, además fue necesario variar el exponente  $q$  de la función núcleo –explorando los valores 2, 3 y 4–. Para el caso de DispEn, es necesario definir la cantidad de clases  $c$  consideradas –se exploraron valores entre 2 y 4 en pasos de 1–. Para esta característica fue necesario modificar los hiperparámetros de la dimensión de inmersión  $m$  –desde 2 hasta  $m_{\text{máx}}$  en pasos de 1, donde  $m_{\text{máx}} = \text{floor}(\log(L)/c)$ –. Por último, el núcleo Gaussiano utilizado para el clasificador SVM se seleccionó variando  $\gamma$  de acuerdo a la ecuación  $\gamma = 1/k^2$ , con  $k$  tomando valores entre 1 y 16 en pasos de 2.

Los hiperparámetros que maximizan el área bajo la curva son muy similares para las diferentes longitudes de ventana  $L$  consideradas. Sin embargo, el AUC siempre incrementa con  $L$ . Las consecuencias de esto serán abordadas en la discusión realizada al final del capítulo.

### 3.2.2. Selección de características

Los métodos FS 1, FS 2 y FS 3 se aplicaron sobre las diferentes bases de datos obtenidas variando  $L$ . Los dos métodos de FFS-SVM se aplicaron como ya fue explicado, obteniendo 70 subconjuntos de características anidados. Para el método basado en RF se obtuvo un *ranking* de características. En este caso, para establecer la cantidad de características a utilizar no fuimos añadiéndolas de a una por vez según su posición en el *ranking*, sino que se generaron varios subconjuntos considerando subgrupos de características con *rankings* similares.

Al ser muchas características de la misma familia, el conjunto original de características presentaba una redundancia muy alta. Los experimentos realizados demostraron que con tan sólo algunas pocas características –aproximadamente 10– es posible obtener un desempeño muy similar al mejor resultado obtenido. En la figura 3.2 pueden verse la exactitud, sensibilidad y especificidad para la base de datos considerando ventanas de longitud  $L = 150$  obtenida mediante los tres métodos de selección de características propuestos. Para evitar ser repetitivos y dado que los resultados son muy similares, no presentaremos las figuras para otras longitudes de  $L$ .

A partir de los resultados de estos experimentos, se seleccionaron las primeras 40 características. En las tablas 3.1, 3.2 y 3.3 se muestra una lista exhaustiva de características seleccionadas por cada método y para cada una de las bases de datos. Las características son generalmente muy similares en todos los métodos, aunque el orden en el que son seleccionadas es ligeramente diferente. Una de las características más importantes, seleccionada por todos los métodos, es el valor medio de la señal. Como se mencionó anteriormente, según los estudios de Penzel, el valor medio de HR decrece hacia las fases del sueño más profundas, incrementando luego para la fase REM [32]. Otra de las características más relevantes es la ApEn. Al ser una medida de regularidad, esta característica tiene la capacidad de diferenciar las fases correspondientes a despierto y dormido dado que, como hemos visto, la vigilia está asociada a una mayor irregularidad en la señal de HR [32]. Existen otras características en las que se hace más difícil determinar en forma directa un correlato fisiológico. Sin embargo,



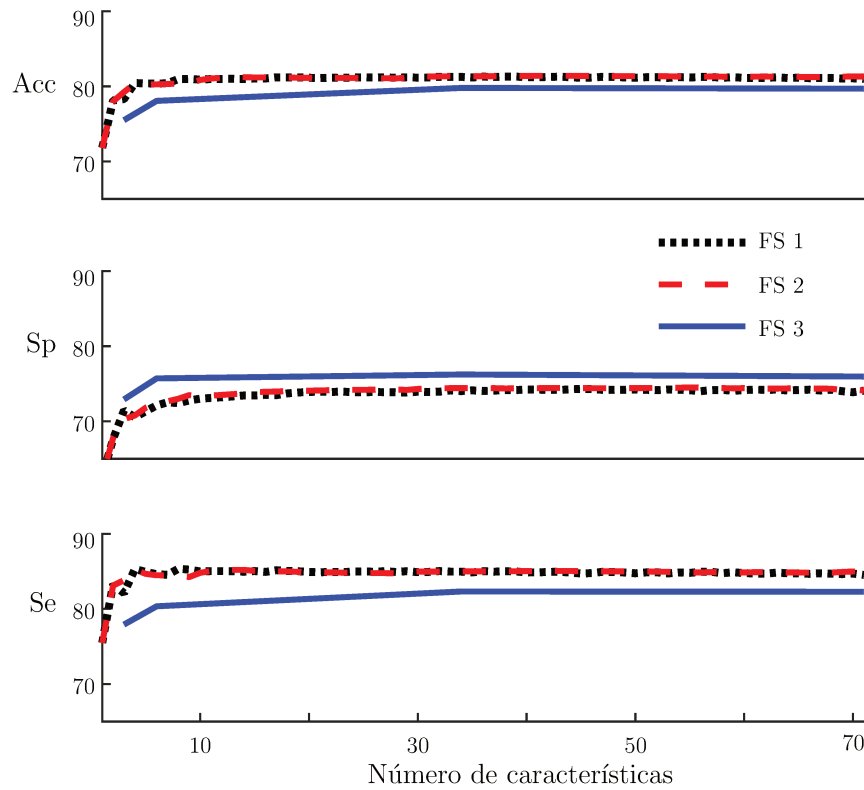


Figura 3.2: Variación del desempeño con los métodos de selección de características. Se muestra la exactitud, Acc, sensibilidad Se y especificidad Sp en función del número de características para los métodos FFS-SVM con penalización en la clase minoritaria (negro), FFS-SVM con balance de clases artificial (rojo) y selección de características mediante RF con balance de clases artificial (azul).

es evidente que la señal cambia su dinámica y que, dados los resultados obtenidos que se verán a continuación, las características utilizadas en este trabajo fueron capaces de reflejar estos cambios. Aunque la interpretabilidad de los resultados es siempre un beneficio, no es una necesidad. El objetivo final en este tipo de aplicaciones es obtener el mejor resultado posible.

### Experimentos preliminares

Durante las etapas preliminares del desarrollo de este algoritmo realizamos algunos experimentos para cuantificar la cantidad de información que agregaba cada característica. Para ello, en primer lugar efectuamos una clasificación a partir de establecer un umbral para cada característica, el cual fue seleccionado en la curva ROC para maximizar la sensibilidad y especificidad. A partir de esto, se calcula un índice de similaridad  $S$ , construido sumando la cantidad de veces que dos características diferentes generaban la misma clasificación. Es decir, se toman las clasificaciones generadas al umbralar dos características diferentes para el mismo paciente. Luego, los segmentos para los cuales el valor de la característica sea mayor al umbral pertenecerán a una clase, mientras que los datos cuyo valor sea menor pertenecerán a la otra clase. Esto se realiza para dos características, y luego se comparan ambas clasificaciones entre sí, contando la cantidad de coincidencias entre ellas. Finalmente, se escala la sumatoria obtenida respecto al total de datos –segmentos– clasificados. De esta forma, este índice oscila entre 0 –todas las clasificaciones de los datos diferentes entre sí– y 1 –coincidencia perfecta de todas las clasificaciones de los datos–. Este proceso se realizó para todos los pares de características posibles y, cuando pares de características tenían una

$L$	FS 1
30	mean, ApEn, TSMRE <sub>5</sub> , LF, LF/HF, $\max(\text{diag}(\mathbf{S}))$ , $r_{\max}$ , AR coefficient 4, TSMRE <sub>26</sub> , TSMRE <sub>8</sub> , $\text{mean}_{(t,s)}$ , HF, LZ of AC, FuzEn, std, TP, TSMRE <sub>7</sub> , SBE <sub>2</sub> , $\text{var}(\text{diag}(\mathbf{S}))$ , VLF, DispEn, $\text{RE}_{(t,s)}$ , SBE <sub>6</sub> , TSMRE <sub>27</sub> , TSMRE <sub>32</sub> , SBE <sub>1</sub> , TSMRE <sub>29</sub> , TSMRE <sub>31</sub> , TSMRE <sub>1</sub> , TSMRE <sub>25</sub> , first AC mín, $E_{\text{SVD}}$ , LZ, TSMRE <sub>30</sub> , AR coefficient 1, TSMRE <sub>6</sub> , TSMRE <sub>23</sub> , TSMRE <sub>13</sub> , TSMRE <sub>15</sub> , TSMRE <sub>14</sub> .
60	mean, ApEn, ApEn <sub>máx</sub> , first $ZC$ , normalize HF, std, TSMRE <sub>5</sub> , TSMRE <sub>17</sub> , $r_{\max}$ , SBE <sub>1</sub> , SBE <sub>8</sub> , TSMRE <sub>32</sub> , TSMRE <sub>24</sub> , $E_{\text{SVD}}$ , SBE <sub>6</sub> , TSMRE <sub>16</sub> , $\text{std}_{(t,s)}$ , TSMRE <sub>25</sub> , $\text{var}(\text{diag}(\mathbf{S}))$ , FuzEn, LF/HF, $\text{mean}_{(t,s)}$ , TSMRE <sub>18</sub> , SBE <sub>3</sub> , VLF, TP, LZ, TSMRE <sub>19</sub> , SampEn, HF, TSMRE <sub>14</sub> , TSMRE <sub>23</sub> , TSMRE <sub>21</sub> , LZ of AC, SBE <sub>7</sub> , TSMRE <sub>12</sub> , TSMRE <sub>22</sub> , TSMRE <sub>15</sub> , TSMRE <sub>13</sub> , TSMRE <sub>27</sub> .
90	mean, ApEn, ApEn <sub>máx</sub> , first $ZC$ , std, TSMRE <sub>3</sub> , TSMRE <sub>1</sub> , SBE <sub>8</sub> , $E_{\text{SVD}}$ , SBE <sub>4</sub> , TSMRE <sub>27</sub> , SampEn, TSMRE <sub>4</sub> , $r_{\max}$ , normalize LF, LF/HF, SBE <sub>5</sub> , LF, SBE <sub>6</sub> , TSMRE <sub>2</sub> , TSMRE <sub>27</sub> , AR coefficient 3, VLF, TP, LZ of AC, TSMRE <sub>13</sub> , FuzEn, normalize HF, TSMRE <sub>14</sub> , HF, $\text{mean}_{(t,s)}$ , TSMRE <sub>9</sub> , TSMRE <sub>12</sub> , SBE <sub>3</sub> , SBE <sub>2</sub> , TSMRE <sub>5</sub> , SBE <sub>1</sub> , SBE <sub>7</sub> , TSMRE <sub>19</sub> , TSMRE <sub>6</sub> .
120	ApEn, mean, first $ZC$ , ApEn <sub>máx</sub> , std, TSMRE <sub>1</sub> , SBE <sub>5</sub> , TSMRE <sub>2</sub> , SampEn, $E_{\text{SVD}}$ , ECM, SBE <sub>4</sub> , TSMRE <sub>16</sub> , SBE <sub>1</sub> , LZ of AC, TSMRE <sub>15</sub> , first AC mín, LF, TSMRE <sub>20</sub> , TSMRE <sub>21</sub> , HF, $\text{std}_{(t,s)}$ , TSMRE <sub>22</sub> , TP, VLF, TSMRE <sub>24</sub> , TSMRE <sub>17</sub> , FuzEn, TSMRE <sub>18</sub> , TSMRE <sub>19</sub> , TSMRE <sub>27</sub> , $\text{mean}_{(t,s)}$ , TSMRE <sub>26</sub> , TSMRE <sub>23</sub> , TSMRE <sub>32</sub> , SBE <sub>6</sub> , AR coefficient 4, TSMRE <sub>14</sub> , SBE <sub>3</sub> , $\max(\text{diag}(\mathbf{S}))$ .
150	TSMRE <sub>3</sub> , mean, TSMRE <sub>5</sub> , ApEn <sub>máx</sub> , ApEn, $E_{\text{SVD}}$ , $\text{var}(\text{diag}(\mathbf{S}))$ , first $ZC$ , SBE <sub>6</sub> , TSMRE <sub>1</sub> , TSMRE <sub>2</sub> , ECM, SBE <sub>2</sub> , HF, SBE <sub>8</sub> , $r_{\max}$ , TSMRE <sub>9</sub> , SBE <sub>4</sub> , LZ, TSMRE <sub>7</sub> , LZ of AC, TSMRE <sub>12</sub> , FuzEn, TSMRE <sub>22</sub> , TSMRE <sub>8</sub> , SampEn, TSMRE <sub>6</sub> , SBE <sub>3</sub> , normalize LF, std, TSMRE <sub>14</sub> , TSMRE <sub>17</sub> , TSMRE <sub>15</sub> , TSMRE <sub>18</sub> , VLF, TSMRE <sub>5</sub> , TSMRE <sub>10</sub> , normalize HF, $\max(\text{diag}(\mathbf{S}))$ , TSMRE <sub>31</sub> .
180	TSMRE <sub>3</sub> , mean, first $ZC$ , SBE <sub>1</sub> , LF, SBE <sub>4</sub> , ApEn <sub>máx</sub> , TSMRE <sub>1</sub> , $E_{\text{SVD}}$ , ECM, ApEn, TSMRE <sub>13</sub> , SBE <sub>8</sub> , TSMRE <sub>15</sub> , std, SBE <sub>2</sub> , FuzEn, SBE <sub>7</sub> , SBE <sub>5</sub> , VLF, TSMRE <sub>2</sub> , LZ of AC, TP, $\text{std}_{(t,s)}$ , TSMRE <sub>18</sub> , SBE <sub>6</sub> , SBE <sub>3</sub> , TSMRE <sub>29</sub> , HF, normalize HF, $\text{mean}_{(t,s)}$ , $\max(\text{diag}(\mathbf{S}))$ , $\text{var}(\text{diag}(\mathbf{S}))$ , AR coefficient 2, TSMRE <sub>31</sub> , TSMRE <sub>27</sub> , normalize LF, LZ, TSMRE <sub>14</sub> , TSMRE <sub>24</sub> .
210	TSMRE <sub>3</sub> , mean, LZ of AC, ApEn <sub>máx</sub> , SBE <sub>5</sub> , TSMRE <sub>1</sub> , std, AR coefficient 4, TSMRE <sub>11</sub> , $E_{\text{SVD}}$ , ECM, FuzEn, LF/HF, SBE <sub>4</sub> , TSMRE <sub>13</sub> , LZ, first $ZC$ , TSMRE <sub>2</sub> , SBE <sub>3</sub> , TSMRE <sub>12</sub> , SBE <sub>7</sub> , SBE <sub>2</sub> , TSMRE <sub>15</sub> , $\text{var}(\text{diag}(\mathbf{S}))$ , TSMRE <sub>8</sub> , SampEn, first AC mín, SBE <sub>6</sub> , $\max(\text{diag}(\mathbf{S}))$ , SBE <sub>8</sub> , TSMRE <sub>14</sub> , TSMRE <sub>5</sub> , TSMRE <sub>10</sub> , $r_{\max}$ , TSMRE <sub>19</sub> , TSMRE <sub>6</sub> , TSMRE <sub>16</sub> , $\text{RE}_{(t,s)}$ , LF, $\text{std}_{(t,s)}$ .
240	TSMRE <sub>3</sub> , mean, first $ZC$ , SBE <sub>8</sub> , ApEn <sub>máx</sub> , TSMRE <sub>1</sub> , $E_{\text{SVD}}$ , TSMRE <sub>17</sub> , LF/HF, std, TSMRE <sub>31</sub> , TSMRE <sub>2</sub> , AR coefficient 4, TSMRE <sub>28</sub> , SBE <sub>3</sub> , LZ of AC, TSMRE <sub>13</sub> , $\text{RE}_{(t,s)}$ , SBE <sub>6</sub> , TSMRE <sub>15</sub> , SBE <sub>2</sub> , $\text{std}_{(t,s)}$ , SampEn, SBE <sub>7</sub> , VLF, TSMRE <sub>30</sub> , ECM, normalize LF, ApEn, $\text{mean}_{(t,s)}$ , TP, TSMRE <sub>4</sub> , SBE <sub>4</sub> , SBE <sub>5</sub> , HF, FuzEn, SBE <sub>1</sub> , TSMRE <sub>18</sub> , TSMRE <sub>16</sub> , normalize HF.
270	TSMRE <sub>3</sub> , mean, LZ of AC, std, SBE <sub>5</sub> , ApEn <sub>máx</sub> , first $ZC$ , TSMRE <sub>1</sub> , TSMRE <sub>10</sub> , LZ, AR coefficient 2, $E_{\text{SVD}}$ , ECM, AR coefficient 1, SBE <sub>4</sub> , AR coefficient 4, TSMRE <sub>4</sub> , TSMRE <sub>2</sub> , SBE <sub>2</sub> , SBE <sub>8</sub> , TSMRE <sub>32</sub> , TSMRE <sub>30</sub> , SBE <sub>7</sub> , $\text{var}(\text{diag}(\mathbf{S}))$ , LF/HF, SBE <sub>6</sub> , SBE <sub>3</sub> , TSMRE <sub>22</sub> , VLF, DispEn, TSMRE <sub>27</sub> , TSMRE <sub>25</sub> , TSMRE <sub>31</sub> , SBE <sub>1</sub> , first AC mín, TP, TSMRE <sub>19</sub> , AR coefficient 3, TSMRE <sub>29</sub> , TSMRE <sub>20</sub> .
300	TSMRE <sub>3</sub> , mean, LZ of AC, ApEn <sub>máx</sub> , TSMRE <sub>1</sub> , SBE <sub>5</sub> , AR coefficient 4, std, TSMRE <sub>2</sub> , LF/HF, $E_{\text{SVD}}$ , ECM, TSMRE <sub>10</sub> , DispEn, normalize LF, SampEn, TSMRE <sub>16</sub> , TSMRE <sub>12</sub> , TSMRE <sub>30</sub> , HF, AR coefficient 2, TSMRE <sub>32</sub> , SBE <sub>6</sub> , TSMRE <sub>25</sub> , TSMRE <sub>15</sub> , AR coefficient 3, $\text{mean}_{(t,s)}$ , TP, TSMRE <sub>4</sub> , TSMRE <sub>8</sub> , $\text{std}_{(t,s)}$ , TSMRE <sub>9</sub> , AR coefficient 1, SBE <sub>3</sub> , VLF, TSMRE <sub>18</sub> , TSMRE <sub>11</sub> , $\text{var}(\text{diag}(\mathbf{S}))$ , LF, SBE <sub>2</sub> .

Tabla 3.1: Características seleccionadas en orden. Se muestran las 40 características seleccionadas utilizando FS 1.

$L$	FS 2
30	mean, ApEn, first $ZC$ , LF, TSMRE <sub>10</sub> , AR coefficient 4, TSMRE <sub>30</sub> , $r_{\text{máx}}$ , normalize LF, SBE <sub>7</sub> , SBE <sub>1</sub> , TSMRE <sub>4</sub> , $\text{std}_{(t,s)}$ , AR coefficient 1, $\text{std}$ , SBE <sub>2</sub> , $\text{RE}_{(t,s)}$ , TSMRE <sub>29</sub> , TSMRE <sub>8</sub> , VLF, TSMRE <sub>3</sub> , LZ, $\text{máx}(\text{diag}(\mathbf{S}))$ , TP, TSMRE <sub>27</sub> , AR coefficient 3, SBE <sub>6</sub> , DispEn, TSMRE <sub>23</sub> , TSMRE <sub>2</sub> , TSMRE <sub>21</sub> , TSMRE <sub>12</sub> , SampEn, FuzEn, TSMRE <sub>16</sub> , TSMRE <sub>17</sub> , $\text{var}(\text{diag}(\mathbf{S}))$ , first AC mín, ApEn <sub>máx</sub> , ESVD.
60	mean, ApEn, ApEn <sub>máx</sub> , TSMRE <sub>3</sub> , SBE <sub>5</sub> , $\text{std}$ , $r_{\text{máx}}$ , first $ZC$ , normalize HF, TSMRE <sub>8</sub> , TSMRE <sub>15</sub> , TSMRE <sub>18</sub> , SBE <sub>4</sub> , TSMRE <sub>23</sub> , DispEn, TSMRE <sub>20</sub> , AR coefficient 1, $\text{mean}_{(t,s)}$ , TSMRE <sub>28</sub> , $\text{máx}(\text{diag}(\mathbf{S}))$ , TSMRE <sub>7</sub> , VLF, SBE <sub>2</sub> , first AC mín, TSMRE <sub>11</sub> , SampEn, TSMRE <sub>12</sub> , TSMRE <sub>14</sub> , $\text{RE}_{(t,s)}$ , TSMRE <sub>31</sub> , LF/HF, SBE <sub>6</sub> , TSMRE <sub>9</sub> , TSMRE <sub>32</sub> , LZ of AC, TSMRE <sub>16</sub> , ESVD, TSMRE <sub>1</sub> , FuzEn, LF.
90	mean, ApEn, ApEn <sub>máx</sub> , first $ZC$ , TSMRE <sub>6</sub> , $\text{std}$ , SBE <sub>4</sub> , ESVD, TSMRE <sub>11</sub> , TSMRE <sub>3</sub> , SBE <sub>6</sub> , SBE <sub>2</sub> , DispEn, TSMRE <sub>20</sub> , LZ of AC, TSMRE <sub>2</sub> , $\text{var}(\text{diag}(\mathbf{S}))$ , AR coefficients 1, TSMRE <sub>5</sub> , SBE <sub>3</sub> , $\text{máx}(\text{diag}(\mathbf{S}))$ , $\text{mean}_{(t,s)}$ , TSMRE <sub>24</sub> , TSMRE <sub>15</sub> , TSMRE <sub>17</sub> , $\text{RE}_{(t,s)}$ , TSMRE <sub>16</sub> , LZ, TSMRE <sub>23</sub> , TSMRE <sub>13</sub> , TSMRE <sub>30</sub> , TSMRE <sub>19</sub> , normalize HF, first AC mín, AR coefficients 4, $r_{\text{máx}}$ , AR coefficients 2, ECM, TSMRE <sub>8</sub> , TSMRE <sub>14</sub> .
120	TSMRE <sub>3</sub> , mean, first $ZC$ , SBE <sub>1</sub> , normalize LF, SBE <sub>7</sub> , TSMRE <sub>18</sub> , ApEn <sub>máx</sub> , ESVD, TSMRE <sub>28</sub> , $\text{std}$ , TSMRE <sub>1</sub> , FuzEn, SBE <sub>5</sub> , TSMRE <sub>2</sub> , first AC mín, $r_{\text{máx}}$ , TSMRE <sub>27</sub> , SBE <sub>6</sub> , $\text{mean}_{(t,s)}$ , SBE <sub>4</sub> , TSMRE <sub>26</sub> , TSMRE <sub>21</sub> , SBE <sub>8</sub> , TSMRE <sub>31</sub> , TSMRE <sub>29</sub> , TSMRE <sub>22</sub> , LF, LZ, $\text{std}_{(t,s)}$ , $\text{var}(\text{diag}(\mathbf{S}))$ , TSMRE <sub>25</sub> , TSMRE <sub>15</sub> , TSMRE <sub>8</sub> , LZ of AC, TSMRE <sub>20</sub> , ECM, AR coefficients 2, AR coefficients 1, TSMRE <sub>24</sub> .
150	TSMRE <sub>3</sub> , mean, first $ZC$ , ApEn <sub>máx</sub> , SampEn, SBE <sub>5</sub> , ESVD, TSMRE <sub>1</sub> , TSMRE <sub>29</sub> , SBE <sub>6</sub> , SBE <sub>1</sub> , normalize HF, HF, LZ, LZ of AC, $\text{std}$ , TSMRE <sub>2</sub> , ApEn, TSMRE <sub>16</sub> , first AC mín, LF, $\text{var}(\text{diag}(\mathbf{S}))$ , TSMRE <sub>13</sub> , TSMRE <sub>20</sub> , TSMRE <sub>31</sub> , TSMRE <sub>17</sub> , LF/HF, TSMRE <sub>24</sub> , SBE <sub>4</sub> , ECM, SBE <sub>2</sub> , normalize LF, FuzEn, TSMRE <sub>6</sub> , SBE <sub>3</sub> , $\text{RE}_{(t,s)}$ , TSMRE <sub>15</sub> , $r_{\text{máx}}$ , SBE <sub>8</sub> , TSMRE <sub>17</sub> .
180	TSMRE <sub>3</sub> , mean, first $ZC$ , SBE <sub>1</sub> , VLF, SBE <sub>5</sub> , ApEn <sub>máx</sub> , FuzEn, SBE <sub>4</sub> , ESVD, ECM, SBE <sub>2</sub> , TSMRE <sub>1</sub> , $\text{std}$ , TSMRE <sub>29</sub> , TSMRE <sub>11</sub> , normalize HF, DispEn, SBE <sub>3</sub> , LZ of AC, TSMRE <sub>30</sub> , TSMRE <sub>19</sub> , SBE <sub>8</sub> , TSMRE <sub>17</sub> , SampEn, TSMRE <sub>2</sub> , SBE <sub>6</sub> , first AC mín, LF, SBE <sub>7</sub> , $\text{std}_{(t,s)}$ , $\text{var}(\text{diag}(\mathbf{S}))$ , LZ, AR coefficients 1, TSMRE <sub>13</sub> , TSMRE <sub>24</sub> , $\text{máx}(\text{diag}(\mathbf{S}))$ , TSMRE <sub>10</sub> , TSMRE <sub>23</sub> , HF.
210	TSMRE <sub>3</sub> , mean, LZ of AC, ApEn <sub>máx</sub> , LF/HF, TSMRE <sub>1</sub> , SBE <sub>6</sub> , $\text{mean}_{(t,s)}$ , ECM, AR coefficients 4, FuzEn, TSMRE <sub>12</sub> , TSMRE <sub>23</sub> , SBE <sub>7</sub> , ESVD, TSMRE <sub>2</sub> , TSMRE <sub>28</sub> , $\text{std}$ , TSMRE <sub>17</sub> , TSMRE <sub>17</sub> , SampEn, TSMRE <sub>18</sub> , $r_{\text{máx}}$ , SBE <sub>8</sub> , LF, VLF, TSMRE <sub>27</sub> , SBE <sub>5</sub> , TSMRE <sub>4</sub> , TSMRE <sub>30</sub> , first $ZC$ , TSMRE <sub>8</sub> , TSMRE <sub>15</sub> , TP, SBE <sub>4</sub> , $\text{std}_{(t,s)}$ , LZ, SBE <sub>3</sub> , TSMRE <sub>16</sub> , TSMRE <sub>5</sub> , ApEn.
240	TSMRE <sub>3</sub> , mean, LZ of AC, SBE <sub>5</sub> , $\text{std}$ , ApEn <sub>máx</sub> , LF/HF, TSMRE <sub>1</sub> , ESVD, TSMRE <sub>26</sub> , TSMRE <sub>2</sub> , LF, TSMRE <sub>30</sub> , first $ZC$ , TSMRE <sub>13</sub> , TSMRE <sub>19</sub> , ApEn, SBE <sub>6</sub> , TSMRE <sub>31</sub> , SBE <sub>2</sub> , TSMRE <sub>24</sub> , TSMRE <sub>10</sub> , LZ, SBE <sub>8</sub> , TSMRE <sub>11</sub> , ECM, SampEn, AR coefficients 1, TSMRE <sub>15</sub> , TSMRE <sub>28</sub> , TSMRE <sub>29</sub> , $\text{var}(\text{diag}(\mathbf{S}))$ , FuzEn, AR coefficients 2, $\text{máx}(\text{diag}(\mathbf{S}))$ , SBE <sub>7</sub> , TP, TSMRE <sub>14</sub> , TSMRE <sub>20</sub> , VLF.
270	TSMRE <sub>3</sub> , mean, LZ of AC, SBE <sub>5</sub> , $\text{std}$ , ApEn <sub>máx</sub> , ESVD, TSMRE <sub>11</sub> , $\text{RE}_{(t,s)}$ , first $ZC$ , ECM, SBE <sub>4</sub> , TSMRE <sub>2</sub> , ApEn, TSMRE <sub>14</sub> , TSMRE <sub>9</sub> , LZ, AR coefficients 2, LF/HR, AR coefficients 1, SBE <sub>2</sub> , TSMRE <sub>20</sub> , TP, $\text{std}$ , SampEn, TSMRE <sub>24</sub> , TSMRE <sub>25</sub> , TSMRE <sub>17</sub> , TSMRE <sub>31</sub> , SBE <sub>3</sub> , SBE <sub>6</sub> , $\text{máx}(\text{diag}(\mathbf{S}))$ , TSMRE <sub>7</sub> , TSMRE <sub>16</sub> , SBE <sub>7</sub> , TSMRE <sub>12</sub> , TSMRE <sub>11</sub> , TSMRE <sub>13</sub> , AR coefficients 3, first AC mín.
300	TSMRE <sub>3</sub> , mean, LZ of AC, TSMRE <sub>1</sub> , ApEn <sub>máx</sub> , SBE <sub>5</sub> , ESVD, TSMRE <sub>2</sub> , LF/HF, ECM, FuzEn, AR coefficients 4, TSMRE <sub>14</sub> , LF, first $ZC$ , TSMRE <sub>16</sub> , SBE <sub>4</sub> , TSMRE <sub>26</sub> , SBE <sub>7</sub> , TSMRE <sub>29</sub> , TSMRE <sub>19</sub> , normalize LF, TSMRE <sub>27</sub> , SBE <sub>2</sub> , AR coefficients 2, TSMRE <sub>8</sub> , DispEn, SBE <sub>6</sub> , $\text{std}_{(t,s)}$ , AR coefficients 1, AR coefficients 3, TSMRE <sub>13</sub> , LZ, $\text{var}(\text{diag}(\mathbf{S}))$ , TSMRE <sub>32</sub> , VLF, SBE <sub>8</sub> , TSMRE <sub>6</sub> , SBE <sub>1</sub> , ApEn.

Tabla 3.2: Características seleccionadas en orden. Se muestran las 40 características seleccionadas utilizando FS 2.

$L$	FS 3
30	mean, HF, LF, ApEn <sub>máx</sub> , TP, std, mean <sub>(t,s)</sub> , VLF, TSMRE <sub>6</sub> , TSMRE <sub>5</sub> , TSMRE <sub>7</sub> , TSMRE <sub>8</sub> , TSMRE <sub>4</sub> , TSMRE <sub>9</sub> , TSMRE <sub>10</sub> , TSMRE <sub>3</sub> , TSMRE <sub>2</sub> , ApEn, ECM, normalize LF, E <sub>SVD</sub> , var(diag( <b>S</b> )), SBE <sub>1</sub> , TSMRE <sub>1</sub> , LF/HF, max <b>S</b> , SBE <sub>4</sub> , FuzEn, TSMRE <sub>12</sub> , normalize HF, SBE <sub>5</sub> , SBE <sub>8</sub> , TSMRE <sub>27</sub> , TSMRE <sub>11</sub> , SampEn, AR coefficients 1, TSMRE <sub>26</sub> , TSMRE <sub>28</sub> , SBE <sub>7</sub> , TSMRE <sub>25</sub> .
60	mean, ApEn <sub>máx</sub> , TSMRE <sub>1</sub> , HF, E <sub>SVD</sub> , ECM, AR coefficients 1, TSMRE <sub>2</sub> , TSMRE <sub>19</sub> , TSMRE <sub>16</sub> , mean <sub>(t,s)</sub> , TSMRE <sub>18</sub> , SBE <sub>1</sub> , TSMRE <sub>17</sub> , LF, RE <sub>(t,s)</sub> , var(diag( <b>S</b> )), máx(diag( <b>S</b> )), TP, TSMRE <sub>20</sub> , normalize LF, TSMRE <sub>15</sub> , VLF, SBE <sub>5</sub> , std, TSMRE <sub>14</sub> , TSMRE <sub>21</sub> , normalize HF, std <sub>(t,s)</sub> , LF/HF, TSMRE <sub>22</sub> , TSMRE <sub>25</sub> , TSMRE <sub>4</sub> , TSMRE <sub>24</sub> , SBE <sub>2</sub> , TSMRE <sub>23</sub> , SampEn, $r_{máx}$ , SBE <sub>8</sub> , TSMRE <sub>13</sub> .
90	mean, ApEn <sub>máx</sub> , TSMRE <sub>2</sub> , TSMRE <sub>3</sub> , AR coefficients 1, ECM, first ZC, E <sub>SVD</sub> , HF, SBE <sub>1</sub> , normalize LF, LF/HF, SBE <sub>5</sub> , normalize HF, mean <sub>(t,s)</sub> , TSMRE <sub>26</sub> , TSMRE <sub>30</sub> , TSMRE <sub>25</sub> , TSMRE <sub>27</sub> , TSMRE <sub>28</sub> , TSMRE <sub>29</sub> , TSMRE <sub>31</sub> , VLF, TSMRE <sub>23</sub> , LF, TSMRE <sub>24</sub> , TSMRE <sub>32</sub> , TSMRE <sub>21</sub> , std <sub>(t,s)</sub> , TSMRE <sub>22</sub> , TSMRE <sub>20</sub> , TP, máx(diag( <b>S</b> )), TSMRE <sub>4</sub> , first AC mín, std, AR coefficients 2, AR coefficients 4, TSMRE <sub>3</sub> , RE <sub>(t,s)</sub> , DispEn.
120	mean, ApEn <sub>máx</sub> , TSMRE <sub>1</sub> , TSMRE <sub>2</sub> , first ZC, AR coefficients 1, ECM, SBE <sub>1</sub> , E <sub>SVD</sub> , SBE <sub>5</sub> , HF, first AC mín, normalize LF, AR coefficients 4, mean <sub>(t,s)</sub> , TSMRE <sub>32</sub> , normalize HF, LF/HF, AR coefficients 2, TSMRE <sub>30</sub> , TSMRE <sub>3</sub> , SBE <sub>8</sub> , TSMRE <sub>31</sub> , VLF, TSMRE <sub>4</sub> , DispEn, LF, TSMRE <sub>22</sub> , TSMRE <sub>5</sub> , SBE <sub>2</sub> , TSMRE <sub>28</sub> , TSMRE <sub>29</sub> , TP, TSMRE <sub>26</sub> , std, $r_{máx}$ , TSMRE <sub>20</sub> , ApEn, TSMRE <sub>24</sub> , TSMRE <sub>18</sub> .
150	mean, ApEn <sub>máx</sub> , TSMRE <sub>1</sub> , first ZC, TSMRE <sub>2</sub> , AR coefficients 1, AR coefficients 4, first AC mín, normalize LF, LF/HF, E <sub>SVD</sub> , SBE <sub>1</sub> , SBE <sub>5</sub> , AR coefficients 2, normalize HF, ECM, TSMRE <sub>3</sub> , HF, LF, mean <sub>(t,s)</sub> , TSMRE <sub>5</sub> , TSMRE <sub>4</sub> , SBE <sub>8</sub> , TSMRE <sub>32</sub> , SBE <sub>2</sub> , TSMRE <sub>31</sub> , $r_{máx}$ , VLF, TSMRE <sub>28</sub> , TSMRE <sub>29</sub> , TSMRE <sub>30</sub> , TSMRE <sub>26</sub> , SampEn, DispEn, TSMRE <sub>22</sub> , LZ of AC, TSMRE <sub>24</sub> , TSMRE <sub>20</sub> , TP, TSMRE <sub>19</sub> .
180	mean, ApEn <sub>máx</sub> , TSMRE <sub>1</sub> , first ZC, TSMRE <sub>2</sub> , AR coefficients 1, AR coefficients 4, first AC mín, SBE <sub>5</sub> , AR coefficients 2, normalize LF, E <sub>SVD</sub> , SBE <sub>1</sub> , ECM, normalize HF, LF/HF, TSMRE <sub>3</sub> , HF, SBE <sub>8</sub> , mean <sub>(t,s)</sub> , TSMRE <sub>4</sub> , VLF, LF, DispEn, TSMRE <sub>30</sub> , TSMRE <sub>32</sub> , SBE <sub>2</sub> , TSMRE <sub>5</sub> , TSMRE <sub>31</sub> , $r_{máx}$ , SBE <sub>4</sub> , ApEn, TSMRE <sub>28</sub> , TSMRE <sub>27</sub> , TSMRE <sub>29</sub> , LZ of AC, TSMRE <sub>7</sub> , TSMRE <sub>6</sub> , TSMRE <sub>25</sub> , SampEn, TSMRE <sub>26</sub> .
210	mean, ApEn <sub>máx</sub> , TSMRE <sub>1</sub> , first ZC, TSMRE <sub>2</sub> , AR coefficients 1, AR coefficients 4, first AC mín, AR coefficients 2, SBE <sub>5</sub> , normalize LF, SBE <sub>1</sub> , LF/HF, normalize HF, E <sub>SVD</sub> , LZ of AC, ECM, TSMRE <sub>3</sub> , SBE <sub>8</sub> , HF, TSMRE <sub>32</sub> , TSMRE <sub>4</sub> , TSMRE <sub>5</sub> , LF, SBE <sub>4</sub> , SampEn, DispEn, mean <sub>(t,s)</sub> , VLF, LZ, TSMRE <sub>6</sub> , TSMRE <sub>7</sub> , SBE <sub>6</sub> , TSMRE <sub>30</sub> , TSMRE <sub>31</sub> , SBE <sub>2</sub> , $r_{máx}$ , ApEn, SBE <sub>7</sub> , TSMRE <sub>29</sub> .
240	mean, ApEn <sub>máx</sub> , TSMRE <sub>1</sub> , first ZC, AR coefficients 4, first AC mín, TSMRE <sub>2</sub> , AR coefficients 1, AR coefficients 2, SBE <sub>5</sub> , normalize LF, E <sub>SVD</sub> , SBE <sub>1</sub> , LF/HF, HF, normalize HF, ECM, TSMRE <sub>3</sub> , LZ of AC, DispEn, SBE <sub>8</sub> , TSMRE <sub>32</sub> , mean <sub>(t,s)</sub> , TSMRE <sub>4</sub> , VLF, SBE <sub>2</sub> , TSMRE <sub>31</sub> , LF, ApEn, TSMRE <sub>5</sub> , SBE <sub>4</sub> , $r_{máx}$ , LZ, TSMRE <sub>29</sub> , SBE <sub>6</sub> , TSMRE <sub>30</sub> , TSMRE <sub>7</sub> , SampEn, TSMRE <sub>6</sub> , TSMRE <sub>19</sub> .
270	mean, ApEn <sub>máx</sub> , TSMRE <sub>1</sub> , first ZC, AR coefficients 4, AR coefficients 1, TSMRE <sub>2</sub> , first AC mín, AR coefficients 2, SBE <sub>5</sub> , LZ of AC, normalize LF, LF/HF, normalize HF, SBE <sub>8</sub> , SBE <sub>1</sub> , E <sub>SVD</sub> , TSMRE <sub>3</sub> , HF, DispEn, ECM, TSMRE <sub>4</sub> , SBE <sub>4</sub> , TSMRE <sub>32</sub> , TSMRE <sub>5</sub> , LZ, SBE <sub>6</sub> , mean <sub>(t,s)</sub> , SBE <sub>2</sub> , TSMRE <sub>31</sub> , LF, TSMRE <sub>7</sub> , ApEn, VLF, SampEn, TSMRE <sub>30</sub> , $r_{máx}$ , TSMRE <sub>6</sub> , SBE <sub>7</sub> , TSMRE <sub>8</sub> .
300	mean, TSMRE <sub>1</sub> , ApEn <sub>máx</sub> , first ZC, AR coefficients 4, AR coefficients 1, first AC mín, AR coefficients 2, TSMRE <sub>2</sub> , SBE <sub>5</sub> , LZ of AC, normalize LF, LF/HF, SBE <sub>1</sub> , ECM, SBE <sub>8</sub> , normalize HF, E <sub>SVD</sub> , TSMRE <sub>3</sub> , DispEn, SBE <sub>4</sub> , HF, mean <sub>(t,s)</sub> , TSMRE <sub>5</sub> , $r_{máx}$ , TSMRE <sub>7</sub> , TSMRE <sub>32</sub> , SBE <sub>2</sub> , TSMRE <sub>4</sub> , SBE <sub>6</sub> , SampEn, VLF, SBE <sub>7</sub> , ApEn, LF, TSMRE <sub>30</sub> , TSMRE <sub>31</sub> , LZ, TSMRE <sub>9</sub> , SBE <sub>3</sub> .

Tabla 3.3: Características seleccionadas en orden. Se muestran las 40 características seleccionadas utilizando FS 3.

		FS 1		FS 2		FS 3		$L$
Clasificación	W	54.6%	19.1%	55.0%	19.4%	62.8%	27.3%	30
	S	45.4%	80.9%	45.0%	80.6%	37.2%	72.7%	
	W	63.8%	14.0%	64.7%	14.6%	69.8%	19.8%	60
	S	36.2%	86.0%	35.3%	85.4%	30.2%	80.2%	
	W	67.8%	14.0%	68.2%	14.3%	72.3%	18.6%	90
	S	32.2%	86.0%	31.8%	85.7%	27.7%	81.4%	
	W	70.4%	13.6%	70.9%	13.9%	74.4%	17.8%	120
	S	29.6%	86.4%	29.1%	86.1%	25.6%	82.2%	
	W	72.8%	13.3%	73.3%	13.8%	76.0%	16.2%	150
	S	27.2%	86.7%	26.7%	86.2%	24.0%	83.8%	
	W	74.4%	12.8%	74.7%	13.3%	77.1%	16.2%	180
	S	25.6%	87.2%	25.3%	86.7%	22.9%	83.8%	
	W	75.9%	12.7%	76.1%	13.0%	78.3%	15.7%	210
	S	24.1%	87.3%	23.9%	87.0%	21.7%	84.3%	
	W	77.2%	12.3%	77.3%	12.6%	79.1%	15.1%	240
	S	22.8%	87.7%	22.7%	87.4%	20.9%	84.9%	
	W	78.3%	12.1%	78.3%	12.2%	79.9%	14.7%	270
	S	21.7%	87.9%	21.7%	87.8%	20.1%	85.3%	
W	79.0%	11.7%	79.2%	11.9%	80.3%	14.7%	300	
S	21.0%	88.3%	20.8%	88.1%	19.7%	85.9%		
		W	S	W	S	W	S	
		Referencia						

Tabla 3.4: Matriz de confusión con los resultados obtenidos en los 4500 pacientes correspondientes a la etapa de evaluación. Se presentan los resultados para las diferentes bases de datos obtenidas al variar  $L$ . Los desempeños obtenidos pueden verse en tonos de gris. FS 1: FFS-SVM con penalización de error en clase minoritaria. FS 2: FFS-SVM con balance artificial de clases. FS 3: selección de variables mediante RF. Las clases despierto y dormido se etiquetaron con  $W$  y  $S$ .

alta similaridad, se procedió a descartar aquella que tenía menor AUC, de forma de evitar la redundancia [103].

En la versión final del algoritmo desarrollado no fue necesario establecer esta comparativa, ya que se estableció una rutina de selección de características superadora en la que se involucraba a todo el sistema para establecer el conjunto óptimo. Sin embargo, el análisis realizado sigue siendo útil para analizar *a posteriori* el porqué de la selección de cada característica.

### 3.2.3. Desempeño de los algoritmos en la etapa de evaluación

Las tres variantes de los algoritmos desarrollados (FS1, FS2, y FS3) se evaluaron utilizando los 4500 sujetos pertenecientes a la base de datos generada para la «etapa de evaluación». Para ello, utilizamos el método de validación cruzada  $k$ -fold CV, seleccionando  $k = 10$ . Todas las medidas de desempeño se calcularon para cada uno de los sujetos evaluados en forma individual, reportando luego las medidas de error promedio obtenidas. De esta forma, logramos que las medidas reportadas sean más cercanas a las que se obtendrían en una aplicación real del algoritmo, donde el objetivo es estimar el TTS del sujeto.

	Acc	Sp	Se	PPV	NPV	$L$
FS 1	<b>73.7</b>	<b>80.9</b>	54.6	<b>48.6</b>	83.1	
FS 2	73.6	80.6	55.0	48.4	83.6	30
FS 3	69.7	72.7	<b>62.8</b>	43.1	<b>84.65</b>	
FS 1	<b>79.4</b>	<b>86.0</b>	63.8	<b>58.3</b>	86.5	
FS 2	79.2	85.4	64.7	57.6	86.7	60
FS 3	76.6	80.2	<b>69.8</b>	52.5	<b>87.6</b>	
FS 1	<b>80.6</b>	<b>86.0</b>	67.8	<b>59.3</b>	87.7	
FS 2	80.4	85.7	68.2	59.0	87.8	90
FS 3	78.2	81.4	<b>72.3</b>	54.4	<b>88.6</b>	
FS 1	<b>81.4</b>	<b>86.4</b>	70.4	<b>60.5</b>	88.5	
FS 2	<b>81.4</b>	86.1	70.9	60.3	88.7	120
FS 3	79.4	82.2	<b>74.4</b>	56.0	<b>89.4</b>	
FS 1	<b>82.3</b>	<b>86.7</b>	72.8	<b>61.8</b>	89.3	
FS 2	82.0	86.2	73.3	61.1	89.4	150
FS 3	80.3	83.0	<b>76.0</b>	57.3	<b>90.0</b>	
FS 1	<b>83.1</b>	<b>87.2</b>	74.4	<b>63.2</b>	89.8	
FS 2	82.8	86.7	74.7	62.4	89.9	180
FS 3	81.2	83.8	<b>77.1</b>	58.6	<b>90.4</b>	
FS 1	<b>83.6</b>	<b>87.3</b>	75.9	<b>64.0</b>	90.3	
FS 2	83.4	87.0	76.1	63.4	90.3	210
FS 3	81.9	84.3	<b>78.3</b>	59.8	<b>90.8</b>	
FS 1	<b>84.2</b>	<b>87.7</b>	77.2	<b>65.2</b>	90.7	
FS 2	84.0	87.4	77.3	64.6	90.7	240
FS 3	82.6	84.9	<b>79.1</b>	61.0	<b>91.1</b>	
FS 1	<b>84.7</b>	<b>87.9</b>	78.3	<b>66.0</b>	91.1	
FS 2	84.6	87.8	78.3	65.7	91.1	270
FS 3	83.1	85.3	<b>79.9</b>	62.0	<b>91.4</b>	
FS 1	<b>85.2</b>	<b>88.3</b>	79.0	<b>67.0</b>	91.3	
FS 2	85.0	88.1	79.2	66.7	91.4	300
FS 3	83.6	85.9	<b>80.3</b>	63.1	<b>91.5</b>	

Tabla 3.6: Desempeño de los algoritmos en los 4500 sujetos para los diferentes valores de  $L$  considerados. FS 1: FFS-SVM con penalización de error en clase minoritaria. FS 2: FFS-SVM con balance artificial de clases. FS 3: selección de variables mediante RF. Los mejores resultados se resaltan en negrita.

En la tabla 3.4 se muestra la matriz de confusión porcentual obtenida al aplicar los algoritmos a las bases de datos de evaluación. La cantidad de falsos positivos (FP), verdaderos positivos (VP), falsos negativos (FN) y verdaderos negativos (VN) se calculó para cada uno de los pacientes y luego se reportaron los valores promedio. Todos los algoritmos tuvieron

un mejor desempeño para la clase mayoritaria –es decir, cuando el paciente se encuentra dormido–. El desempeño, además, se incrementa con la longitud del segmento  $L$  considerado.

En la tabla 3.6 hemos resumido algunas de las medidas de desempeño más comunes: exactitud, sensibilidad, especificidad, valor predictivo positivo y valor predictivo negativo. Estas medidas se reportaron para cada uno de las 10 bases de datos obtenidas al variar  $L$ . Puede verse a partir de estos resultados que los algoritmos basados en SVM –FS 1 y FS 2– tienen una mayor sensibilidad, mientras que el algoritmo basado en RF –FS 3– presenta una mayor especificidad. Para calcular estas medidas, la clase correspondiente a despierto se consideró la clase positiva. La mejor exactitud alcanzada por FS 1 y FS 2 se explica por el hecho de que clasifican mejor la clase mayoritaria, la cual, en última instancia, es la que tiene mayor peso en el cálculo de esta medida.

### 3.3. Discusión

En esta sección discutiremos los alcances de este trabajo y de sus resultados, además de la inserción de este trabajo en el estado del arte.

#### 3.3.1. Ventajas y dificultades de la oximetría de pulso

Como hemos mencionado varias veces en capítulos anteriores, el uso de la oximetría de pulso como única señal para diagnosticar SAOS tiene fuertes ventajas para métodos de cribado por su bajo costo y la simpleza relativa de su instrumentación. Sin embargo, continúa siendo un método controversial y bajo continuo estudio. Existe una importante variabilidad de la señal adquirida al utilizar diferentes dispositivos y ante diferentes situaciones y pacientes –factores como la edad, el color de piel y porcentaje de grasa corporal son importantes causas de variación–. Por todo esto, la dispersión de los valores obtenida es muy alta, haciendo que la interpretación de las medidas obtenidas sea ambigua sin una experiencia previa con el dispositivo utilizado [104]. Adicionalmente, los oxímetros de pulso son muy sensibles a los artefactos de movimiento [53]. A pesar de esto se han obtenido resultados bastante significativos a partir de técnicas de procesamiento de señales para evitar este tipo de inconvenientes. En resumidas cuentas, las señales obtenidas mediante oxímetros de pulso poseen un conjunto de características que las hacen sumamente atractivas para el cribado, pero para que los algoritmos diseñados puedan cumplir con sus objetivos muchas veces es necesario incrementar su complejidad dada la baja calidad de la señal.

La clasificación de las fases del sueño se realiza principalmente mediante la señal de EEG, aunque también es usual extraer información del EOG, fundamentalmente para la diferenciación entre W y REM, y EMG, para detectar movimientos asociados a las diferentes fases del sueño. En cambio, detectar si un segmento de la señal de HR corresponde a despierto o a dormido es una tarea bastante complicada. Si bien existen cambios en la dinámica de la señal de HR [32], la información es mucho más difícil de interpretar. Si a esto le añadimos la baja exactitud de la estimación de la HR obtenida, la sensibilidad a los artefactos de movimiento y la baja frecuencia de muestreo –lo que dificulta la obtención adecuada de algunas características en la etapa de extracción– podemos empezar a tener una idea de la dificultad para alcanzar un desempeño aceptable.

#### 3.3.2. Longitud de segmento $L$

En este apartado abordaremos una de las discusiones que venimos anticipando desde comienzos de este capítulo: la longitud  $L$  utilizada. Recordemos que según la AASM, la

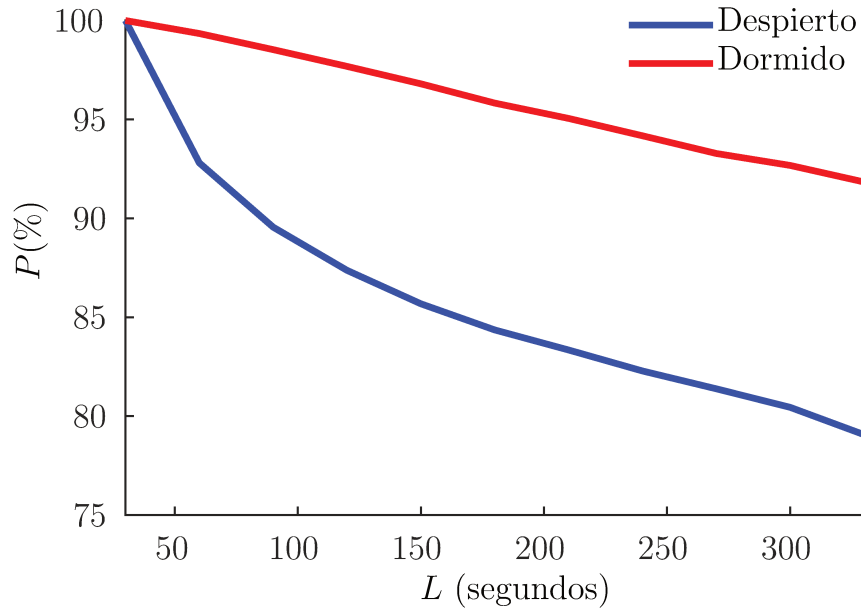


Figura 3.3: Porcentaje  $P(\%)$  del total de tiempo despierto/dormido considerando únicamente segmentos de longitud mayores que  $L$ .

longitud para calcular los hipnogramas es de 30 segundos. A partir de esto es sencillo imaginarse que longitudes de  $L$  menores harán más simple trasladar los resultados obtenidos a un hipnograma acorde a las convenciones vigentes.

Para explicar esto, supongamos que el hipnograma  $H$  es una secuencia de la forma  $H = \{t_{W_1}, t_{S_1}, t_{W_2}, t_{S_2}, \dots, t_{S_{N-1}}, t_{W_N}\}$ , donde  $t_{W_i}$  y  $t_{S_i}$  son las longitudes del  $i$ -ésimo segmento correspondiente a despierto y dormido. En la figura 3.3 se muestra el porcentaje  $P$  del TTS considerando únicamente los segmentos cuya longitud es mayor a  $L$ , esto es:

$$P(\%) = \frac{\sum_{i \in \mathcal{I}} t_{W_i}}{\sum_{\forall i} t_{W_i}} \cdot 100, \quad (3.18)$$

donde  $\mathcal{I}$  es el conjunto de segmentos tales que  $t_{W_i} > L$ . La gráfica obtenida al considerar segmentos de sueño es análoga.

Debemos recordar que en estos experimentos nosotros hemos decidido descartar los segmentos que contenían una mezcla de los estados de despierto y dormido –esta mezcla sólo es posible al incrementar  $L$  a valores mayores que 30 segundos–. Sin embargo, al aplicar el método en una situación real es necesario tomar la señal de HR y partirla en segmentos no solapados de longitud  $L$ , y luego clasificar dichos segmentos. En este caso, si existen segmentos correspondientes a despierto o dormido cuya longitud es menor que  $L$ , entonces algunos de los segmentos no pertenecerán a una única clase. Como es de esperarse, con el incremento de  $L$  también se incrementan los segmentos con clases «mezcladas». El sistema no está diseñado para lidiar con estas mezclas, ya que en las características extraídas se reflejará información con patrones de ambas clases. De la figura 3.3 puede deducirse cómo incrementa la magnitud de segmentos mezclados al incrementar  $L$ .

Por otro lado, de acuerdo a lo que hemos reportado en las tablas de resultados, a medida que incrementamos  $L$  el desempeño del algoritmo mejora, ya que para longitudes mayores hay más información disponible que permita detectar los cambios en la dinámica y, además, las características pueden ser calculadas de forma más exacta. Por dar un ejemplo, la resolución en las características en el dominio frecuencial se incrementa al utilizar segmentos más largos. Por este motivo, al modificar  $L$  existe una relación de compromiso entre el desempeño del algoritmo y su aplicabilidad en una situación real. Cuando seleccionamos longitudes mayores



Método	Señales	C	N	Acc	Se	Sp	PPV	NPV	$\kappa$
CWT feat.+RF [30]	EEG	5	16	93.4	95.9	89.6	93.3	93.5	0.86
CNN + RNN [28]	EEG	5	20	94.3	96.8	83.4	86.0	96.2	0.81
CNN + RNN [65]	EEG	5	61	95.7	96.9	90.6	87.8	97.7	0.86
Feat.+SVM+RFE [34]	ECG	2	18	80.0	84.5	69.1	87.0	64.5	0.52
CNN [36]	ECG (CGMH-val)	2	27	83.1	89.4	52.4	90.1	50.5	0.41
CNN [36]	ECG (DREAMS)	2	20	81.4	87.1	53.1	90.2	45.2	0.38
CNN [36]	ECG (UCDSADB)	2	25	73.7	81.9	43.4	84.3	39.2	0.24
Feat. + FFS-SVM [64]	HR	2	5000	73.7	80.9	54.6	48.6	84.7	0.44
Feat.+SVM [37]	HRV	2	10	73.1	74.0	72.0	–	–	0.46
Feat.+SVM [37]	PPG	2	10	76.8	76.0	77.0	–	–	0.53
Feat.+kNN [37]	PPG+HRV	2	10	79.4	77.0	81.0	–	–	0.59
CNN [36]	PPG (CGMH-val)	2	27	84.2	90.9	51.5	90.1	53.6	0.43
Feat.+RNN [66]	ECG+acc	5	778	90.1	94.0	72.9	–	75.1	0.65
Feat.+LDA [38]	PPG+acc	4	60	90.6	94.6	69.3	94.3	70.5	0.64

Tabla 3.7: Comparación con otros trabajos que realizan clasificación automática de las fases del sueño. C representa la cantidad de clases en las que se realiza la clasificación, 2 corresponde a despierto y dormido, 4 a despierto, sueño liviano, sueño profundo y REM y 5 a la clasificación según AASM. N es la cantidad de sujetos en la base de datos. Se presenta en gris el resultado correspondiente al algoritmo desarrollado en este capítulo. En el trabajo de Malik y col. [36] se presentan los resultados al evaluar un algoritmo entrenado utilizando un subconjunto de la base de datos CGMH que fue testeado en diferentes bases de datos.

obtenemos mejores resultados, pero sacrificamos exactitud en los límites de las transiciones entre las fases despierto y dormido y, además, probablemente nos perderemos transiciones cortas. El desarrollo de este algoritmo está en una etapa inicial, para poder ser aplicado en la práctica clínica antes es necesario realizar una evaluación a gran escala del método de forma de validar su utilización. No obstante, si el interés no está en la dinámica del sueño, sino en obtener con el menor error posible una estimación del TTS, la opción de utilizar  $L$  grandes es absolutamente válida.

### 3.3.3. Especificaciones técnicas de las señales

Las señales utilizadas en este trabajo tienen una frecuencia de muestreo de 1 Hz. Pueden generarse algunas confusiones, ya que la AASM estipula como deseable la utilización de 25 Hz en oximetría, siendo 10 Hz el mínimo recomendado. Sin embargo, no hay recomendaciones específicas respecto a la frecuencia de muestreo de la señal de HR [8]. Considerando el rango de frecuencias de HR posibles –menos a 200 latidos por minuto–, vemos que no existe necesidad de frecuencias de muestreo mucho más altas que la utilizada.

Existen oxímetros que presentan diferentes frecuencias de muestreo para reportar el HR. La utilización de una frecuencia de muestreo diferente podría cambiar los desempeños obtenidos, ya que muchas de las características varían con este parámetro. En caso de aplicar este sistema a señales con una mayor frecuencia de muestreo, sería necesario sub-muestrear la señal. Por otro lado, utilizar señales con una frecuencia de muestreo menor que 1 Hz no es posible sin realizar nuevamente todos los experimentos para encontrar nuevos parámetros óptimos.

### 3.3.4. Comparación con estado del arte

Existen muchísimos trabajos previos que realizan la clasificación automática de las fases del sueño. Una buena parte de ellos han sido analizados en la sección 2.6, donde se describieron y presentaron los resultados obtenidos por estos trabajos. En la tabla 3.7 volveremos a presentar la tabla de resultados presentada en esa sección, pero añadiendo el mejor resultado obtenido por este desarrollo, para que sean fácilmente comparables. Esta práctica se repetirá en cada uno de los siguientes capítulos en que se presenten nuevos desarrollos.

Los resultados que se han obtenido hasta aquí son bastante esperanzadores, ya que abordan una importante limitación de todos los métodos de cribado de apnea del sueño basados únicamente en desaturación. El riesgo de sobreajuste (*overfitting*) de nuestro algoritmo es muy bajo, ya que hemos utilizado un número de registros enorme que han sido obtenidos en condiciones reales. También es necesario comentar que, si bien hemos aplicado varias estrategias con el objetivo de prevenir el desbalance de clases, puede observarse que el método aún conserva un sesgo hacia la clase mayoritaria que deberíamos abordar en el futuro.

### 3.3.5. Diseño del método y utilización subsiguiente

En este capítulo hemos presentado el desarrollo de un algoritmo para clasificar las fases del sueño en despierto y dormido utilizando únicamente señales de HR proveniente de un oxímetro de pulso. Para resumir, los principales pasos metodológicos que deben ser aplicados para repetir el esquema con otras bases de datos es el siguiente:

1. Preprocesar la señal de HR para tener media cero y varianza unitaria, y luego segmentarla en ventanas no solapadas de longitud  $L$ .
2. Extraer características y optimizar sus hiperparámetros utilizando un subconjunto de la base de datos para el diseño del sistema.
3. Seleccionar las características para obtener un subconjunto óptimo para un clasificador dado.
4. Extraer las características óptimas seleccionadas para dicho clasificador utilizando otro subconjunto de la base de datos para evaluar el desempeño del clasificador.

Para reutilizar el método descrito con nuevos pacientes, es necesario aplicar el preprocesamiento descrito en la sección 3.1.1 para estandarizar y segmentar la señal. Luego, deben extraerse las características seleccionadas –y sus respectivos hiperparámetros óptimos– utilizados para entrenar el clasificador. Si bien el diseño descrito en este trabajo es computacionalmente complejo, su utilización, una vez que el sistema ha sido diseñado, es muy rápida.

## 3.4. Conclusiones

En este capítulo se describió el desarrollo de un método para clasificar en forma automática las fases del sueño en despierto y dormido utilizando señales de HR provenientes de un oxímetro de pulso. Ha quedado demostrado que las características relacionadas con la teoría de la información y medidas de complejidad, así como sus extensiones al dominio tiempo-frecuencia son útiles para diferenciar estas dos clases. Asimismo, a partir de este trabajo podemos observar que señales muy simples y de bajo costo como lo son las obtenidas por oxímetros de pulso pueden obtener desempeños que son comparables con señales que contienen muchísima más información.

Se evaluaron diferentes combinaciones de clasificadores y estrategias para abordar el desbalance de clases, dando lugar a tres métodos posibles: FS 1, FS 2 y FS 3. El método FS 1 –FFS-SVM con penalización en los errores en la clase minoritaria– tuvo un desempeño levemente mejor que los otros métodos. Sin embargo, todos los métodos presentan resultados muy similares.

Si bien este algoritmo es una primera incursión en el problema, es necesario continuar sus desarrollos para lograr la obtención de un hipnograma y para implementarlo en un uso clínico. El hecho de que cada segmento se clasifique individualmente y en forma aislada –la clasificación realizada no depende de clasificaciones previas y futuras– es una gran desventaja, ya que la información de segmentos adyacentes al clasificado aporta información. Esto se hace más notorio al considerar longitudes de segmento cada vez más pequeñas. Esto será abordado en los siguientes capítulos, desarrollando dos nuevos métodos que tienen en consideración esta desventaja.

El algoritmo desarrollado en este capítulo dio lugar a las siguientes publicaciones en congresos y revistas científicas indexadas:

- Casal, Ramiro, y Gastón Schlotthauer. "Sleep detection in heart rate signals from photoplethysmography." 2017 XVII Workshop on Information Processing and Control (RPIC). IEEE, 2017.
- Casal, Ramiro, Leandro E. Di Persia, y Gastón Schlotthauer. "Sleep-wake stages classification using heart rate signals from pulse oximetry." *Heliyon* 5.10 (2019): e02529.

## Capítulo 4

# Desarrollo de un algoritmo basado en redes neuronales recurrentes

El trabajo presentado en el capítulo anterior tiene dos importantes debilidades, que de alguna manera están relacionadas entre sí. Por un lado está la cuestión relativa a la longitud de ventana  $L$ . Para mejorar el desempeño es necesario incrementar la longitud de los segmentos considerados. Sin embargo, al incrementar  $L$  la aplicabilidad del algoritmo a una situación real se hace más difícil. Por otro lado, los segmentos de señal son clasificados en forma aislada, sin tener en cuenta su posición relativa en la secuencia. Como hemos visto, el sueño suele tener una estructura temporal medianamente definida. Conocer esta estructura nos puede ayudar a mejorar las clasificaciones realizadas.

Estas debilidades son abordadas mediante el desarrollo de un algoritmo que se presentará en este capítulo, clasificando las fases del sueño en despierto y dormido mediante la aplicación de redes neuronales recurrentes (*recurrent neural networks*, RNN). Para este algoritmo no sólo consideramos como entrada posible la señal de HR, sino que también incorporamos la SpO<sub>2</sub>. Dado que ambas señales provienen del oxímetro de pulso, esta incorporación no incrementa la complejidad del dispositivo final –aunque sí la del algoritmo–. A diferencia del algoritmo anterior, aquí no se hace uso de características diseñadas y extraídas manualmente, sino que las entradas consisten en las señales con un mínimo acondicionamiento previo. El procesamiento y la clasificación que realizan las RNN tienen la ventaja de que, teóricamente, considera información sobre la secuencia entera almacenada en los «vectores de estado», lo que les permite aprender las dependencias temporales de la estructura interna del sueño [105]. Ésto permite obtener desempeños destacables a partir de señales cuya instrumentación es muy simple. De hecho, los resultados obtenidos son comparables con algoritmos que utilizan señales mucho más informativas.

A continuación describiremos el método y los experimentos desarrollados para evaluar arquitecturas con diferentes parámetros y señales de entrada a fin de ver las variaciones en su desempeño. Luego presentaremos y discutiremos los resultados obtenidos. Finalmente se comentará una modificación realizada sobre el algoritmo con el objetivo de lograr una mayor discriminación de las fases del sueño de acuerdo a la AASM –originalmente el algoritmo fue desarrollado para clasificar la señal en despierto y dormido–.

### 4.1. Metodología

En este capítulo se propone un modelo de aprendizaje profundo basado en RNN para clasificar las fases del sueño –primero en despierto y dormido, luego realizando una estadiificación completa según la AASM–. Se evaluaron diferentes arquitecturas basadas en GRU [106], un tipo de RNN que son una variante simplificada de las LSTM [107]. La arquitectura

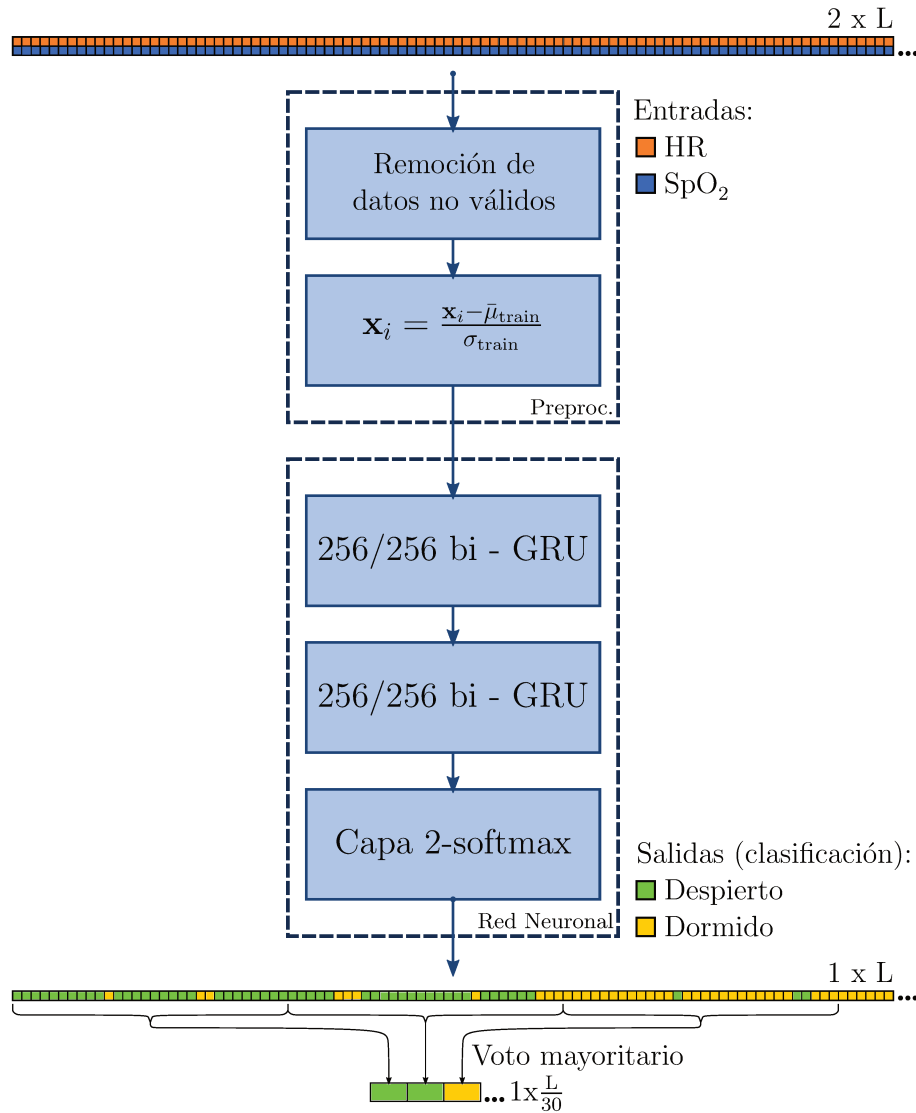


Figura 4.1: Diagrama en bloques del algoritmo desarrollado. El algoritmo consiste en dos capas de bi-GRU y una capa *softmax* para clasificar las salidas de las GRU, que luego son sometidas a una votación mayoritaria para obtener la clasificación final.

de la red consiste en dos capas apiladas de GRU bidireccionales. La bidireccionalidad permite a la red explotar información tanto pasada como futura. Los datos que serán utilizados como entrada tienen un preprocesamiento muy simple. Luego, son procesados por las GRU y, finalmente, clasificados por una capa *softmax*. Este algoritmo clasifica las fases del sueño para cada uno de los elementos de la señal –la entrada y la salida de la GRU tienen la misma longitud–. Vemos entonces que en este algoritmo tenemos una clasificación por segundo, cuando el requisito establecido por la AASM es una clasificación cada 30 segundos. Por este motivo, se realizó un voto mayoritario considerando segmentos de 30 segundos. En la figura 4.1 se muestra un diagrama de bloques del algoritmo.

La implementación de este algoritmo se realizó utilizando Python 3 y PyTorch. Además, se utilizó la base de datos SHHS 1, utilizando los mismos 5000 pacientes que se usaron para el algoritmo anterior. Sin embargo en este caso se optó por un esquema de validación más simple, dividiendo en forma aleatoria la base de datos en tres subconjuntos: 2500 sujetos fueron seleccionados para entrenar la red, y luego dos subconjuntos de 1250 sujetos cada uno se utilizaron para validar y testear los modelos [108, 109].

## Conjuntos de entrenamiento, validación y testeo vs. $k$ -fold CV

A diferencia del caso anterior, en este trabajo optamos por un esquema de validación más simple, priorizando la velocidad para realizar los experimentos. Entrenar estas redes utilizando  $k$ -fold CV es infactible en términos computacionales, hubiese llevado varios meses considerando la capacidad de cómputo disponible. Si bien entendemos que la forma más simple y directa para comparar los métodos es utilizando exactamente el mismo esquema de validación, en este apartado explicaremos por qué razones teóricas y prácticas los resultados aún pueden ser comparables.

Con el objetivo de comparar ambos métodos, podemos pensar en un esquema basado en conjuntos de entrenamiento y validación –conocido como *holdout* en inglés– como la versión más simplificada de  $k$ -fold CV, con  $k = 1$ . Recordemos que, según la explicación dada en el capítulo anterior respecto a la selección de  $k$  en un esquema de  $k$ -fold CV, cuanto mayor es el tamaño de la base de datos menor puede ser el  $k$  seleccionado [77].

Basándonos en el trabajo de Arlot y cols. [110] podemos afirmar que utilizando el esquema de entrenamiento, validación y testeo podemos alcanzar resultados equivalentes a los obtenidos mediante  $k$ -fold CV, pero más rápido. En ese trabajo, siguiendo a Nadeau y Bengio [111], los autores estudian la varianza de una función de costo dada con respecto al tamaño de los conjuntos de entrenamiento y validación. Si se deja el tamaño del conjunto de entrenamiento fijo, se concluye que a medida que se incrementa la cantidad de datos para validación disminuye la varianza en la estimación de la función de costo. Es decir, una mayor cantidad de datos de validación conduce a una estimación más exacta de la función de costo. Por lo tanto, si tenemos suficiente cantidad de datos para validación, no es estrictamente necesario realizar esquemas de evaluación más complejos como  $k$ -fold CV –de hecho, estos esquemas de validación cruzada surgen como una forma de obtener mejores estimaciones al contar con poca cantidad de datos–. Si resulta de interés para el lector ahondar en este tema, puede encontrarse el desarrollo completo en [110].

Visto de otra forma, si se aplica  $k$ -fold CV sobre una base de datos lo suficientemente grande –como la utilizada en la presente tesis–, los conjuntos de validación tendrán un gran tamaño. Por esto, según Arlot y cols., es de esperar que la variación para los diferentes subconjuntos de entrenamiento y validación sea muy pequeña –ya que a mayor tamaño del conjunto de validación, menor varianza–, y, por lo tanto, podemos pensar en la utilización de  $k = 1$ . Luego, además, se utilizó un conjunto extra de testeo para corroborar que la estimación lograda en el conjunto de validación sea correcta.

Por estos motivos, y añadiendo que en este trabajo se utiliza un subconjunto extra de testeo para corroborar que el desempeño obtenido en el conjunto de validación puede ser generalizado a datos que no han sido vistos hasta ahora, podemos afirmar que es factible comparar los desempeños obtenidos con este algoritmo y el anterior a pesar de estar basados en esquemas de validación diferentes.

Kohavi sigue un razonamiento complementario al expuesto anteriormente [112]: suponiendo que la exactitud de la estimación incrementa con la cantidad de datos, realizar la partición en entrenamiento y validación conduce a una estimación pesimista, ya que sólo se utiliza una porción de los datos para entrenar el clasificador. Entonces, cuanto mayor sea la cantidad de datos que dejamos para el conjunto de validación, mayor será el sesgo del estimador. Esta idea es observada en forma experimental en los resultados de los experimentos realizados por Yadav y Shukla [113].

### 4.1.1. Preprocesamiento

Se aplicó un preprocesamiento muy simple sobre las señales de HR y SpO<sub>2</sub>. En primer lugar, se efectuó la remoción de las porciones inválidas de la señal y su reemplazo mediante la

interpolación entre los datos confiables anterior y posterior, como se describió en el capítulo 2.

Luego, se estandarizaron las señales antes de ser utilizadas como entradas para las GRU. Para que las señales tengan media cero y varianza unitaria, se calculó la media global y el desvío estándar de la base de datos de entrenamiento, utilizando luego esos valores para estandarizar las bases de datos de entrenamiento, validación y testeo.

### 4.1.2. Redes neuronales recurrentes: LSTM y GRU

Las RNN son una familia de redes neuronales que han tenido un gran auge para el procesamiento de datos secuenciales. En las RNN las secuencias son procesadas de a un elemento por vez, almacenando en un «estado interno» información sobre el historial completo de la entrada –todas las muestras previas–. Esta persistencia de la información es alcanzada mediante conexiones internas –conocidas como conexiones recurrentes– que realimentan las salidas previas en la entrada [105]. Las RNN clásicas tienen dificultad para entrenarse dado que los gradientes retropropagados tienden a crecer en forma exponencial o desvanecerse rápidamente [114].

Las LSTM fueron propuestas por Hochreiter en el año 1997 [107] para superar el problema del desvanecimiento de gradientes. En este tipo de arquitecturas es posible aprender las dependencias de largo plazo mediante estados internos persistentes en cada una de las unidades de LSTM. Estos estados internos son modificados mediante estructuras denominadas «compuertas» que controlan el flujo de información. Puede verse a los estados internos como un camino a través del cual los gradientes pueden persistir durante largas duraciones. Formalmente, para un tiempo  $t$  y una unidad  $i$ , se calculan las siguientes ecuaciones:

$$\begin{aligned}
 \mathbf{f}_i^{(t)} &= \sigma(\mathbf{W}_{f,i}[\mathbf{h}_i^{(t-1)}, \mathbf{x}^{(t)}] + \mathbf{b}_{f,i}), \\
 \mathbf{g}_i^{(t)} &= \sigma(\mathbf{W}_{g,i}[\mathbf{h}_i^{(t-1)}, \mathbf{x}^{(t)}] + \mathbf{b}_{g,i}), \\
 \tilde{\mathbf{s}}_i^{(t)} &= \tanh(\mathbf{W}_{s,i}[\mathbf{h}_i^{(t-1)}, \mathbf{x}^{(t)}] + \mathbf{b}_{s,i}), \\
 \mathbf{s}_i^{(t)} &= \mathbf{f}_i^{(t)} \circ \mathbf{s}_i^{(t-1)} + \mathbf{g}_i^{(t)} \circ \tilde{\mathbf{s}}_i^{(t)}, \\
 \mathbf{o}_i^{(t)} &= \sigma(\mathbf{W}_{o,i}[\mathbf{h}_i^{(t-1)}, \mathbf{x}^{(t)}] + \mathbf{b}_{o,i}), \\
 \mathbf{h}_i^{(t)} &= \mathbf{o}_i^{(t)} \circ \tanh(\mathbf{s}_i^{(t)})
 \end{aligned} \tag{4.1}$$

donde  $\mathbf{f}_i^{(t)}$ ,  $\mathbf{g}_i^{(t)}$  y  $\mathbf{o}_i^{(t)}$  son las compuertas de «olvido» (*forget*), «entrada» (*input*) y «salida» (*output*), respectivamente.  $\mathbf{s}_i^{(t)}$  es el estado interno y  $\mathbf{h}_i^{(t)}$  es la salida de la  $i$ -ésima unidad de LSTM.  $\mathbf{W}_{(\cdot),i}$  y  $\mathbf{b}_{(\cdot),i}$  son los pesos y sesgos, y  $\sigma$  representa una función sigmoidea definida como  $\sigma(y) = 1/(1 + \exp(-y))$  [109]. El operador  $\circ$  representa un producto punto a punto, mientras que los vectores entre corchetes representan una concatenación vertical. La entrada a la red está representada por  $\mathbf{x}$ .

Podemos ver que el nuevo estado interno  $\mathbf{s}_i^{(t)}$  es una función del último estado interno –la memoria de la unidad–, de una versión «filtrada» de las entradas pasadas y de la entrada actual –actualización–. Las compuertas de olvido y de entrada controlan la memoria y las actualizaciones, respectivamente. Por otro lado, la salida  $\mathbf{h}_i^{(t)}$  es una versión «filtrada» de los estados internos multiplicados por la compuerta de salida.

Se han propuesto varias modificaciones a las LSTM, principalmente con el objetivo de simplificar su estructura interna. Una de las variaciones más populares son las GRU. La principal diferencia entre las LSTM y las GRU es que en esta última existe una única compuerta que unifica las compuertas de «entrada» y «olvido». Es decir, el estado interno es modificado por una única compuerta. Las ecuaciones de actualización para las GRU son

como sigue:

$$\begin{aligned}
\mathbf{u}_i^{(t)} &= \sigma(\mathbf{W}_{u,i}[\mathbf{h}_i^{(t-1)}, \mathbf{x}^{(t)}] + \mathbf{b}_{u,i}), \\
\mathbf{r}_i^{(t)} &= \sigma(\mathbf{W}_{r,i}[\mathbf{h}_i^{(t-1)}, \mathbf{x}^{(t)}] + \mathbf{b}_{r,i}), \\
\tilde{\mathbf{h}}_i^{(t)} &= \tanh(\mathbf{W}[\mathbf{r}_i^{(t)}\mathbf{h}_i^{(t-1)}, \mathbf{x}^{(t)}] + \mathbf{b}_{s,i}), \\
\mathbf{h}_i^{(t)} &= (1 - \mathbf{u}_i^{(t)}) \circ \mathbf{h}_i^{(t-1)} + \mathbf{u}_i^{(t)} \circ \tilde{\mathbf{h}}_i^{(t)}
\end{aligned} \tag{4.2}$$

donde  $\mathbf{u}$  representa la compuerta de «actualización» (*update*) y  $\mathbf{r}$  la compuerta de «reinicio» (*reset*). De nuevo, el operador  $\circ$  representa un producto punto a punto, mientras que los vectores entre corchetes representan una concatenación vertical. La compuerta de «actualización» decide si copiar –en uno de los extremos de la sigmoide– o ignorar –en el otro extremo– el último vector de estado  $\mathbf{h}_i^{(t-1)}$  mediante su reemplazo con el nuevo candidato a vector de estado  $\tilde{\mathbf{h}}_i^{(t)}$ . La compuerta de «reinicio» controla qué parte del estado interno actual es utilizado para calcular el siguiente estado [109]. Las entradas a las GRU están representadas por  $\mathbf{x}$ .

Como ha sido presentado hasta ahora, las RNN tienen un comportamiento causal ya que los estados internos únicamente almacenan información de las entradas pasadas y presentes. En la aplicación a abordar en este trabajo es preferible que la clasificación dependa de la secuencia completa. Es decir, tener información completa tanto del pasado como del futuro permitiría un mejor entendimiento del contexto mediante la **eliminación ambigüedades**. Schuster y Paliwal [115] propusieron las RNN bidireccionales al combinar dos RNN. Una de ellas procesa las entradas en un orden temporal, mientras que la otra las procesa en orden inverso. De esta forma, mediante estas dos RNN independientes es posible explotar información tanto del pasado como del futuro.

En nuestro trabajo hemos utilizado arquitecturas basadas en GRU en lugar de LSTM, dado que hacen un menor uso de memoria. Además, dado que el procesamiento en nuestro caso se hace fuera de línea, se consideraron GRU bidireccionales.

### 4.1.3. Capa *softmax*

Las GRU son responsables de aprender las reglas de transición y las dinámicas temporales de la secuencia. Luego, una capa *softmax* se utiliza para clasificar cada elemento de la secuencia en dos clases: despierto y dormido. Para ello, se aplica una transformación no lineal haciendo:

$$\mathbf{y} = \text{ReLU}(\mathbf{W}\mathbf{x} + \mathbf{b}) \tag{4.3}$$

donde  $\mathbf{W}$  son los coeficientes,  $\mathbf{b}$  es el sesgo y  $\mathbf{x}$  es la entrada a esta capa –es decir, la salida de la segunda GRU bidireccional–. Como función de activación se utilizó una unidad lineal rectificadora (*rectified linear unit* ReLU), definida como  $\text{relu}(x) = \max(0, x)$ . Este vector de salida  $\mathbf{y}$  es transformado en una probabilidad de pertenencia para cada clase mediante una función *softmax*.

Se utilizó la entropía cruzada (*cross-entropy*) como función de costo, la cual para un problema de clasificación binaria puede ser definida formalmente como:

$$CE(y, c) = \begin{cases} -\log(y) & \text{si } c = 1 \\ -\log(1 - y) & \text{en otro caso,} \end{cases} \tag{4.4}$$

$c \in \{\pm 1\}$  especifica la clase a la que pertenece la muestra, mientras que  $y \in [0, 1]$  es la probabilidad estimada por la capa *softmax*. Por otro lado, se utilizó el optimizador Adam [116] para encontrar los parámetros óptimos de la red completa.



#### 4.1.4. Voto mayoritario

El algoritmo diseñado produce una clasificación para cada uno de los elementos de las señales de entrada –una clasificación por segundo, dado que la frecuencia de muestreo es de 1 Hz–. Esta resolución es mucho más alta que los 30 segundos requeridos por la AASM. Con el objetivo de adaptar la salida de este algoritmo a la «resolución» estándar, realizamos una votación mayoritaria considerando segmentos de 30 muestras –30 segundos–. El resultado reportado corresponde a esta operación. No está demás mencionar que los desempeños obtenidos antes y después de la votación no variaron en forma significativa. Es decir, el sistema diseñado no tiende a generar clasificaciones aisladas de una clase durante pocos segundos, sino que los cambios en las salidas del clasificador son más bien suaves.

## 4.2. Resultados

En esta sección se describirán varios de los experimentos llevados a cabo para evaluar el desempeño de las redes y cómo éste varía al modificar los parámetros de la arquitectura. También se presentarán las medidas de desempeño utilizadas, los parámetros de entrenamiento y el optimizador seleccionado, y se explicarán las etapas de entrenamiento y testeado de la red.

### 4.2.1. Medidas de desempeño

Para evaluar el desempeño se utilizaron las mismas medidas que en el capítulo anterior: exactitud, sensibilidad, especificidad, valor de predicción positivo y valor de predicción negativo. Además, calculamos el coeficiente Kappa de Cohen [117]. Nuevamente las medidas se calcularon individualmente para cada uno de los pacientes, para finalmente reportar el valor promedio en la tabla 4.1. Recordemos que para calcular estas medidas consideramos la clase «despierto» como la clase positiva.

Como en este trabajo superamos la limitación que nos imponía la longitud de ventana del capítulo anterior, podemos transferir directamente el resultado obtenido a una estimación del TTS. Para medir el error en esta estimación –que es el error más importante para nuestra aplicación–, se obtuvieron dos medidas de error para las estimaciones del TTS. El error medio absoluto se define como:

$$\mathbf{E}_1 = \frac{1}{N} \sum_{i=1}^N |\text{TST}_i - \hat{\text{TST}}_i| \quad (4.5)$$

donde  $N$  es el número total de pacientes y  $\hat{\text{TST}}_i$  es la estimación del TTS, obtenida al contar los segmentos clasificados como dormido. De forma similar, se calculó el error medio absoluto porcentual, definido como:

$$\mathbf{E}_2 = \frac{1}{N} \sum_{i=1}^N \frac{|\text{TST}_i - \hat{\text{TST}}_i|}{\text{TST}_i} \cdot 100. \quad (4.6)$$

### 4.2.2. Parámetros de la red y del optimizador

Se entrenaron varios modelos variando el número de neuronas ocultas de las GRU bidireccionales con el objetivo de evaluar el tamaño óptimo de esta capa. Los valores explorados fueron 64, 128 y 256. Se entrenaron los modelos con las entradas preprocesadas, evaluando dos alternativas diferentes: utilizar únicamente HR –como en el algoritmo anterior– y utilizar tanto HR como SpO<sub>2</sub> –la recuperación del oxígeno en sangre en la señal de SpO<sub>2</sub> podría tener

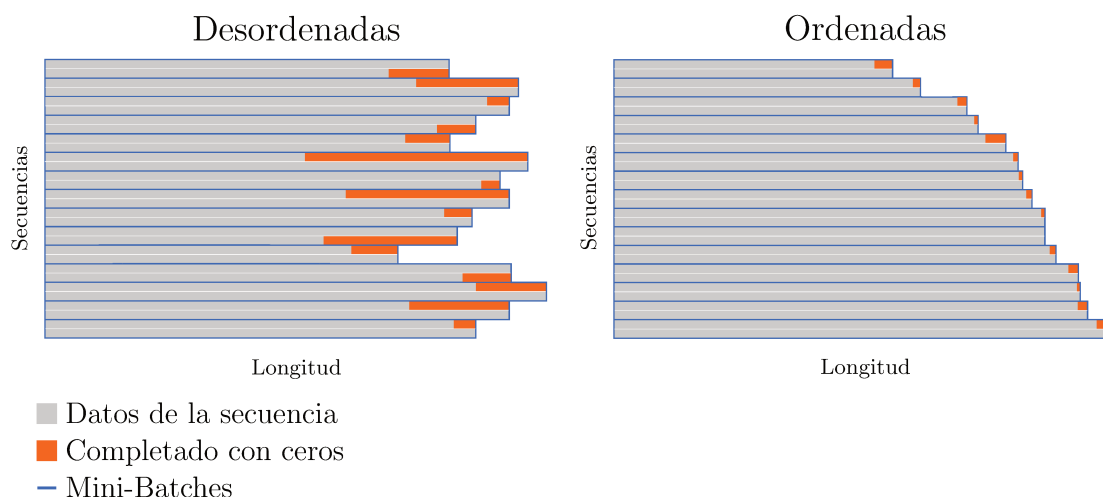


Figura 4.2: Efecto del ordenamiento de las secuencias según su longitud para evitar el relleno con ceros excesivo, permitiendo un procesamiento más rápido.

información útil asociada a los despertares—. Se evaluaron todas las combinaciones posibles de estos parámetros.

Ya dijimos que hemos utilizado el optimizador Adam. Los parámetros de este optimizador son la tasa de aprendizaje  $\alpha$ , la tasa de decaimiento exponencial para el primer momento  $\beta_1$  y la tasa de decaimiento exponencial para el segundo momento  $\beta_2$ . Estos parámetros se establecieron en  $\alpha = 10^{-4}$ ,  $\beta_1 = 0.9$  y  $\beta_2 = 0.99$ . Relacionado con el optimizador, es necesario establecer el tamaño del *mini-batch* utilizado para entrenar la red. Debido a restricciones de memorias de las GPU utilizadas, el tamaño de *mini-batch* establecido fue 2.

Las entradas al algoritmo son secuencias cuyo tamaño es variable –cada registro tiene una duración determinada para los pacientes—. Las RNN son capaces de manejar datos con longitud variable ya que realizan el procesamiento de a un elemento por vez. Sin embargo, las secuencias que forman parte de un *mini-batch* deben tener la misma longitud para conformar una matriz. Por este motivo, se completaron las secuencias con ceros (*padding*) de forma tal que todas las secuencias dentro de un *mini-batch* tengan la misma longitud que la secuencia más larga del *mini-batch*. Con el objetivo de reducir la cantidad de elementos agregados, las entradas se ordenaron según su longitud [109]. Es decir, los sujetos cuya longitud de registro es similar se agruparon en un mismo *mini-batch*. El efecto de esta operación se muestra en la figura 4.2. Lógicamente, las salidas del sistema correspondientes a los elementos «agregados» no se tuvieron en cuenta para calcular la función de costo. La longitud promedio de los datos utilizados es  $30432 \pm 2175$ , mientras que las longitudes máxima y mínima son 35970 y 10800 respectivamente.

### 4.2.3. Entrenamiento de la red

El mejor modelo se entrenó y seleccionó utilizando los subconjuntos de entrenamiento y validación. Para ello, los parámetros de la GRU se ajustaron en forma iterativa utilizando el subconjunto de entrenamiento con el objetivo de minimizar la función de costo –entropía cruzada—. Al finalizar cada época, el modelo fue evaluado utilizando el subconjunto de validación. El número máximo de épocas para entrenar el sistema se estableció en 100, cantidad que según los experimentos fue suficiente para encontrar los modelos «óptimos». Los entrenamientos se finalizaron mediante «detención temprana» (*early stopping*), seleccionando el modelo que presentó la mejor exactitud en el subconjunto de validación. La figura 4.3 muestra el proceso de entrenamiento en uno de los experimentos realizados y el modelo seleccionado

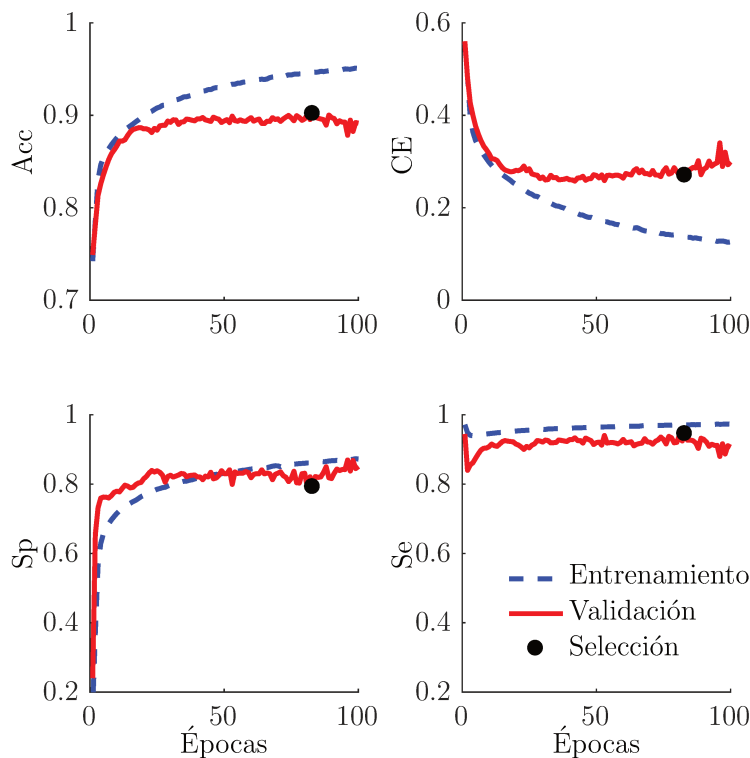


Figura 4.3: Entrenamiento del sistema (considerando 256 neuronas ocultas) y selección del modelo. Entradas: HR y SpO<sub>2</sub>. La línea punteada (azul) y la línea sólida (roja) son los desempeños en los subconjuntos de entrenamiento y validación, respectivamente. Las medidas de desempeño para este modelo son Acc = 90.13 %, Se = 94.13 % y Sp = 80.26 %.

que poseía la máxima exactitud. Los demás experimentos presentaron un comportamiento muy similar. Finalmente, el desempeño del modelo «óptimo» obtenido se evaluó utilizando el subconjunto de testeo [108, 109].

Similar a lo que se realizó en el capítulo anterior, también se evaluó el desempeño al guiar el entrenamiento utilizando el error  $\mathbf{E}_1$  como función de costo para optimizar la red, ya que es la medida más importante para nuestra aplicación. Sin embargo, los resultados obtenidos no fueron satisfactorios y la red tuvo un entrenamiento mucho más inestable.

#### 4.2.4. Desempeño en el subconjunto de testeo

El desempeño de los sistemas desarrollados se evaluó en el subconjunto de testeo de 1250 sujetos. Las medidas reportadas para medir el desempeño no son afectadas por el desbalance que existe entre las clases despierto/dormido [75]. El índice Kappa de Cohen que hemos calculado permite medir la tasa de acuerdos entre las clasificaciones hechas por el algoritmo y la referencia. Además, como ya mencionamos, se calcularon los errores  $\mathbf{E}_1$  y  $\mathbf{E}_2$  para tener una medida del error relacionada con la aplicación para la cual fue diseñado el método.

En la tabla 4.1<sup>1</sup> se muestran las medidas de desempeño obtenidas en los subconjuntos de entrenamiento, validación y testeo para cada una de las arquitecturas de red evaluadas. Los desempeños mejoran levemente al añadir SpO<sub>2</sub> como entrada. El mejor resultado basado en la exactitud, en el valor de entropía cruzada obtenido y en el coeficiente Kappa de Cohen se obtuvo utilizando HR y SpO<sub>2</sub> como entradas y una arquitectura basada en dos

<sup>1</sup>Abreviaciones en la tabla 4.1 y 4.5: Acc para exactitud, Se para sensibilidad, Sp para especificidad, PPV para valor predictivo positivo, NPV para valor predictivo negativo,  $\kappa$  para coeficiente Kappa de Cohen,  $\mathbf{E}_1$  para el error absoluto medio y  $\mathbf{E}_2$  para el error absoluto medio porcentual.

Entradas	Red	Db	Acc	Se	Sp	PPV	NPV	$\kappa$	$E_1$	$E_2$
HR+SpO <sub>2</sub>	2 GRU-(256)	Entr.	94.3	96.8	85.7	94.8	91.4	0.840	0.219	3.88 %
		Val.	90.4	94.6	79.6	91.9	85.5	0.741	0.491	9.30 %
		Test	<b>90.1</b>	94.1	80.3	92.1	<b>84.7</b>	<b>0.740</b>	0.484	8.90 %
HR+SpO <sub>2</sub>	2 GRU-(128)	Entr.	92.6	94.3	86.4	94.8	85.4	0.798	0.269	4.83 %
		Val.	89.8	92.5	82.8	92.9	81.7	0.734	0.483	8.97 %
		Test	89.7	92.5	82.6	<b>92.8</b>	81.6	0.733	0.475	8.62 %
HR+SpO <sub>2</sub>	2 GRU-(64)	Entr.	91.0	93.4	84.5	93.7	82.5	0.761	0.354	6.33 %
		Val.	89.7	92.7	82.8	92.7	81.1	0.732	0.469	8.70 %
		Test	89.6	92.5	<b>83.1</b>	<b>92.8</b>	80.5	0.731	<b>0.456</b>	<b>8.35</b> %
HR	2 GRU-(256)	Entr.	88.9	92.0	80.9	92.2	79.6	0.710	0.456	8.43 %
		Val.	88.2	91.8	79.4	91.5	78.8	0.691	0.523	9.85 %
		Test	88.0	91.3	80.0	91.6	78.4	0.691	0.536	9.68 %
HR	2 GRU-(128)	Entr.	90.7	95.1	78.8	91.8	85.6	0.745	0.399	7.42 %
		Val.	89.2	94.5	76.5	90.6	83.6	0.710	0.508	9.99 %
		Test	89.1	<b>94.2</b>	76.8	90.7	83.6	0.710	0.522	9.08 %
HR	2 GRU-(64)	Entr.	89.0	93.7	77.3	91.0	82.1	0.706	0.482	9.18 %
		Val.	88.4	93.5	76.0	90.4	81.6	0.691	0.525	10.26 %
		Test	88.2	93.0	76.6	90.5	81.0	0.688	0.548	10.12 %

Tabla 4.1: Desempeño de las redes neuronales en los subconjuntos de entrenamiento (Entr.), validación (Val.) y testeo (Test).

capas de GRU con 256 neuronas ocultas. Teniendo en cuenta las medidas de desempeño reportadas, los resultados mejoran a medida que se incrementa el tamaño de la capa oculta. Consecuentemente, al incrementar esta capa también incrementa el riesgo de sobreajustar la red neuronal.

Dado que las secuencias utilizadas poseen una longitud muy grande –un registro de 8 horas tiene 28800 muestras–, el entrenamiento de la red es computacionalmente costoso. Además, este costo se incrementa al incrementar el tamaño de la red. Por este motivo, existe una relación de compromiso entre el tamaño de la capa oculta y el costo computacional para entrenar la red.

Se realizó la prueba de Friedman considerando todas las redes desarrolladas para evaluar si los resultados obtenidos tienen una diferencia significativa, obteniendo un valor  $p = 1.55 \times 10^{-131}$ . Como resultado de esta prueba se concluye que la red que obtiene el mejor resultado –HR y SpO<sub>2</sub>, además de 256 unidades en la capa oculta– tiene una diferencia significativa respecto a las otras.

En la figura 4.4 se muestran tres hipnogramas simplificados obtenidos mediante la red «óptima»: un hipnograma en torno al error promedio y los otros dos con un desempeño superior e inferior al promedio.

#### 4.2.5. Clasificación de fases del sueño según AASM

Alentados por el buen desempeño obtenido en los algoritmos presentados, introdujimos algunas modificaciones en las redes con el objetivo de obtener una mayor discriminación en las fases del sueño. En este apartado presentaremos algunos resultados preliminares de estos experimentos a partir de los cuales podría, en el futuro, desprenderse un desarrollo lateral a la línea desarrollada en la presente tesis.

Simplemente se procedió a modificar la capa de salida, considerando ahora una capa

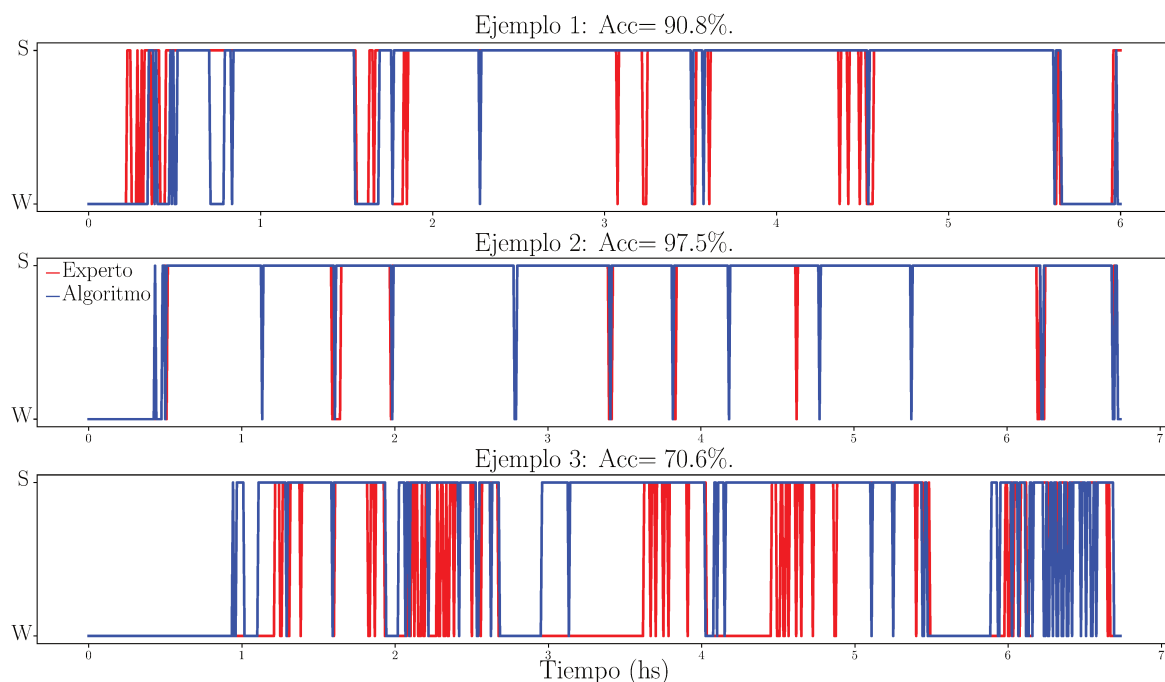


Figura 4.4: Hipnogramas simplificados obtenidos mediante clasificación con la red neuronal «óptima» desarrollada. En la parte superior de la figura se muestran los hipnogramas de referencia (etiquetado por un conjunto de expertos) y la clasificación realizada por el algoritmo en un sujeto para el que se obtuvo un desempeño cercano al promedio. En el medio se muestra otro ejemplo, esta vez con un desempeño superior al promedio. Abajo, en cambio, se muestra un tercer ejemplo para el cual el desempeño fue inferior al promedio. En líneas rojas se presenta el etiquetado realizado por expertos, mientras que en líneas azules se presenta el etiquetado realizado por el algoritmo.

5-*softmax* capaz de clasificar el sueño en las fases W, N1, N2, N3 y REM. Se utilizó la arquitectura con la que se obtuvo el mejor desempeño en los experimentos anteriores. Es decir, 2-GRU con 256 neuronas en la capa oculta. Se consideraron como entradas HR y SpO<sub>2</sub>, utilizando además los mismos parámetros de entrenamiento que en los experimentos anteriores, con excepción de la cantidad de épocas máxima, la cual fue necesario incrementar a 120 épocas dado que la convergencia de este experimento era un poco más lenta. Finalmente, el esquema de evaluación utilizado fue idéntico al descrito anteriormente, entrenando y validando el algoritmo con los subconjuntos de entrenamiento y validación para, luego, calcular su desempeño en un subconjunto independiente de testeo.

El desempeño de la red propuesta fue evaluado en la base de datos de testeo, compuesta por 1250 sujetos. A continuación mostraremos una serie de tablas con los resultados obtenidos para nuestro algoritmo. Como se mencionó en el capítulo 2, diferentes trabajos del estado del arte establecen diferentes agrupaciones de las fases del sueño para realizar la clasificación. En general, cuando los métodos son diseñados para cribado se realizan agrupaciones más grandes. Para mejorar la comprensión en la comparación de los métodos, presentaremos un conjunto de tablas con **los resultados** obtenidos y los principales métodos del estado del arte que realizan la misma agrupación de fases. En la tabla 4.2 se presenta el desempeño de nuestro algoritmo y algoritmos realizados por terceros que clasifican según la normativa de la AASM. Hasta donde conocemos, no existen algoritmos de cribado que clasifiquen las fases del sueño según la AASM. Por este motivo, fue necesario compararse contra métodos que utilizan EEG. El trabajo de Supratak y cols. [28] se evaluó en dos bases de datos diferentes. Reportamos ambos resultados en la tabla para que el lector pueda tener una dimensión de la variación en los desempeños de un método por el sólo hecho de cambiar de base de datos.

Método	Entradas	N	Acc	Acc <sub>W</sub>	Acc <sub>N1</sub>	Acc <sub>N2</sub>	Acc <sub>N3</sub>	Acc <sub>REM</sub>	$\kappa$
<b>256-biGRU</b>	HR+SpO <sub>2</sub>	5000	73.2	85.6	0	75.7	60.8	75.4	0.63
CWT feat.+RF [30]	EEG	10	82.6	89.6	43.2	84.8	69.4	76.4	0.76
CNN+RNN [28]	EEG	62	86.2	87.3	59.8	90.3	81.5	89.3	0.80
CNN+RNN [28]	EEG†	20	82.0	84.7	46.6	85.9	84.8	82.4	0.76
CNN + RNN [65]	EEG†	61	84.3	89.2	52.2	86.8	85.1	85.0	0.79

Tabla 4.2: Desempeños y comparación con la literatura según AASM. Se presentan tres trabajos con los mejores desempeños obtenidos. †Estos trabajos utilizaron la misma base de datos, por lo que pueden ser comparados directamente.

Método	Entradas	N	Acc	Acc <sub>W</sub>	Acc <sub>Light</sub>	Acc <sub>Deep</sub>	Acc <sub>REM</sub>	$\kappa$
<b>256-biGRU</b>	HR+SpO <sub>2</sub>	5000	75.7	85.6	73.7	60.1	75.4	0.65
Feat.+LDA [38]	PPG + acc	60	68.7	69.3	69.2	62.5	71.6	0.52

Tabla 4.3: Desempeño y comparación con la literatura al clasificar en W, liviano, profundo y REM.

El trabajo de Mousavi y cols. [65] utiliza una de las bases de datos que usó Supratak, por lo que pueden compararse completamente ambos desempeños. En la tabla 4.3 se agrupan las fases de sueño liviano N1 y N2 y se realiza una comparación con el único método de cribado que realiza esta agrupación, desarrollado por Beattie y cols. [38]. Finalmente, en la tabla 4.4 la clasificación simplemente discrimina entre sueño REM y no REM (agrupando N1, N2 y N3), comparándose con los trabajos de Yücelbaş y cols. [35] y de Xiao y cols. [118].

En función de los resultados obtenidos podemos observar que se logra un desempeño relativamente aceptable. Lo más notorio es la incapacidad de algoritmo para lograr clasificar la fase N1, una fase muy minoritaria en relación

En la figura 4.5 se muestran tres hipnogramas obtenidos mediante la red «óptima»: un hipnograma para el cual se obtuvo un resultado en torno al error promedio y los otros dos con un desempeño superior e inferior al promedio. Puede verse que en general el algoritmo sigue bastante bien la clasificación realizada por los expertos.

En la siguiente sección discutiremos lo desarrollado en este capítulo. Sin embargo, la discusión de los resultados obtenidos al realizar esta modificación del algoritmo para clasificar las fases del sueño según la AASM y su comparación con los principales trabajos del estado del arte deberá esperar hasta el próximo capítulo, donde se describirá el desarrollo de otro algoritmo para clasificación de las fases del sueño, siguiendo una estructura análoga a la presentada en este capítulo y se abordará lo que dejamos pendiente aquí.

### 4.3. Discusión

En este capítulo se describió el desarrollo de una red neuronal basada en GRU bidireccionales para clasificar las fases del sueño en despierto y dormido utilizando únicamente señales provistas por un oxímetro de pulso. Como entradas al clasificador se utilizaron señales «crudas» en lugar de características diseñadas manualmente como en el algoritmo anteriormente presentado.

Muchas de las consideraciones generales respecto de la variabilidad en los oxímetros de pulso, así también como de las ventajas y dificultades de su utilización para el cribado de las patologías del sueño, son idénticas a las descritas en el capítulo anterior. Hemos mencionado las desventajas de la señal de oximetría de pulso en relación a su baja frecuencia de muestreo y exactitud. En el algoritmo anterior se comprobó que la clasificación mejoraba

Método	Entradas	N	Acc	Acc <sub>W</sub>	Acc <sub>NREM</sub>	Acc <sub>REM</sub>	$\kappa$
<b>256-biGRU</b>	HR+SpO <sub>2</sub>	5000	85.2	85.6	87.5	75.4	0.74
Feat.+RF [35]	ECG (P1)	5	78.1	74.3	82.8	40.9	0.57
	ECG (H1)	5	87.1	83.5	91.1	52.1	0.74
	ECG (P2)	16	77.0	–	–	–	0.57
	ECG (H2)	2	76.8	–	–	–	0.43
Feat.+RF [118]	ECG	45	72.6	56.4	81.3	59.8	0.46

Tabla 4.4: Desempeño y comparación con la literatura al clasificar en W, REM y no REM. Para el caso de Yücelbaş se utilizaron dos bases de datos diferentes. Se presentan los resultados reportados en ese artículo, donde P1 y H1 corresponden a la base de datos 1 para sujetos con patologías (*patients*) y sanos (*healthy*). Lo mismo sucede para la segunda base de datos, con P2 y H2.

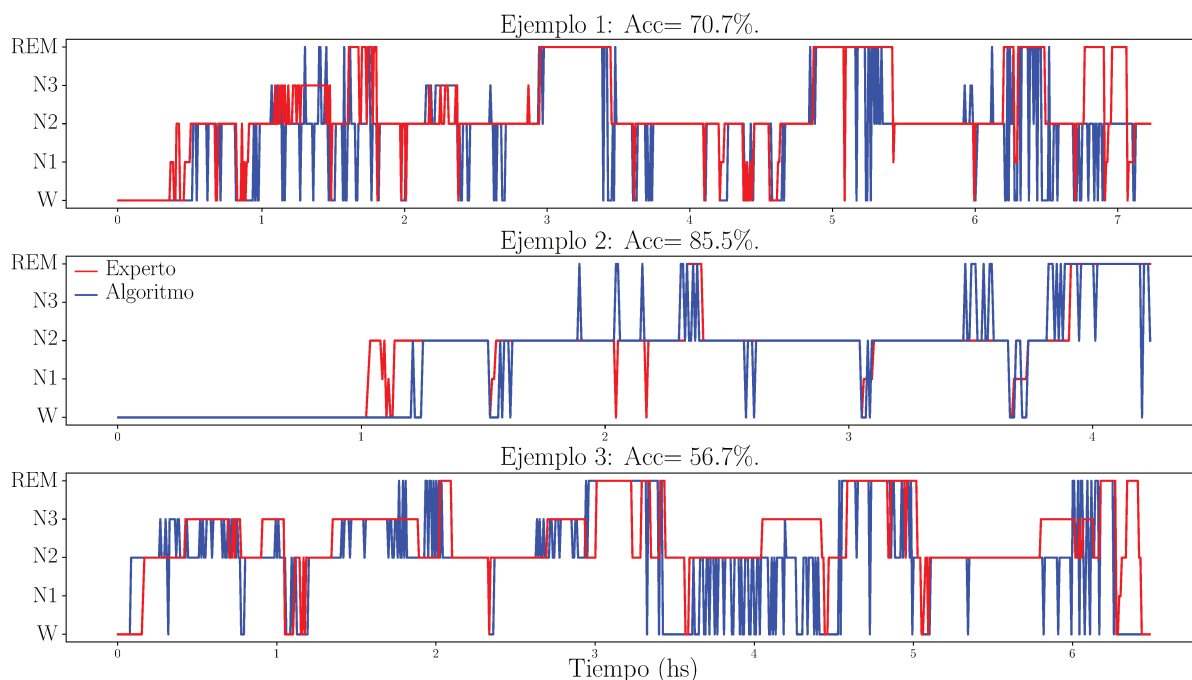


Figura 4.5: Hipnogramas obtenidos mediante clasificación con la red neuronal «óptima» desarrollada. En la parte superior de la figura se muestran los hipnogramas de referencia (etiquetado por un conjunto de expertos) y la clasificación realizada por el algoritmo en un sujeto para el que se obtuvo un desempeño cercano al promedio. En el medio se muestra otro ejemplo, esta vez con un desempeño superior al promedio. Abajo, en cambio, se muestra un tercer ejemplo para el cual el desempeño fue inferior al promedio. En líneas rojas se presenta el etiquetado realizado por expertos, mientras que en líneas azules se presenta el etiquetado realizado por el algoritmo.

al incrementar la longitud de los segmentos. Sin embargo, esto repercutía en forma directa sobre la aplicabilidad del método en una situación real, donde la longitud requerida era 30 segundos. En el desarrollo propuesto en este capítulo se logró abordar este problema al incorporar una relación secuencial entre las clasificaciones, donde las salidas actuales son función de las salidas previas y futuras. Gracias a esta suerte de «memoria» con la que cuentan las RNN, se logró mejorar significativamente el desempeño respecto al algoritmo anterior. Tal es así, que se han superado los desempeños obtenidos en el estado del arte con señales mucho más informativas, con excepción de aquellas que utilizan EEG. Este resultado además fue obtenido mediante arquitecturas relativamente simples.

### 4.3.1. Incorporación de SpO<sub>2</sub>

Al observar los resultados obtenidos, puede verse que la incorporación de la SpO<sub>2</sub> como entrada no genera demasiadas variaciones en el desempeño respecto a utilizar únicamente la HR. Sin embargo, produce un ligero incremento en el desempeño. Esta señal puede ser útil para desambiguar algunas situaciones confusas. Por ejemplo, las recuperaciones rápidas en la saturación de oxígeno en sangre luego de un evento de apnea/hipopnea usualmente están asociadas con eventos de despertares. Asimismo, los decaimientos en la señal de SpO<sub>2</sub> asociados con los eventos de apnea únicamente pueden aparecer durante el sueño. Sin embargo, aunque estadísticamente significativa según los experimentos, la variación en el desempeño al incorporar SpO<sub>2</sub> no es muy grande.

Además, hay que considerar que al duplicar la cantidad de entradas se incrementa mucho la cantidad de parámetros de la red. Esto, junto con la longitud de las secuencias, puede complicar las restricciones de memoria para entrenar el algoritmo en las GPU. En los experimentos realizados aquí se utilizó la señal de forma completa –sin segmentar– con el objetivo de que se tenga en cuenta el historial de toda la señal para cada una de las clasificaciones realizadas. Esto, si bien teóricamente es cierto, en la práctica no es del todo correcto ya que en general la información se desvanece luego de un cierto número de elementos de la secuencia. En caso de complejizar la red –incrementando la cantidad de neuronas o la cantidad de capas, o bien incorporando algún nuevo bloque– probablemente sería necesario trabajar con una estrategia de segmentado de las secuencias para superar las restricciones de memoria. La segmentación de las secuencias también es una práctica común para poder incrementar el tamaño de los *mini-batches* y, consecuentemente, la velocidad de entrenamiento.

### 4.3.2. Comparación con estado del arte

Al igual que en la discusión anterior, incorporaremos el desempeño obtenido mediante este algoritmo a nuestra comparación con el estado del arte. Los algoritmos presentados en el capítulo anterior fueron unos de los pocos en clasificar las fases del sueño en forma simplificada utilizando oximetría de pulso. De hecho, al momento de ser publicado, el único trabajo existente que utilizaba únicamente señales de oximetría de pulso era el de Uçar y col. [37], siendo los resultados de ambos trabajos relativamente similares.

Sin embargo, la comparación resultaba desfavorable para nuestro trabajo al considerar el desempeño obtenido por aquellos que utilizaban señales de ECG, señal que se supone que porta información «equivalente», aunque posiblemente más exacta dada la mejor calidad de la frecuencia cardíaca instantánea. Por ejemplo, los trabajos de Adnane y col. [34] y de Malik y col. [36] tenían un desempeño considerablemente superior. Esta diferencia se incrementaba aún más al comparar con algoritmos que consideraban –además de la señal de ECG– información de señales de acelerómetros, como el de Beattie y col. [38] o el de Fonseca y col. [66].

El nuevo algoritmo aquí presentado, en cambio, supera con creces los resultados obtenidos por trabajos que hacen uso de señales cuya información es equivalente –oximetría de pulso, PPG–, llegando incluso a competir con aquellos que utilizan señales más informativas. Esta notoria mejora de desempeño hace posible pensar en la utilización de este desarrollo en una aplicación real sin demasiadas modificaciones. Más aún si recordamos los inconvenientes que existen en la tasa de acuerdos interprofesional e intraprofesional comentados en el capítulo 2. Volveremos sobre esto más adelante, al presentar y discutir los resultados obtenidos modificando el algoritmo para clasificar todas las fases del sueño según la AASM.

Finalmente, vemos que, a pesar de la importante mejora respecto a nuestro trabajo anterior –y respecto al estado del arte en general–, los algoritmos basados en oximetría de pulso todavía están bastante lejos de poder competir con aquellos que utilizan EEG.



Como realizamos en todos los capítulos en los que se presentan desarrollos, en la tabla 4.5 repetimos los resultados obtenidos por los principales trabajos del estado del arte, incluyendo ahora el desempeño obtenido del nuevo algoritmo. Presentamos únicamente los mejores resultados utilizando HR y HR junto con SpO<sub>2</sub>.

Método	Señales	C	N	Acc	Se	Sp	PPV	NPV	$\kappa$
CWT feat.+RF [30]	EEG	5	16	93.4	95.9	89.6	93.3	93.5	0.86
CNN + RNN [28]	EEG	5	20	94.3	96.8	83.4	86.0	96.2	0.81
CNN + RNN [65]	EEG	5	61	95.7	96.9	90.6	87.8	97.7	0.86
Feat.+SVM+RFE [34]	ECG	2	18	80.0	84.5	69.1	87.0	64.5	0.52
CNN [36]	ECG (CGMH-val)	2	27	83.1	89.4	52.4	90.1	50.5	0.41
CNN [36]	ECG (DREAMS)	2	20	81.4	87.1	53.1	90.2	45.2	0.38
CNN [36]	ECG (UCDSADB)	2	25	73.7	81.9	43.4	84.3	39.2	0.24
2 GRU-(128) [119]	HR	2	5000	89.1	94.2	76.8	90.7	83.6	0.71
2 GRU-(256) [119]	HR+SpO <sub>2</sub>	2	5000	90.1	94.1	80.3	92.1	84.7	0.74
Feat. + FFS-SVM [64]	HR	2	5000	73.7	80.9	54.6	48.6	84.7	0.44
Feat.+SVM [37]	HRV	2	10	73.1	74.0	72.0	–	–	0.46
Feat.+SVM [37]	PPG	2	10	76.8	76.0	77.0	–	–	0.53
Feat.+kNN [37]	PPG+HRV	2	10	79.4	77.0	81.0	–	–	0.59
CNN [36]	PPG (CGMH-val)	2	27	84.2	90.9	51.5	90.1	53.6	0.43
Feat.+RNN [66]	ECG+acc	5	778	90.1	94.0	72.9	–	75.1	0.65
Feat.+LDA [38]	PPG+acc	4	60	90.6	94.6	69.3	94.3	70.5	0.64

Tabla 4.5: Comparación con otros trabajos que realizan clasificación automática de las fases del sueño. C representa la cantidad de clases en las que se realiza la clasificación, 2 corresponde a despierto y dormido, 4 a despierto, sueño liviano, sueño profundo y REM y 5 a la clasificación según AASM. N es la cantidad de sujetos en la base de datos. Se presentan en gris más oscuro los desempeños obtenidos por el trabajo presentado en este capítulo y en gris más claro el desarrollos presentado en el capítulo anterior.

Nuevamente, es necesario hacer notar el tamaño de la base de datos utilizada en relación con los demás trabajos. Esto hace que el riesgo de sobreajuste de los algoritmos desarrollados sea muy bajo. Como contrapartida, su entrenamiento consumió una cantidad de tiempo muy significativa [109].

Finalmente, queremos resaltar que el algoritmo desarrollado por Malik y col. también utiliza técnicas relacionadas con las redes neuronales profundas para clasificar en despierto/dormido, aunque con señales provenientes de ECG. En ese trabajo consideraron diferentes bases de datos a partir de las cuales se obtuvieron las señales de HR con las que se entrenaron y validaron los algoritmos, obteniendo desempeños similares en todas ellas. Además, se realizaron experimentos donde la señal de HR es obtenida a partir de PPG, con buenos resultados. La arquitectura implementada en ese trabajo consiste de cinco bloques convolucionales, cada uno de ellos compuestos por dos redes convolucionales. Estos bloques son los encargados de extraer características de las entradas. Luego, estas características son clasificadas con un perceptrón multicapa. En total, considerando tanto los bloques convolucionales como el perceptrón multicapa, esa arquitectura está compuesta por un total de 12 capas. Vemos que en comparación con ese trabajo, nosotros hemos obtenido un desempeño superior con una arquitectura más simple basada en GRU.

## 4.4. Conclusiones

En este capítulo se diseñó un modelo basado en GRU bidireccionales para clasificar las fases del sueño en despierto y dormido a partir de señales de HR y SpO<sub>2</sub>. Ha quedado demostrado que mediante arquitecturas relativamente simples se pueden alcanzar buenos resultados. Además, si se somete a la señal de HR a un procesamiento adecuado se pueden lograr resultados sumamente útiles para cribado de las fases del sueño. Hasta donde llega nuestro conocimiento, la red propuesta supera el estado del arte de algoritmos que utilizan señales mucho más difíciles de adquirir, además de más informativas. La única –y obvia– excepción son los algoritmos que utilizan EEG.

Aquí abordamos una limitación que poseen todos los algoritmos de cribado de apnea del sueño basados en desaturación de oxígeno en sangre. Además, la red neuronal diseñada puede ser aplicada fácilmente a otras aplicaciones como monitoreo de somnolencia en conductores, dispositivos «vestibles» para monitoreo personal de la salud, entre otros. Al igual que en el capítulo anterior, la base de datos utilizada aquí es mucho mayor que las que se utilizan en otros trabajos relacionados, lo que conlleva a un riesgo de sobreajuste mínimo.

Por otro lado, en este capítulo también se presentó una modificación del algoritmo de forma que discrimine con mayor detalle las fases del sueño. El modelo demostró ser capaz de aprender las reglas de transición entre las fases del sueño. Nuevamente el algoritmo desarrollado superó el estado del arte –exceptuando los algoritmos con EEG–. Sin embargo, esta vez la diferencia de desempeño con los demás algoritmos es significativamente alta, superando en aproximadamente un 7% a los otros algoritmos que utilizan no sólo la señal de HR, sino también incorporan información de acelerómetros. Además, esos algoritmos funcionan para sujetos sanos, cuando la base de datos utilizada aquí contiene tanto sujetos sanos como sujetos con patologías. Sin embargo, este algoritmo debe ser mejorado para poder resolver mejor el problema de desbalance de clases, ya que tiene una tasa de error bastante alta para detectar la fase N2, y es completamente incapaz de detectar la fase N1. A pesar de esto, el desempeño general alcanzado tiene un grado de acuerdo equivalente al grado de acuerdo interprofesional obtenido por médicos expertos, por lo que este algoritmo tiene un enorme potencial en el cribado automático de las fases del sueño.

El algoritmo desarrollado en este capítulo dio lugar a las siguientes publicaciones en congresos y revistas científicas indexadas:

- Casal, Ramiro, Leandro E. Di Persia, y Gastón Schlotthauer. “Sleep/wake classification with pulse oximeter signals using recurrent neural networks.” *LatinX in AI Research at ICML 2019*, 2019, Póster.
- Casal, Ramiro, Leandro E. Di Persia, y Gastón Schlotthauer. “Classifying sleep–wake stages through recurrent neural networks using pulse oximetry signals.” *Biomedical Signal Processing and Control* 63 (2020): 102195.

## Capítulo 5

# Desarrollo de un algoritmo basado en redes convolucionales temporales y mecanismos de atención

En este capítulo presentaremos el tercer y último algoritmo desarrollado para abordar el problema de la clasificación automática del sueño con el objetivo de superar las desventajas del etiquetado manual, de permitir evaluar la prevalencia de los trastornos del sueño en grandes poblaciones evitando la saturación del sistema de salud y de combatir el subdiagnóstico tan presente en estas patologías. En el capítulo 3 se presentó un algoritmo en el que se extrajeron características en forma manual de las señales de HR. Estas características fueron diseñadas con el objetivo de reconocer los patrones característicos correspondientes a las etapas W y S, utilizando medidas de entropía y complejidad, análisis en el dominio temporal y en el dominio tiempo-escala y estadísticos clásicos. Con estas características se entrenaron varios tipos de clasificadores para determinar las fases del sueño.

Vimos en el capítulo 4 que, con el surgimiento del aprendizaje profundo, el diseño de características en forma manual ha sido gradualmente reemplazado por aprendizaje automático de características a partir de los datos de entrada. Las arquitecturas más populares para la extracción automática de características son las CNN, en las cuales se aprenden filtros que luego son aplicados sobre las entradas para obtener características invariantes en el tiempo. Múltiples capas –compuestas principalmente por estos filtros, funciones de activación no lineales y operaciones de agrupado (*pooling*)– se aplican en forma secuencial para aprender representaciones de los datos con diferentes niveles de abstracción [105]. Sin embargo, para tareas que involucran series temporales como entradas –como en la clasificación automática del sueño–, la opción por defecto han sido históricamente las RNN, gracias a su capacidad para almacenar información de todos los elementos pasados de la secuencia, aprendiendo las dependencias temporales de la serie [105]. Teniendo este aspecto en mente, también en el capítulo 4 presentamos el desarrollo de un algoritmo basado en RNN para clasificar las fases del sueño. Si bien ese algoritmo superó ampliamente los resultados previos –tanto propios como ajenos–, presentaba una serie de desventajas asociadas con las RNN. La naturaleza secuencial de estas herramientas impide su paralelización, lo cual conduce a un proceso de aprendizaje muy lento, generando importantes restricciones en la memoria [120]. Además, el proceso de entrenamiento era altamente inestable.

En este capítulo diseñaremos una arquitectura basada en mecanismos de atención para superar las desventajas de las RNN, obteniendo entrenamientos muchísimo más rápidos sin pérdida significativa de desempeño.

Las redes de atención también permiten modelizar las dependencias temporales entre muestras de las secuencias de entrada y salida, aunque –a diferencia de las RNN– sin de-

pendencia alguna de la distancia entre las muestras. Se suele decir que a diferencia de las RNN, las redes de atención aprenden la dinámica de las series temporales a través de, como su nombre lo indica, «atención», en lugar de hacerlo mediante «memoria» [120]. Además, tienen la ventaja de ser más «amigables» para las GPUs que las RNN.

La mayoría de estas arquitecturas han sido diseñadas para aplicaciones de procesamiento del lenguaje natural. En ese área es habitual realizar un preprocesamiento denominado *token embedding*, en el cual se asigna cada palabra a un vector de alta dimensión cuyo valor está relacionado con su significado. Por este motivo, las entradas a las redes de atención están pensadas para ser de alta dimensión. Como este no es el caso de las señales del oxímetro de pulso –1D si únicamente consideramos HR–, en este trabajo se utilizaron redes convolucionales temporales –una arquitectura que combina CNN y redes neuronales residuales (*residual neural networks*, ResNet)– para transformar la secuencia de entrada a una dimensión más alta, a la vez que se reduce su longitud temporal [121]. De esta forma, los datos de entrada pueden ser procesados por las redes de atención utilizando una secuencia cuya longitud es menor que la entrada original y, consecuentemente, más rápida de aprender.

El objetivo del algoritmo presentado es clasificar el estado del sueño en despierto y dormido utilizando la señal de HR provista por el oxímetro de pulso, obteniendo una salida confiable y rápida. En resumen, en este capítulo se presentarán tres alternativas de arquitecturas para la obtención de una representación –una basada en perceptrón de capa única (*single-layer perceptron*, SLP), otra basada en CNN y la última basada en redes convolucionales temporales (*temporal convolutional networks*, TCN)–. Luego, la salida de esta representación servirá de alimentación para una red basada en mecanismos de atención, donde se clasificarán las señales de HR sin utilizar RNN o características diseñadas y extraídas en forma manual y sin pérdida de desempeño en comparación con el estado del arte. Para su diseño y evaluación, se utilizó una base de datos basada únicamente en HR obtenido a partir de oximetría de pulso para clasificar las fases del sueño, alcanzando un resultado que es comparable con algoritmos que utilizan señales más informativas y confiables. Estos resultados permitirían un cribado de las fases del sueño.

## 5.1. Metodología

En este capítulo se propone una arquitectura para clasificar las fases del sueño en forma simplificada a partir de señales de HR obtenidas por un oxímetro de pulso. En esta sección describiremos la arquitectura propuesta, analizando cada una de las herramientas utilizadas. Al final de la sección presentaremos algunas cuestiones relativas a su implementación.

Para diseñar, entrenar y evaluar este algoritmo se siguió el mismo esquema presentado en el capítulo anterior. Es decir, 2500 sujetos para el entrenamiento de la red, 1250 sujetos para validación y 1250 sujetos para testeo de la red entrenada. Además, se utilizó únicamente la base de datos SHHS 1, descartando la SHHS 2 para evitar la presencia de sujetos repetidos en los datos.

### 5.1.1. Arquitectura propuesta

La arquitectura comprende dos partes fundamentales. La primera de ellas se diseñó para obtener una representación de la señal de HR que luego será utilizada como entrada de la segunda parte. Esta última –basada principalmente en *transformers*, un modelo construido únicamente con mecanismos de atención– se desarrolló para aprender las reglas de transición entre las fases del sueño en la secuencia de entrada. La arquitectura, en su totalidad, recibe como entradas las señales de HR con un preprocesamiento muy simple y produce como salidas las clasificaciones en despierto y dormido cada 30 segundos.

## Aprendizaje de la representación

Para lograr aprender una representación, que luego será utilizada para modelar la secuencia por el siguiente bloque del algoritmo, se evaluaron varias alternativas diferentes. Esta parte de la red fue diseñada con dos objetivos diferentes. El primer objetivo es aprender una representación en alta dimensión de la entrada con la cual alimentar la siguiente parte de la red. El segundo objetivo es reducir la longitud de la secuencia a través del tiempo, en beneficio del costo computacional. Este bloque fue pensado para funcionar como un extractor automático de características de las señales de HR.

La primera de las propuestas realizadas fue utilizar una de las redes neuronales más simples y clásicas: el SLP. Para ello se planteo la arquitectura más simple posible, un perceptrón de una única capa que recibe como entradas 30 muestras consecutivas de la secuencia y las transforma a la dimensión  $d_{\text{model}}$ , teniendo como función de activación a una tangente hiperbólica.

La segunda de las alternativas fue utilizar una de las arquitecturas más utilizadas con el propósito de extracción de características: las CNN. En este sentido, se diseñaron varias arquitecturas posibles basadas en CNN, cuyas capas realizan en forma secuencial las siguientes operaciones: convolución 1D, *dropout*, normalización por lotes y aplicación de una función de activación ReLU. Para esta alternativa se modificó el número de capas consideradas, los tamaños de filtro y los parámetros del submuestreo de la convolución de manera tal de alcanzar los dos objetivos propuestos con el mejor desempeño posible.

La tercera de las alternativas es la más novedosa y está basada en TCN. Las TCN son una familia de arquitecturas basadas en CNN que emergieron con el objetivo de comparar los desempeños de las RNN y las CNN para modelización de secuencias, cuestionando la consabida máxima de que las RNN son la opción por defecto para estos casos. Las TCN fueron propuestas por Bai y col. [121] utilizando 1D-CNN, en las que cada capa transforma las entradas en una salida de igual longitud. Para ello, completa con ceros en ambos extremos de la señal para que el filtro se pueda aplicar de manera tal que su salida tenga la misma longitud. Las convoluciones consideradas son causales, lo cual significa que la salida para un tiempo dado depende sólo de las entradas previas. Además, los autores emplearon convoluciones dilatadas que permiten incrementar el campo receptivo exponencialmente. Formalmente, para una secuencia de entrada  $\mathbf{x} = [x[0], x[1], \dots, x[N]]$  y un filtro  $\mathbf{h} = [h[0], h[1], \dots, h[k-1]]$ , siendo  $k$  el tamaño del filtro, la operación de convolución dilatada sobre el elemento  $n$  de la secuencia está definida por:

$$y[n] = x[n] *_d h[n] = \sum_{i=0}^{k-1} h[i]x[n - di], \quad (5.1)$$

donde  $d$  es el factor de dilatación. En forma sencilla, puede describirse como una convolución normal, pero submuestreando la secuencia de entrada en un factor  $d$ . En Dumoulin y cols. puede encontrarse una referencia completa de las convoluciones y el aprendizaje profundo [122]. El campo receptivo, entonces, depende de la dilatación de las convoluciones, del tamaño del filtro  $k$  en las CNN y de la profundidad de la red [121].

Las TCN están construidas a partir de unos bloques básicos –llamados bloques residuales (*residual blocks*)– que consisten en dos capas de convoluciones dilatadas con una ReLU como función de activación no lineal [123]. Cada bloque residual tiene una conexión residual [124] que le permite aprender únicamente la modificación de una transformación identidad –es decir, el residuo– en lugar de la salida completa. Este simple artificio simplifica la convergencia en redes profundas, ya que en caso de que haya capas «sobrantes», estas pueden converger a la identidad de manera muy simple debido a la realimentación. En caso de que las dimensiones coincidan, estos «atajos» de la identidad simplemente se suman con la

salida de la red. Sin embargo, cuando la red realiza alguna transformación en la dimensión de los datos, es necesario adaptar la entrada al bloque de forma que pueda ser sumada a la salida. He y cols. han planteado dos opciones para superar este inconveniente. La primera de ellas considera adaptar las dimensiones simplemente añadiendo ceros para incrementar la dimensión –o submuestreando para reducirla–. La segunda de ellas es lograr la coincidencia de las dimensiones aplicando una transformación lineal a la entrada, por ejemplo mediante una red convolucional. En este trabajo utilizamos la segunda opción. Todo lo relativo a los bloques residuales está extensamente explicado en He y cols. [124].

Para alcanzar los dos objetivos mencionados para este bloque se introdujeron algunas modificaciones en las TCN. En primer lugar, no se consideraron convoluciones causales, ya que en esta aplicación no es necesario obtener un procesamiento en línea. Tal es así que en el algoritmo anterior se consideraron GRUs bidireccionales para poder explotar la información pasada y futura [119]. En segundo lugar, se incluyó en la operación de convolución un submuestreo de la señal (*stride operation*) para lograr el segundo objetivo mencionado –trabajos recientes sugieren que las operaciones de agrupamiento (*pooling*) pueden ser reemplazada por submuestreo sin una degradación del desempeño [124, 125]–. Por último, se aplicó normalización por lotes (*batch normalization*) para normalizar las entradas en cada capa [126] y se utilizó el método de *dropout* como regularización [127].

En la figura 5.1 se muestra un esquema correspondiente a esta parte de la red. En nuestro modelo, cada bloque residual consiste en dos subcapas convolucionales, las cuales realizan cuatro operaciones en forma secuencial: convolución dilatada 1D –con filtros de tamaño  $k$ , cantidad de canales de salida (o número de filtros)  $c$ , parámetro de dilatación  $d$  y submuestreo  $str$ –, normalización por lotes, ReLU y *dropout*. La reducción de dimensión es realizada únicamente en la segunda subcapa por medio del submuestreo, mientras que en la primera subcapa convolucional se fija  $str = 1$ .

Esta parte de la red consiste en cuatro bloques residuales, cuyos parámetros pueden observarse en la figura 5.1. Como puede verse, cada uno de los bloques residuales incrementa el número de canales –es decir, la dimensión– en forma gradual desde 1 hasta  $d_{\text{model}}$ , además de reducir la longitud de la secuencia mediante submuestreo desde  $L$  hasta  $L/30$ . Este submuestreo se realiza para condensar en una única muestra los datos que se utilizarán para clasificar un segmento de 30 segundos. Sin embargo, no está de más aclarar que el resultado de esta reducción de dimensiones a lo largo del tiempo no contiene únicamente información correspondiente a un segmento de 30 segundos, sino que, dado el amplio campo receptivo de la TCN, incluye además información de varios segmentos adyacentes que confluyen en un único dato temporal que luego será procesado por los *transformers*.

Las salidas del cuarto bloque, entonces, son las características invariantes en el tiempo y de alta dimensión que han sido extraídas a partir de la señal de HR.

## Modelización de la secuencia

Para la segunda parte de la red, se aplicó una estructura *encoder-decoder* para realizar la modelización de la secuencia. Para ello se utilizó un tipo de arquitectura llamado *transformer*, el cual sólo utiliza mecanismos de atención [120]. El *encoder* transforma la secuencia de entrada en una secuencia de representación, la cual es luego utilizada como entrada al *decoder*, para que éste genere la secuencia de salida de a un elemento por vez. En nuestro modelo, el *encoder* se utilizó tal y como lo propuso Vaswani y cols. [120]. El *decoder*, en cambio, se simplificó a un perceptrón multicapa para interpretar y clasificar la secuencia «codificada». Tanto el *encoder* como el *decoder* serán explicados en el siguiente apartado.

Como ya se dijo, la mayoría de los modelos de traducción de secuencias han sido diseñados para procesamiento del lenguaje natural y es por ésto que utilizan una transformación que

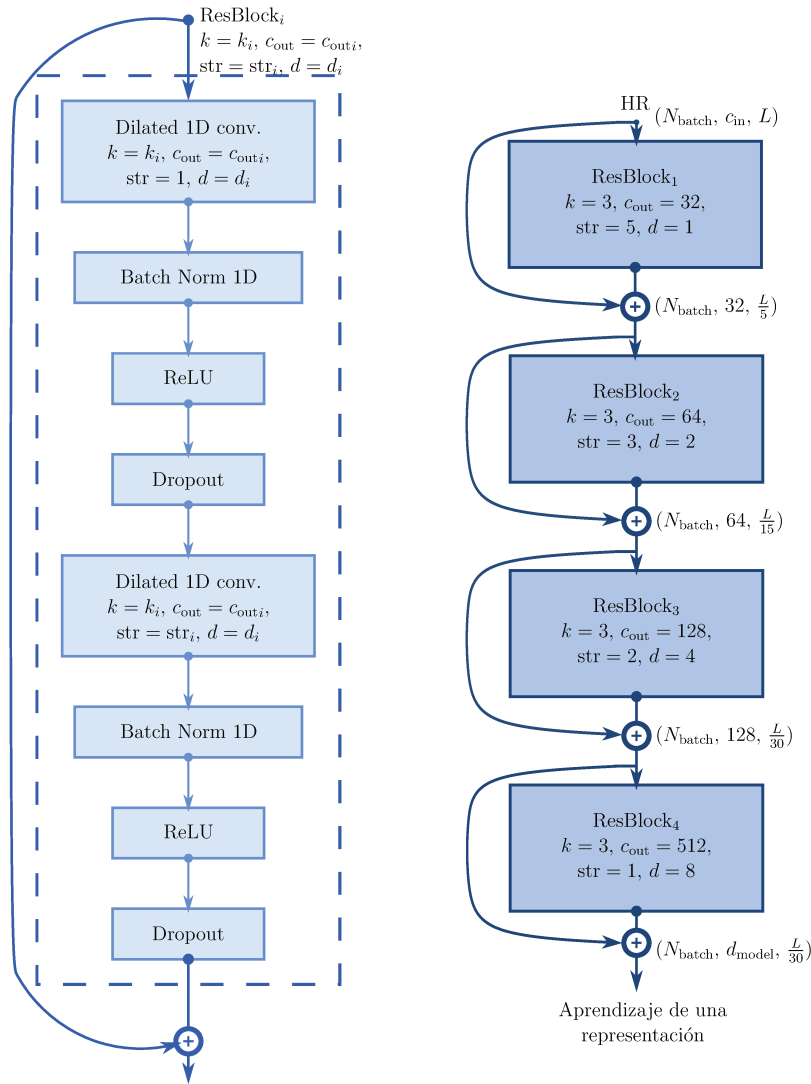


Figura 5.1: Esquema de la arquitectura de la parte de aprendizaje de la representación en la red. A la izquierda se encuentra la composición de un bloque residual y sus parámetros. A la derecha se encuentra la red diseñada, la cual consiste en cuatro bloques residuales. Luego de cada capa se muestra la transformación de dimensiones realizada.

convierte las palabras en vectores de dimensión  $d_{\text{model}}$ . A esta transformación se conoce en inglés como *embedding*. Para adaptar esta transformación a nuestro problema, se utilizó la arquitectura basada en TCNs descrita anteriormente para aprender una representación cuya dimensión es  $d_{\text{model}}$  –éste era el primer objetivo de esta parte de la red–. A continuación describiremos todos los elementos de la parte de la red encargada de la modelización de la secuencia. Para ello seguiremos la misma organización que utilizaron Vaswani y col. [120]. Un esquema de esta parte de la red se muestra en la figura 5.2.

**Encoder.** El *encoder* está compuesto por dos capas apiladas, las cuales a su vez contienen dos subcapas. La primera subcapa consiste de un mecanismo de atenciones múltiples (*multi-head attention mechanism*), mientras que la segunda es un perceptrón multicapa. Ambas subcapas tienen conexiones residuales seguidas por una capa de normalización [128].

Para calcular la matriz de autoatención (*self-attention*), las entradas codificadas son agrupadas en una matriz, la cual es proyectada mediante tres matrices entrenables  $\mathbf{W}^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}$ ,  $\mathbf{W}^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$ ,  $\mathbf{W}^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$ . Como resultado de esta proyección se obtienen

tres matrices, las cuales se denominan en inglés *query* ( $\mathbf{Q}$ ), *key* ( $\mathbf{K}$ ) y *value* ( $\mathbf{V}$ ). Cada fila de estas matrices se corresponde con la proyección de cada elemento de la entrada. Luego, la matriz de atención se calcula haciendo:

$$A(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax} \left( \frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}} \right) \mathbf{V} \quad (5.2)$$

donde  $\sqrt{d_k}$  es la dimensión de  $\mathbf{K}$  y  $\frac{1}{\sqrt{d_k}}$  es un factor de escala que lleva los elementos de la matriz resultante a una región más sensible de la función *softmax*, definida como  $\text{softmax}(x_1, \dots, x_n) = \left( \frac{\exp\{x_i\}}{\sum_j \exp\{x_j\}} \right)_i$ , siendo  $x_i$  la componente de un vector  $\mathbf{x}$ . La función *softmax* «normaliza» las filas para que la suma total sea 1. De esta forma, la  $i$ -ésima fila de la matriz resultante de la operación  $\text{softmax} \left( \frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}} \right)$  contendrá un coeficiente entre 0 y 1 que representa el «peso» que tiene cada elemento de la secuencia de entrada en el  $i$ -ésimo elemento de la entrada. Luego, al multiplicarse este coeficiente por  $\mathbf{V}$  se obtiene una nueva matriz  $A(\mathbf{Q}, \mathbf{K}, \mathbf{V})$ , cuyas filas contienen los vectores de autoatención para cada uno de los elementos de entrada.

Se emplearon mecanismos de atenciones múltiples de la misma forma que lo realizado por Vaswani y col. [120], utilizando  $h$  matrices  $\mathbf{W}_i^Q$ ,  $\mathbf{W}_i^K$ ,  $\mathbf{W}_i^V$  diferentes, con  $i$  tomando valores enteros entre 1 y  $h$ , que fueron inicializadas en forma aleatoria para obtener  $h$  proyecciones de las entradas. Esta modificación permite al modelo tener múltiples representaciones del subespacio a partir del cual calcular  $h$  matrices de atención. Luego, estas  $h$  matrices de atención son concatenadas verticalmente y multiplicadas por otra matriz –también entrenable–  $\mathbf{W}^0 \in \mathbb{R}^{hd_v \times d_{\text{modelo}}}$  para producir la salida del bloque de atenciones múltiples, el cual es finalmente utilizado como entrada de la segunda subcapa –es decir, del perceptrón simple–.

En este trabajo se empleó  $h = 8$ , reduciendo la dimensión a  $d_k = d_{\text{modelo}}/h$  con la intención de mantener relativamente constante el costo computacional.

El perceptrón multicapa en la segunda subcapa está compuesto por dos transformaciones lineales con una función de activación ReLU. Formalmente, se calcula haciendo

$$\text{output encoder}_i = \mathbf{W}_2^{e_i} \text{ReLU}(\mathbf{W}_1^{e_i} x + b_1^{e_i}) + b_2^{e_i}. \quad (5.3)$$

donde  $e_i$  representa al  $i$ -ésimo *encoder*. La dimensión tanto de las entradas y las salidas del SLP es  $d_{\text{modelo}}$ , mientras que la dimensión de las capas internas es  $d_{\text{fn}} = 512$ .

La codificación de la posición (*positional encoding*) es el último elemento que debemos abordar para entender la arquitectura del *encoder*. Esta operación es un simple artificio para añadir información de la posición al vector de entrada a los mecanismos de atención y se realiza únicamente en la entrada al primer *encoder*. Para ello, hacemos [120]:

$$\begin{aligned} \text{PE}(p, 2i) &= \text{sen}(p/10000^{2i/d_{\text{modelo}}}), \\ \text{PE}(p, 2i + 1) &= \text{cos}(p/10000^{2i/d_{\text{modelo}}}) \end{aligned} \quad (5.4)$$

donde  $p = 0, 2, \dots, L_{\text{máx}} - 1$  es la posición del elemento en la secuencia, mientras que  $i = 0, 2, \dots, d_{\text{modelo}} - 1$  son las coordenadas del vector.  $L_{\text{máx}}$  es la máxima longitud de secuencia posible. La codificación posicional permite al modelo aprender patrones específicos que contienen información relativa a la posición del  $p$ -ésimo elemento de la secuencia total, representando el orden de la secuencia. Para poder luego ser sumados, tanto la entrada al *encoder* como la codificación de la posición deben tener la misma dimensión. Luego, la salida del TCN –vector de características– se suma con el PE previo a entrar en el primer *encoder*, tal y cómo se ilustra en la figura 5.2.

**Decoder.** El *encoder* fue diseñado para aprender las reglas de transición en la secuencia. Esta información es luego clasificada por el *decoder*, el cual simplemente consiste en



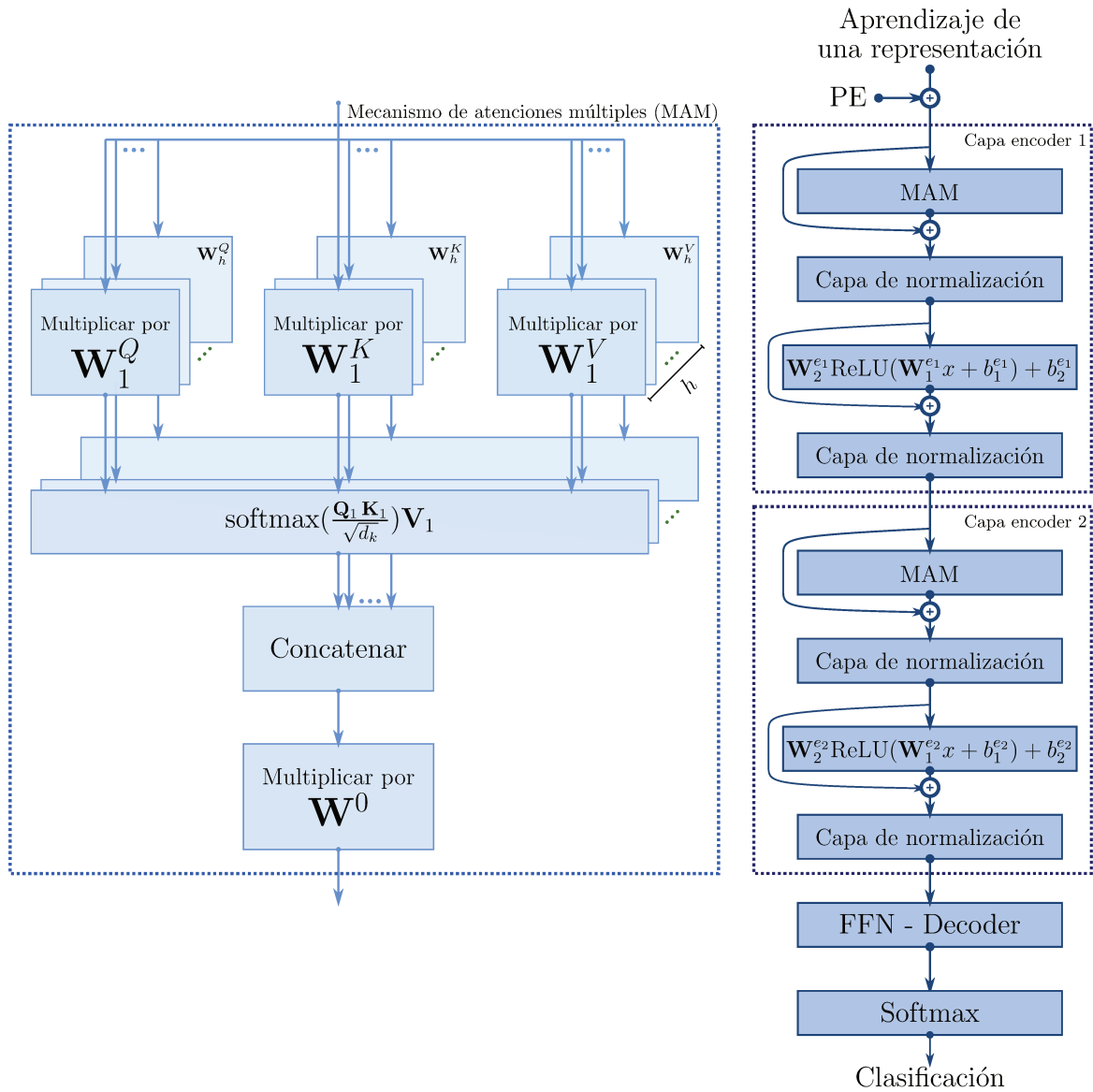


Figura 5.2: Esquema de la arquitectura de la parte de modelización de la secuencia en la red. A la izquierda se encuentra el bloque de atenciones múltiples. A la derecha se describe la red, la cual consiste en dos capas de *encoders* apilados y una capa de *decoder*.

un perceptrón multicapa compuesto por dos transformaciones lineales con una función de activación ReLU entre ambas, calculado haciendo

$$\text{output decoder} = \mathbf{W}_2^d \text{ReLU}(\mathbf{W}_1^d x + b_1^d) + b_2^d. \quad (5.5)$$

donde el supraíndice  $d$  es para representar que los pesos corresponden al *decoder*. La dimensión de entrada y salida de la primera capa es  $d_{\text{model}}$ . En la segunda capa la dimensión de salida es reducida al número de etapas del sueño consideradas, que en este caso es 2. Finalmente, se utiliza una capa *softmax* para realizar la clasificación.

## 5.2. Resultados

En esta sección se describirán todos los detalles de la implementación de las arquitecturas, los experimentos realizados para evaluar su desempeño y un conjunto de arquitecturas desarrolladas con propósitos comparativos.

### 5.2.1. Detalles de implementación

Se mencionarán a continuación las configuraciones de los experimentos y los parámetros de entrenamiento necesarios para repetir el desarrollo realizado.

#### *Hardware y software utilizados*

El modelo se implementó utilizando Python 3 y la librería de aprendizaje maquina PyTorch. Para entrenar las redes se utilizaron varias GPUs instaladas en un *cluster* de alto desempeño equipado con tres GPU NVIDIA GTX 1080Ti, además de una computadora personal equipada con una GPU NVIDIA Titan Xp.

#### Preprocesamiento y agrupación de los datos para entrenamiento

Como mencionamos, se dividió aleatoriamente a la base de datos SHHS 1 en tres subconjuntos independientes para entrenamiento, validación y testeo, compuestos por 2500, 1250 y 1250 sujetos diferentes, respectivamente. Todas las señales correspondientes a estos sujetos fueron preprocesadas antes de ser utilizadas en la arquitectura propuesta. En primer lugar, y como en todos los casos anteriores, se identificaron los segmentos defectuosos de la señal de HR utilizando la señal complementaria de calidad provista por el oxímetro de pulso. Una vez identificados, los datos no fidedignos se reemplazaron realizando una interpolación lineal entre los datos confiables previo y posterior. En segundo lugar, se estandarizaron las señales de HR para tener media cero y varianza unitaria. Para esto, se extrajeron la media y la desviación estándar global de la base de datos de entrenamiento. Estos valores se utilizaron luego para estandarizar los tres conjuntos de datos para entrenamiento, validación y testeo.

En la mayoría de los experimentos que involucran aprendizaje profundo es muy común agrupar los datos en lotes –habitualmente conocidos como *minibatches*–, los cuales son utilizados para calcular el error y actualizar los coeficientes del modelo por el optimizador. De esta forma, los datos de entrada son agrupados en un *minibatch*, donde se completan con cero las señales para tener la misma longitud que la secuencia más larga del *minibatch*. Para evitar un completado con ceros excesivo, las entradas fueron previamente ordenadas por longitud. De esta manera, los datos que pertenecen a un mismo *minibatch* tienen longitudes similares. Esto ya fue mencionado en el capítulo anterior y está ilustrado en la figura 4.2. Para el tamaño del *minibatch*, se varió  $N_{\text{batch}}$  desde 2 hasta 32, en pasos de  $2^n$ . El mejor desempeño se obtuvo con un tamaño de *minibatch* igual a 16.

#### Optimizador y regularización

El modelo se entrenó utilizando el optimizador Adam, que es una extensión del descenso por gradiente estocástico [116]. Los parámetros en este optimizador son la tasa de aprendizaje  $lr$  y los exponentes de la tasa de decaimiento de las estimaciones del primer y segundo momento,  $\beta_1$  y  $\beta_2$ . La tasa de aprendizaje  $\alpha$  se varió desde  $10^{-2}$  hasta  $10^{-6}$ , obteniéndose el mejor desempeño para el valor de  $10^{-4}$ . Para los parámetros  $\beta_1$  y  $\beta_2$ , se evaluaron varios valores cercanos a los establecidos por defecto. Finalmente, se estableció  $\beta_1 = 0.9$  y  $\beta_2 = 0.99$ .

Además, se utilizó una regularización  $L_2$ , la cual modifica la función de costo añadiendo un término de penalización para evitar el incremento de la magnitud de los coeficientes del modelo. El coeficiente que multiplica al término de penalización  $L_2$  se estableció igual a  $10^{-3}$ , el cual se obtuvo luego de variar los valores desde  $10^{-3}$  hasta  $10^{-5}$ .

Finalmente se utilizó la técnica de *dropout*, la cual es una técnica de regularización que hace ceros las salidas de los nodos de la red neuronal en forma aleatoria y con una probabilidad  $p$  [129]. De esta forma, la red es forzada a aprender características más robustas, evitando el sobreajuste. Se varió la probabilidad  $p$  desde 0 (sin *dropout*) hasta 0.9 en pasos de 0.1.

El mejor desempeño se obtuvo con los valores 0.1 y 0.8. Como el proceso de entrenamiento tuvo una convergencia más estable con el valor de 0.1, se seleccionó este valor.

Durante el entrenamiento, los pesos para todo el modelo son actualizados en forma iterativa buscando obtener el mínimo en la función de «costo focalizado» (*focal loss*) [130]. El desempeño de la red se evaluó al final de cada época de entrenamiento, utilizando el conjunto de validación. El modelo con el que se obtuvo la mejor exactitud en la base de datos de validación se conservó para ser evaluado con la base de datos de testeo. El número máximo de épocas se estableció igual a 500.

Como ya ha sido dicho, dado que los registros de la base de datos SHHS se obtuvieron en sujetos sometidos a una PSG con el objetivo de evaluar la calidad del sueño, en la mayoría de los registros existe un desbalance de clases. Específicamente, la mayoría de los segmentos totales de los registros se corresponden con la clase «dormido». La relación entre el número de segmentos correspondientes a dormido y el número total de segmentos es de 0.74, este dato ya fue reportado en varios momentos en capítulos anteriores, mientras que la relación entre el número de segmentos correspondientes a despierto y el número de segmentos correspondientes a dormido es de 0.35 –este es el dato del desbalance entre ambas clases, es decir, existe aproximadamente 3 segmentos que corresponden a dormido por cada uno correspondiente a despierto–. Para prevenir el sesgo del clasificador hacia la clase mayoritaria hay varios métodos posibles. En este desarrollo, se utilizó una función de costo que aquí hemos traducido como «costo focalizado» [130]. Esta función de costo efectúa una modificación sobre la entropía cruzada de forma tal que los ejemplos más difíciles de clasificar tengan mayor peso durante el entrenamiento. Recordemos la ecuación 4.4 que define a la entropía cruzada para un problema de clasificación binaria. Por conveniencia en la notación, siguiendo a Lin y cols. [130], definimos  $y_t$ :

$$y_t = \begin{cases} y & \text{si } c = 1 \\ 1 - y & \text{en otro caso,} \end{cases} \quad (5.6)$$

en donde  $c \in \{\pm 1\}$  especifica la clase a la que pertenece la muestra, mientras que  $y \in [0, 1]$  es la probabilidad estimada por la capa *softmax*. De esta forma podemos reescribir la entropía cruzada como  $CE(y, c) = CE(y_t) = -\log(y_t)$ . Existe también una variante de la CE que permite compensar el desbalance de clases mediante la introducción de un parámetro  $\alpha \in [0, 1]$  para  $c = 1$  y  $1 - \alpha$  para  $c = -1$ . Siguiendo una notación análoga a la de  $y_t$ , definimos  $\alpha_t$ :

$$\alpha_t = \begin{cases} \alpha & \text{si } c = 1 \\ 1 - \alpha & \text{en otro caso,} \end{cases} \quad (5.7)$$

y escribimos la entropía cruzada  $\alpha$  como:

$$CE(y_t) = -\alpha_t \log(y_t). \quad (5.8)$$

Con todo esto, podemos definir la función de «costo focalizado» como:

$$FL(y_t) = -\alpha_t(1 - y_t)^\gamma \log(y_t) \quad (5.9)$$

donde el factor  $(1 - y_t)^\gamma$  permite modular la función de costo de acuerdo a qué tan fáciles o difíciles de clasificar son los ejemplos, mientras que  $\alpha_t$  balancea la importancia de la función de costo dependiendo de si el ejemplo corresponde a la clase negativa o positiva. No está de más aclarar que, en este contexto, un ejemplo «fácil» es aquel en el cual  $y$  está cercano a 0 o a 1, por lo que una de las clases domina completamente la probabilidad de clasificación. De forma contraria, un ejemplo «difícil» es aquel en el cual las probabilidades están más balanceadas, siendo  $y$  cercano a 0.5.

Para esta función de costo se evaluaron diferentes parámetros. El parámetro  $\gamma$ , encargado de disminuir la contribución de los ejemplos fáciles de clasificar al costo total, se varió desde 0 hasta 3 en pasos de 1. Además, para controlar la importancia relativa de las clases en la función de costo se procedió a variar el parámetro  $\alpha$  en el rango  $[0, 1]$ . Un  $\alpha = 0.5$  corresponde a la misma importancia para ambas características, mientras que a medida que se acerca a 1 o a 0 incrementa la importancia relativa para una de las clases. En nuestro trabajo se obtuvo el mejor desempeño al establecer  $\alpha = 1/3$  –equivalente aproximado al desbalance de clases– y  $\gamma = 1$ . Una descripción completa de la función de «costo focalizado» puede encontrarse en [130].

### Diseño de la arquitectura

Para diseñar la arquitectura se variaron metódicamente tanto los parámetros de la primera parte de la red, encargada del aprendizaje de una representación, como de la segunda parte encargada de modelizar la secuencia. Para el aprendizaje de una representación, las variaciones se realizaron sobre las tres alternativas.

La primera de las alternativas consistía en un SLP que permitía realizar una transformación de dimensiones, tomando 30 muestras consecutivas y transformándolas a una dimensión  $d_{\text{model}}$  y utilizando como función de activación una tangente hiperbólica. Esta simple arquitectura nos permite evaluar la contribución de la parte de la red encargada de modelizar la secuencia. Además, debido a su desempeño, este experimento reveló la necesidad de desarrollar arquitecturas más complejas para permitir una mejor representación de la señal para su posterior clasificación.

Esto nos condujo a desarrollar la segunda alternativa basada en CNNs. Para ella, se modificaron el número de capas consideradas, los tamaños de filtro y los parámetros del submuestreo de la convolución de manera tal de alcanzar los dos objetivos propuestos con el mejor desempeño posible. El mejor desempeño se obtuvo con una arquitectura que contiene cuatro capas con un tamaño de filtro igual a 3 y estableciendo el submuestreo de cada capa en 5, 3, 2 y 1, respectivamente. Al mismo tiempo, se incrementó la dimensión desde 1 hasta  $d_{\text{model}}$  de la misma forma que para la arquitectura basada en TCN.

De manera similar, para el caso de las TCN se variaron todos los parámetros de los bloques residuales –es decir el tamaño del filtro, el número de canales de salida, el submuestreo y la dilatación de las convoluciones–. Además, se evaluó el efecto de utilizar diferentes números de bloques residuales. El mejor desempeño se obtuvo con la arquitectura que se muestra en la figura 5.1.

Por último, dado que para modelar la secuencia se utilizó un diseño basado en *transformer*, el número de arquitecturas posibles para abordar el problema es relativamente limitado, ya que simplemente contiene una repetición de bloques básicos. Los parámetros del *transformer* que variamos fueron el número de mecanismos de atenciones múltiples considerados  $h$ , el número de capas de *encoders* y la dimensión del perceptrón multicapa utilizado en la arquitectura del *transformer*.

#### 5.2.2. Desempeño en el subconjunto de testeo

A continuación presentaremos los resultados obtenidos con la red diseñada y sus alternativas. El desempeño de los modelos se evaluó utilizando las medidas que ya han sido presentadas: exactitud, sensibilidad, especificidad, valor predictivo positivo, valor predictivo negativo y coeficiente Kappa de Cohen. Estas medidas se seleccionaron con el objetivo de presentar métricas que permitan evaluar el sesgo debido al desbalance de clases entre despierto y dormido [75], evaluando el desempeño para cada clase en particular y no únicamente

Método	Entradas	Acc	Se	Sp	PPV	NPV	$\kappa$	TE	$\mathbf{E}_1$	$\mathbf{E}_2$
TCN+ <i>transformer</i>	HR	90.0	94.9	78.1	91.3	85.2	0.73	558.9	0.472	8.93 %
CNN+ <i>transformer</i>	HR	86.5	<b>95.0</b>	65.3	87.3	83.8	0.62	304.4	–	–
SLP+ <i>transformer</i>	HR	84.8	94.5	63.1	85.8	80.7	0.59	<b>25.9</b>	–	–
2 GRU-(128)[119]	HR	89.1	94.2	76.8	90.7	83.6	0.71	8821.7	0.522	9.08 %
2 GRU-(256)[119]	HR+SpO <sub>2</sub>	<b>90.1</b>	94.1	<b>80.3</b>	<b>92.1</b>	<b>84.7</b>	<b>0.74</b>	12376.7	<b>0.456</b>	<b>8.35 %</b>
Feat.+FFS-SVM[64]	HR	73.7	80.9	54.6	84.8	48.6	0.44	–	–	–

Tabla 5.1: Desempeño de las redes desarrolladas en el subconjunto de testeo. Se muestran además los métodos desarrollados en capítulos anteriores. Se reportan las medidas de exactitud (Acc), sensibilidad (Se), especificidad (Sp), valor predictivo positivo (PPV), valor predictivo negativo (NPV), índice kappa de Cohen ( $\kappa$ ) y tiempo de entrenamiento en minutos (TE), en caso de ser comparables. También se calcula el error absoluto medio  $\mathbf{E}_1$  y el error absoluto medio porcentual  $\mathbf{E}_2$ .

un desempeño global, que puede ser engañoso <sup>1</sup>. Al igual que en los algoritmos anteriores, en primer lugar se calcularon todas estas métricas para cada uno de los sujetos de la base de datos y luego se obtuvo el promedio general, que finalmente es la medida reportada. Como siempre, se consideró la clase «despierto» como clase positiva.

El desempeño del método desarrollado y sus alternativas se evaluó utilizando el subconjunto de testeo. En la tabla 5.1 se muestran los resultados obtenidos, comparando las tres familias de algoritmos desarrolladas en esta tesis. Además de las medidas de desempeño, se reportan las señales utilizadas y el tiempo total de entrenamiento, excepto para el primer algoritmo desarrollado basado en SVM. Esto se debe a que para desarrollar dicho algoritmo hay un proceso de pasos a seguir –la búsqueda de parámetros óptimos de las características, la extracción de las características, el proceso de selección y clasificación– que dificulta el cálculo de un único tiempo de entrenamiento. Sin embargo, mencionaremos que los experimentos llevados a cabo para ese desarrollo llevaron varios meses de tiempo de computación, incluso utilizando un *cluster* de alto desempeño. Por este motivo, si se hicieran todos los experimentos en una única computadora –como es el caso del tiempo de entrenamiento reportado para los otros algoritmos–, habría una diferencia de varios órdenes de magnitud.

Puede verse que el mejor desempeño de los algoritmos desarrollados en este capítulo se obtienen a partir de la combinación de las TCN y los *transformer*, seguido por la combinación de las CNN y los *transformer*. Por otro lado, el tiempo total de entrenamiento incrementa al complejizar la arquitectura de la red. Sin embargo, puede observarse que el tiempo de entrenamiento de los métodos desarrollados en este capítulo tienen un orden de magnitud menor que los dos métodos que utilizan RNN, presentados en el capítulo anterior. Además, en los experimentos aquí presentados fue necesario establecer el número máximo de épocas en 500 para alcanzar el óptimo, mientras que en el desarrollo presentado en el capítulo anterior sólo se necesitaron 100. La diferencia entre ambos se debe a que en este último algoritmo se utilizó la penalización  $L_2$ , la cual conduce a una convergencia más suave pero, como contraparte, a un mayor tiempo de entrenamiento.

Al igual que en el capítulo anterior, en la figura 5.3 se muestran tres hipnogramas simplificados obtenidos mediante la red «óptima»: un hipnograma para el cual se obtuvo un resultado en torno al error promedio y los otros dos con un desempeño superior e inferior al promedio.

<sup>1</sup>El caso más conocido de cómo el desbalance afecta las métricas es el de la exactitud. Ante una base de datos con un desbalance, digamos, del 90%, un clasificador que clasifique todas las muestras como clase mayoritaria tendrá una exactitud del 90%.

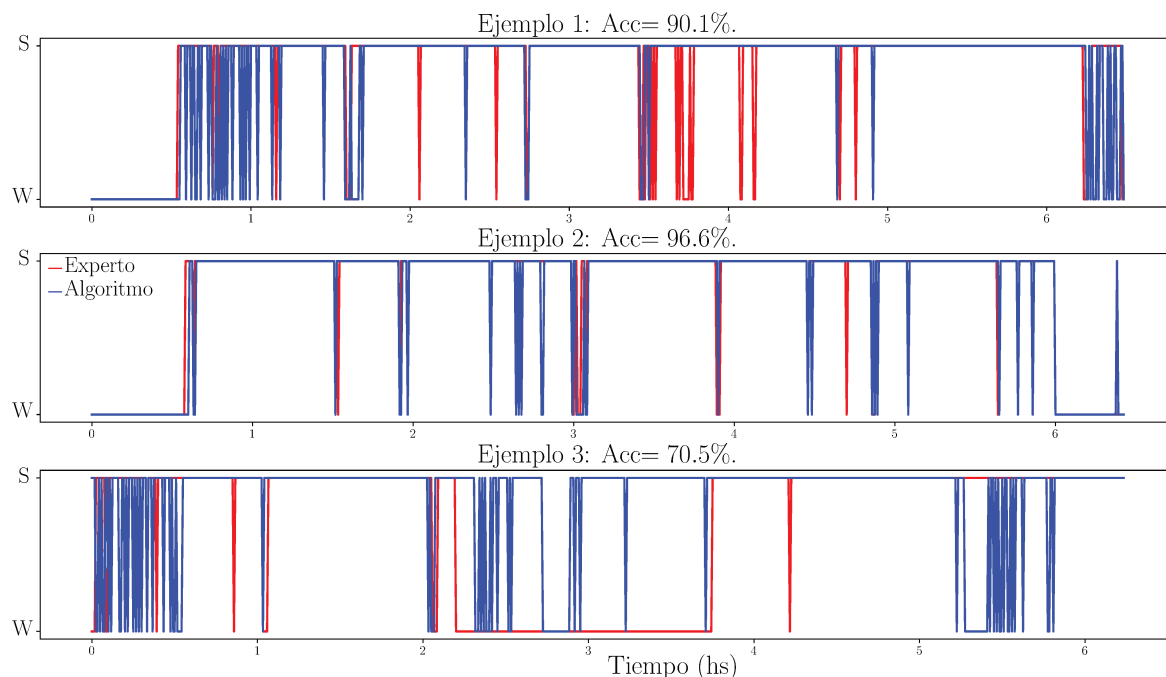


Figura 5.3: Hipnogramas simplificados obtenidos mediante clasificación con la red neuronal «óptima» desarrollada en este capítulo, basada en TCN y *transformers*. En la parte superior de la figura se muestran los hipnogramas de referencia (etiquetado por un conjunto de expertos) y la clasificación realizada por el algoritmo en un sujeto para el que se obtuvo un desempeño cercano al promedio. En el medio se muestra otro ejemplo, esta vez con un desempeño superior al promedio. Abajo, en cambio, se muestra un tercer ejemplo para el cual el desempeño fue inferior al promedio. En líneas rojas se presenta el etiquetado realizado por expertos, mientras que en líneas azules se presenta el etiquetado realizado por el algoritmo.

### 5.2.3. Clasificación de fases del sueño según AASM

De la misma forma que en el capítulo anterior, realizamos algunas modificaciones a la arquitectura, como parte de un estudio preliminar, para lograr una mayor discriminación de las fases de sueño. De esta forma, logramos adecuarnos a los estándares requeridos por la AASM, lo que podría resultar en la continuación de una línea lateral a la desarrollada por esta tesis, ampliando su posibilidad de aplicación.

De la misma forma que antes, el punto de partida fue la arquitectura con la que se obtuvo el mejor desempeño en los experimentos anteriores, pero modificando la capa de salida *2-softmax* transformándola en una *5-softmax*, de forma de que sea capaz de clasificar las 5 fases del sueño –W, N1, N2, N3 y REM–. Se consideró como entrada la HR obtenida del oxímetro de pulso, utilizando los mismos parámetros de entrenamiento que antes. A su vez, el esquema de evaluación utilizado fue exactamente igual que antes, tomando la misma partición de la base de datos: un subconjunto de entrenamiento y validación para obtener el algoritmo óptimo que luego será evaluado en un subconjunto de testeo.

En función de los resultados obtenidos se fueron realizando algunas modificaciones adicionales a la arquitectura en pos de mejorar el desempeño. Dado que problemas de mayor complejidad suelen necesitar de clasificadores más complejos, se incrementó la cantidad de *encoders* a 3. Por otro lado, se utilizó como función de costo la CE tradicional, ya que la función de «costo focalizado» no presentó mejoras en este caso.

Como mencionamos, el algoritmo se evaluó utilizando los 1250 pacientes correspondientes a la base de datos de testeo. Se presentarán a continuación las mismas tablas del capítulo anterior, incluyendo ahora los resultados correspondientes a este nuevo desarrollo. Como

Método	Entradas	N	Acc	Acc <sub>W</sub>	Acc <sub>N1</sub>	Acc <sub>N2</sub>	Acc <sub>N3</sub>	Acc <sub>REM</sub>	$\kappa$
TCN+ <i>transformer</i>	HR	5000	72.2	80.8	1.7	81.0	39.6	76.3	0.60
<b>256-biGRU</b>	HR+SpO <sub>2</sub>	5000	73.2	85.6	0	75.7	60.8	75.4	0.63
CWT feat.+RF [30]	EEG	10	82.6	89.6	43.2	84.8	69.4	76.4	0.76
CNN+RNN [28]	EEG	62	86.2	87.3	59.8	90.3	81.5	89.3	0.80
CNN+RNN [28]	EEG†	20	82.0	84.7	46.6	85.9	84.8	82.4	0.76
CNN + RNN [65]	EEG†	61	84.3	89.2	52.2	86.8	85.1	85.0	0.79

Tabla 5.2: Desempeños y comparación con la literatura según AASM. Se presentan tres trabajos con los mejores desempeños obtenidos. †Estos trabajos utilizaron la misma base de datos, por lo que pueden ser comparados directamente.

Método	Entradas	N	Acc	Acc <sub>W</sub>	Acc <sub>Light</sub>	Acc <sub>Deep</sub>	Acc <sub>REM</sub>	$\kappa$
TCN+ <i>transformer</i>	HR	5000	74.1	80.8	78.7	39.6.1	76.3	0.62
<b>256-biGRU</b>	HR+SpO <sub>2</sub>	5000	75.7	85.6	73.7	60.1	75.4	0.65
Feat.+LDA [38]	PPG + acc	60	68.7	69.3	69.2	62.5	71.6	0.52

Tabla 5.3: Desempeño y comparación con la literatura al clasificar en W, liviano, profundo y REM.

explicamos en el capítulo anterior, existen diferentes agrupaciones de las fases del sueño de acuerdo con la complejidad de los métodos. En las tablas 5.2, 5.3 y 5.4 se presentarán los resultados según los requerimientos de la AASM, agrupando las fases de sueño liviano N1 y N2, y agrupando las fases de sueño N1, N2 y N3 para diferenciar entre no REM y REM. Además, para el caso de la tabla 5.2, se indican entre paréntesis las bases de datos utilizadas por trabajos de terceros, de forma de poder observar las variaciones en los desempeños y realizar una mejor comparación entre los métodos presentados.

Vemos que el nuevo algoritmo desarrollado presenta un desempeño levemente menor que el obtenido con el método basado en GRU. También puede verse que es diferente la tasa de aciertos para cada clase. El método basado en GRU presenta mejor desempeño para la detección de despierto y sueño profundo (N3), mientras que el método basado en mecanismos de atención lo supera en las fases de sueño liviano (N1 y N2). En este algoritmo se ve nuevamente el inconveniente del desbalance de clases, que complica el aprendizaje de las clases minoritarias.

En la figura 5.4 se muestran tres hipnogramas obtenidos mediante la red «óptima», basada en TCN y *transformers*: un hipnograma para el cual se obtuvo un resultado en torno al error promedio y los otros dos con un desempeño superior e inferior al promedio.

En la siguiente sección abordaremos una discusión exhaustiva del método desarrollado, además de una comparación con otros métodos del estado del arte, tal y como lo hemos hecho en los capítulos anteriores.

### 5.3. Discusión

En este capítulo se propuso una arquitectura basada principalmente en *transformers* para clasificar en despierto y dormido segmentos de señales de HR obtenidas por un oxímetro de pulso. Si bien el desempeño obtenido por este algoritmo es similar al presentado en el artículo anterior, el tiempo de entrenamiento es considerablemente menor. Esta característica, en conjunto con el hecho necesitar sólo un oxímetro de pulso para obtener los datos, hace que esta red tenga un potencial enorme para ser parte de un dispositivo de cribado para clasificar las fases del sueño en el hogar, así también como para dispositivos «vestibles».

Método	Entradas	N	Acc	Acc <sub>W</sub>	Acc <sub>NREM</sub>	Acc <sub>REM</sub>	$\kappa$
TCN+ <i>transformer</i>	HR	5000	83.6	80.8	86.8	76.3	0.71
<b>256-biGRU</b>	HR+SpO <sub>2</sub>	5000	85.2	85.6	87.5	75.4	0.74
Feat.+RF [35]	ECG (P1)	5	78.1	74.3	82.8	40.9	0.57
	ECG (H1)	5	87.1	83.5	91.1	52.1	0.74
	ECG (P2)	16	77.0	–	–	–	0.57
	ECG (H2)	2	76.8	–	–	–	0.43
Feat.+RF [118]	ECG	45	72.6	56.4	81.3	59.8	0.46

Tabla 5.4: Desempeño y comparación con la literatura al clasificar en W, REM y no REM. Para el caso de Yücelbaş se utilizaron dos bases de datos diferentes. Se presentan los resultados reportados en ese artículo, donde P1 y H1 corresponden a la base de datos 1 para sujetos con patologías (*patients*) y sanos (*healthy*). Lo mismo sucede para la segunda base de datos, con P2 y H2.

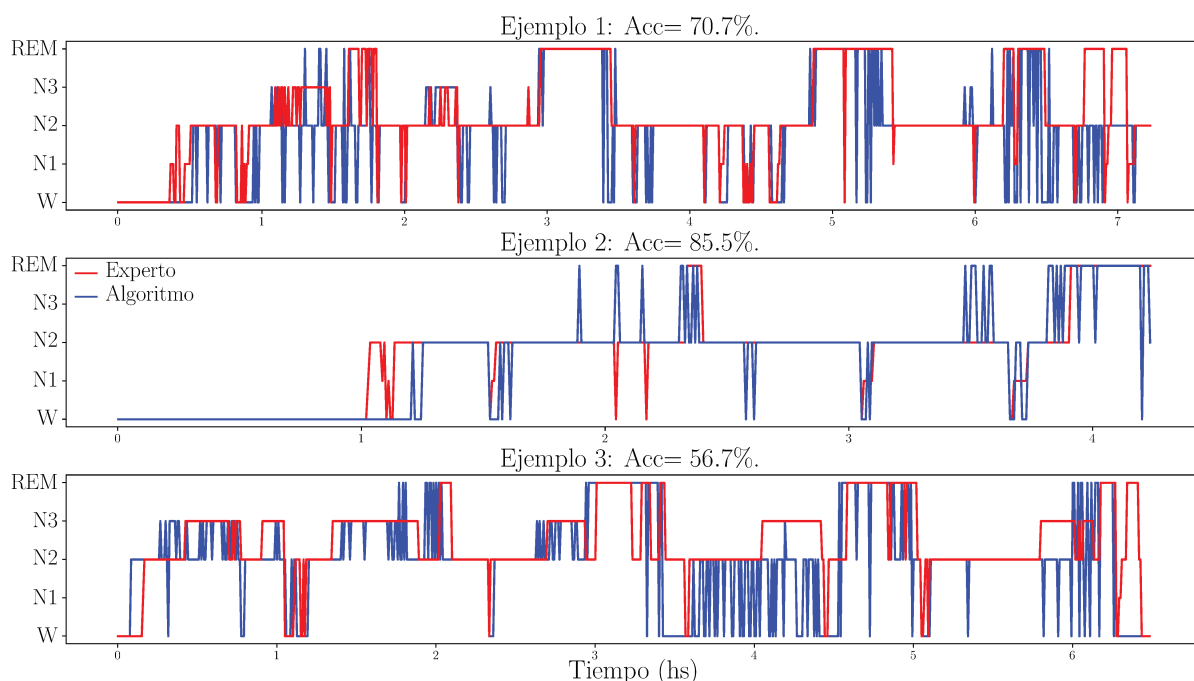


Figura 5.4: Hipnogramas obtenidos mediante clasificación con la red neuronal «óptima» desarrollada en este capítulo, basada en TCN y *transformers*. En la parte superior de la figura se muestran los hipnogramas de referencia (etiquetado por un conjunto de expertos) y la clasificación realizada por el algoritmo en un sujeto para el que se obtuvo un desempeño cercano al promedio. En el medio se muestra otro ejemplo, esta vez con un desempeño superior al promedio. Abajo, en cambio, se muestra un tercer ejemplo para el cual el desempeño fue inferior al promedio. En líneas rojas se presenta el etiquetado realizado por expertos, mientras que en líneas azules se presenta el etiquetado realizado por el algoritmo.

En esta sección introduciremos algunas discusiones respecto a las arquitecturas utilizadas y, hacia el final, se realizará una comparación detallada con el estado del arte para ver en que posición se han logrado ubicar los algoritmos desarrollados.

### 5.3.1. RNN y redes basadas en mecanismos de atención

Las RNNs ya cuentan con una larga historia, remontándose sus orígenes a la década de los ochenta con la primera formulación de algo que podría asemejarse a una RNN, realizada



por John Hopfield [131]. Esta arquitectura, presentada en 1982, surgió desde el área de las neurociencias con el objetivo de comprender los mecanismos subyacentes asociados a la memoria. Pero el gran salto en este tipo de arquitecturas se produjo en el año 1997, cuando Hochreiter y Schmidhuber presentaron las LSTM [107]. Mediante estas redes se lograba abordar un problema muy extendido en todas las redes que se entrenaban mediante retropropagación, especialmente en aquellas que tenían conexiones recurrentes: hablamos del problema de desvanecimiento del gradiente. Si bien al abordar este problema se logró que este tipo particular de RNN, las LSTM, fueran capaces de almacenar información a través del tiempo –memoria de la red–, no fue hasta el surgimiento de las arquitecturas modernas de GPUs en que estas redes fueron capaces de utilizarse en múltiples aplicaciones, tomando especial relevancia en el procesamiento de secuencias.

Sin embargo, aunque efectivo, el procesamiento secuencial realizado por las LSTM y sus variantes continúa siendo difícil de entrenar, ya que no se adapta para ser paralelizados y aprovechar las ventajas del *hardware*, lo que resulta en un tiempo de entrenamiento prolongado. Por estos motivos, y dado que la cantidad de datos de las secuencias disponibles para ser procesados crece a una velocidad mucho mayor que la capacidad de cómputo, comenzaron a surgir métodos cuyo objetivo era lograr un correcto procesamiento de secuencias, pero a una mayor velocidad. La mayoría de estas arquitecturas estaban relacionadas con las CNNs, ya que son uno de los métodos más optimizados para ser entrenados en GPUs. En este terreno, aparecieron en 2017 los *transformers*, basados en redes convolucionales y mecanismos de atención [120]. Mediante estas redes se lograron resultados comparables a los obtenidos mediante RNNs en tareas que involucraban entradas y salidas secuenciales. Sin embargo, estas arquitecturas presentan la ventaja de tener un entrenamiento muchísimo más veloz y, en teoría, serían capaces de «retener» la información durante mucho más tiempo. Es decir, las salidas para un tiempo dado pueden tener una influencia considerable de muestras que han sucedido hace mucho tiempo, ya que focalizan su atención en elementos determinados de la secuencia –de allí su nombre–, en lugar de propagar la información a través de todos los estados internos.

En los experimentos realizados en este capítulo hemos comprobado empíricamente estos postulados. Y si bien los resultados obtenidos no son superiores a los presentados en el capítulo anterior, la velocidad de entrenamiento superior de este método permite estudiar modificaciones en las arquitecturas y procesar los datos de forma mucho más veloz, lo que permite una mejor exploración de los parámetros de la red. Esto es una característica muy importante para este tipo de desarrollos tecnológicos, en los que se cuenta con una capacidad limitada de cálculo. Más aún si consideramos que en un futuro podrían formar parte de dispositivos domiciliarios para el cribado de patologías del sueño, en donde la capacidad del *hardware* asociado a su procesamiento es una restricción a tener en cuenta.

### 5.3.2. Comparación con el estado del arte

Presentaremos nuevamente, y por última vez, la tabla 5.5 con un resumen de todos los trabajos en el área, incorporando, ahora sí, todos los desarrollos realizados en esta tesis. Se indican, además, la cantidad de clases en que realiza la clasificación, el método utilizado y la cantidad de señales. Por propósitos comparativos, las fases N1, N2, N3 y REM se agruparon en una única clase correspondiente a dormido. Por ser la última vez que se presenta esta comparación, recordaremos las principales características de estos trabajos, describiéndolos en forma más extensa que en los capítulos anteriores.

Los primeros algoritmos que intentaron clasificar el sueño surgieron como una forma natural de automatizar las prácticas reglamentadas por las guías de etiquetado del sueño [8]. Hemos mencionado anteriormente que estos métodos pueden agruparse en dos categorías: los

Método	Señales	C	N	Acc	Se	Sp	PPV	NPV	$\kappa$
CWT feat.+RF [30]	EEG	5	16	93.4	95.9	89.6	93.3	93.5	0.86
CNN + RNN [28]	EEG	5	20	94.3	96.8	83.4	86.0	96.2	0.81
CNN + RNN [65]	EEG	5	61	95.7	96.9	90.6	87.8	97.7	0.86
Feat.+SVM+RFE [34]	ECG	2	18	80.0	84.5	69.1	87.0	64.5	0.52
CNN [36]	ECG (CGMH-val)	2	27	83.1	89.4	52.4	90.1	50.5	0.41
CNN [36]	ECG (DREAMS)	2	20	81.4	87.1	53.1	90.2	45.2	0.38
CNN [36]	ECG (UCDSADB)	2	25	73.7	81.9	43.4	84.3	39.2	0.24
TCN+transformer	HR	2	5000	90.0	94.9	78.1	91.3	85.2	0.73
2 GRU-(128) [119]	HR	2	5000	89.1	94.2	76.8	90.7	83.6	0.71
2 GRU-(256) [119]	HR+SpO <sub>2</sub>	2	5000	90.1	94.1	80.3	92.1	84.7	0.74
Feat. + FFS-SVM [64]	HR	2	5000	73.7	80.9	54.6	48.6	84.7	0.44
Feat.+SVM [37]	HRV	2	10	73.1	74.0	72.0	–	–	0.46
Feat.+SVM [37]	PPG	2	10	76.8	76.0	77.0	–	–	0.53
Feat.+kNN [37]	PPG+HRV	2	10	79.4	77.0	81.0	–	–	0.59
CNN [36]	PPG (CGMH-val)	2	27	84.2	90.9	51.5	90.1	53.6	0.43
Feat.+RNN [66]	ECG+acc	5	778	90.1	94.0	72.9	–	75.1	0.65
Feat.+LDA [38]	PPG+acc	4	60	90.6	94.6	69.3	94.3	70.5	0.64

Tabla 5.5: Comparación con otros trabajos que realizan clasificación automática de las fases del sueño. C representa la cantidad de clases en las que se realiza la clasificación, 2 corresponde a despierto y dormido, 4 a despierto, sueño liviano, sueño profundo y REM y 5 a la clasificación según AASM. N es la cantidad de sujetos en la base de datos. Se presentan en gris más oscuro los desempeños obtenidos por el trabajo presentado en este capítulo y en gris más claro los desarrollos presentados en capítulos anteriores.

algoritmos multicanal y los de canal único. Los primeros están basados en uno o más canales de EEG, EMG y EOG, y suelen tener un mejor desempeño que los segundos, basados en un único canal de EEG. Sin embargo, estos algoritmos no suelen ser adecuados para estudios domiciliarios debido a su complejidad, por lo que pueden afectar el sueño del paciente [31].

Siguiendo la forma en que se presentaron las discusiones en los capítulos anteriores, aquí se incluirán en la comparación dos trabajos previos en los cuales se utiliza EEG. Fraiwan y cols. [30] desarrollaron un método en el cual se clasifican las fases del sueño a partir de un canal de EEG de acuerdo con los estándares propuestos por la AASM. En este método se propuso el uso de medidas de entropías y distribuciones tiempo-frecuencia para la extracción de características, que luego son clasificadas mediante bosques aleatorios. A pesar de que en este trabajo las características fueron diseñadas *ad hoc* por los investigadores, los autores lograron un desempeño notable. De hecho, hemos mencionado que es el algoritmo con el mejor desempeño en una comparación realizada en la revisión de Boostani y cols. del año 2017 [31]. En ese mismo año, y ya con los métodos de aprendizaje profundo en pleno auge, Supratak y cols. [28] propusieron una red neuronal basada en CNN para extraer las características en forma automática, también utilizando un único canal de EEG. Luego, mediante RNN se obtuvo la clasificación de acuerdo con la AASM. También basado en métodos de aprendizaje profundo basados en un único canal de EEG, Mousavi y cols. [65] utilizaron una arquitectura muy similar a la de Supratak, pero que además incorporaba mecanismos de atención. Hasta donde conocemos, estos artículos contienen los mejores resultados para clasificación de las fases del sueño a partir de un único canal de EEG. Es por esto que se constituyen como la referencia final para cuantificar cuán aceptable es el rendimiento de nuestros métodos.

Además de los algoritmos basados en un único canal de EEG, existen los algoritmos que utilizan señales cardíacas para desarrollar dispositivos domiciliarios de cribado de las fases del sueño. En estos trabajos se explota la relación entre las fases del sueño y la regulación de

la HR, modificada por el balance simpático/parasimpático [32]. En este grupo de trabajos se encuentra el de Adnane y cols. [34] que desarrollaron un método clásico de aprendizaje maquina utilizando ECG para clasificar las fases del sueño en despierto y dormido. Este método también está basado en características diseñadas y extraídas *ad hoc* por los investigadores, sobre las que se aplica un clasificador basado en máquinas de soporte vectorial junto con un método de eliminación recursiva de características. En ese trabajo se desarrolló un esquema sujeto-específico, es decir, registros correspondientes a un mismo sujeto se utilizaron tanto para entrenar como para testear. Esto en general conduce a un incremento en las medidas de desempeño respecto a usar diferentes sujetos para entrenar y testear. Otro trabajo es el de Malik y cols. [36] donde se hizo uso de una CNN para clasificar las fases del sueño a partir de señales de HR obtenidas a partir de ECG. También se probaron los algoritmos diseñados entrenando y testeando con bases de datos diferentes. Más aún, en algunos experimentos evaluaron el desempeño al utilizar la HR obtenida a partir de PPG. Todos los desempeños obtenidos son similares, por lo que en la tabla de discusión reportaremos únicamente aquellos que corresponden a ECG y PPG para una misma base de datos.

Uçar y cols. [37] desarrollaron un método para clasificar las fases del sueño en despierto y dormido utilizando PPG, variabilidad de HR y una combinación de ambas. La clasificación se realizó utilizando SVM o  $k$ -vecinos más cercanos. En la tabla 5.5 reportamos los mejores resultados para cada una de las combinaciones de entradas.

Por último, existen algunos trabajos que incorporan información adicional a la información cardíaca. Una de las señales más utilizadas con este objetivo es la que proviene de acelerómetros, dada la simpleza con que pueden añadirse estos sensores a los dispositivos electrónicos sin incrementar su complejidad en términos de usabilidad o instrumentación. Beattie y cols. [38] utilizaron en forma combinada señales de PPG y acelerómetros para clasificar las fases del sueño en despierto, dormido y sueño REM. Si bien utilizaron características diseñadas *ad hoc* por los investigadores, los desempeños obtenidos son muy buenos. Otro trabajo que sigue esta línea es el de Fonseca y cols., [66] en el cual se utilizan ECG y señales de acelerómetros para clasificar las fases del sueño, esta vez de acuerdo a los lineamientos propuestos por la AASM. En este caso el algoritmo tiene un enfoque mixto, donde se utilizaron características diseñadas manualmente, pero luego se clasificaron utilizando una arquitectura basada en RNN. Los resultados que obtienen de esta forma son notables. Además, en este trabajo se utilizaron diferentes bases de datos para entrenar y testear la red, lo cual es una ventaja adicional.

Como hemos mencionado en todos los capítulos, la comparación entre diferentes algoritmos es un proceso bastante complicado, ya que todos ellos fueron diseñados utilizando diferentes bases de datos y tipos de señales. Pero a pesar de esta variedad de ejemplos, vemos que los resultados obtenidos por nuestros algoritmos –particularmente el algoritmo basado en TCN y *transformer*, así también como el anterior, basado en RNNs– obtienen desempeños comparables con los mejores resultados del estado del arte. Además, estos resultados se obtuvieron utilizando una gran base de datos, lo que beneficia la generalización de nuestros resultados.

### 5.3.3. Clasificación de las fases del sueño según AASM

En este apartado describimos la modificación del algoritmo desarrollado en este capítulo para clasificar el estado de sueño en despierto y dormido, de forma de que sea capaz de distinguir las fases del sueño con mayor discriminación, según las recomendaciones de la AASM. Ahora discutiremos los resultados obtenidos, comparándolos con los principales trabajos del estado del arte para dimensionar en qué posición estamos respecto al desempeño. Recordemos que esta discusión abarca también los resultados del algoritmo basado en GRU para

clasificar las fases del sueño, cuya discusión quedó pendiente en el capítulo anterior para evitar reiterarla nuevamente aquí.

El mejor desempeño para la clasificación automática de las fases del sueño es obtenido mediante señales de EEG. Existen innumerables trabajos que realizan esta clasificación a partir de ECG. En este apartado mencionamos únicamente tres de ellos. Boostani y col. [31] han realizado una revisión completa comparando numerosos trabajos basados en EEG. Nosotros hemos comparado nuestro resultado con el trabajo que obtuvo el mejor desempeño en esa revisión, realizado por Fraiwan y col. [30]. En él se propusieron tres variaciones de un método para clasificar, considerando además diferentes clasificadores. Aquí reportamos el mejor desempeño obtenido, cuya exactitud es 82.6 % y el índice Kappa de Cohen es 0.76. Los sujetos utilizados eran sospechosos de patologías del sueño. También presentamos un trabajo basado en CNN y RNN desarrollado por Supratak y cols. [28], en el cual se utilizaron dos bases de datos correspondientes a sujetos sanos. Con la arquitectura propuesta obtuvieron una exactitud de 86.2 % para una de las bases de datos –compuesta por 62 sujetos– y 82.0 % para la otra –compuesta por 20 sujetos–, mientras que los índices Kappa de Cohen obtenidos fueron 0.80 y 0.76, respectivamente. Por último, presentamos un trabajo también basado en CNN y RNN, pero incorporando un mecanismo de atención, desarrollado por Mousavi y cols. [65], en el cual se utilizó una de las bases de datos utilizada por Supratak. El mejor desempeño obtenido por este trabajo fue 84.3 %, mientras que el índice Kappa de Cohen obtenido fue 0.79. Para ello utilizaron 61 sujetos pertenecientes a una de las bases de datos usadas por Supratak. Estos resultados obtenidos con EEG son una referencia, una especie de límite ideal al que deberíamos aspirar con nuestro algoritmo. Sin embargo, es lógico que nuestro desempeño esté por debajo al partir de una clasificación realizada con señales de oximetría de pulso. Además, nuestra base de datos está constituida por pacientes con patologías cardíacas, por lo que es de esperar que al utilizar sujetos sanos la clasificación mejore un poco. En la tabla 5.2 se presentan estos resultados.

La fase N1 es muy minoritaria respecto al resto, siendo una de las más difíciles de detectar –la formulación actual de nuestro algoritmo no ha podido aprenderla–. Esta fase se considera una fase de transición hacia N2, durando muy poco tiempo. Por estos motivos, es común combinar las fases N1 y N2. Además, la discriminación del sueño en W, liviano (N1 y N2), profundo (N3) y REM continúa siendo clínicamente muy útil. Para tener una idea de la magnitud del desbalance, hemos calculado en nuestra base de datos el porcentaje correspondiente a cada clase, representando un 30 % de las muestras la fase W, un 3 % la fase N1, un 40 % la fase N2, un 13 % la fase N3 y un 14 % la fase REM. En la tabla 5.3 se presenta un trabajo que realiza esta clasificación desarrollado por Beattie y col. [38]. Este trabajo ha sido el mejor de los presentados hasta ahora que no utilizan EEG. Además, tiene la particularidad de que también utiliza la señal de PPG –en su totalidad, no únicamente la HR–, aunque complementando con información provista por acelerómetros. En ese trabajo se obtiene una exactitud de 68.7 % y un índice kappa de Cohen de 0.52. Vemos que el desempeño obtenido por ese algoritmo es considerablemente inferior al nuestro, a pesar de que utiliza señales mucho más informativas. Finalmente, en favor de nuestro algoritmo, la base de datos utilizada por Beattie y col. corresponde a sujetos con sueño normal.

En la tabla 5.4, la última presentada, se exponen los resultados de trabajos que agrupan las fases del sueño de forma de diferenciar vigilia, REM y no REM. Yücelbaş y col. [35] utilizaron señales de EEG provenientes de dos bases de datos diferentes, las cuales contienen 10 y 18 sujetos. En este trabajo se discriminan los resultados obtenidos para los sujetos sanos y para aquellos que presentan patologías. Vemos que en casi todos los casos, nuestro trabajo supera en aproximadamente un 10 % la exactitud obtenida por Yücelbaş y col., sucediendo algo similar con el índice kappa de Cohen. Además, en ese trabajo se utilizaron segmentos del mismo paciente para entrenar y testear las redes, lo cual incrementa el desempeño obtenido.

Este tipo de enfoques suelen justificarse para presentar sistemas adaptables a un sujeto particular. En caso de métodos de cribado, esto no aplica y lo ideal sería utilizar pacientes diferentes para entrenar y testear, con el consecuente deterioro en el desempeño.

Otro trabajo que realiza la misma agrupación de fases del sueño es el de Xiao y col. [118]. En ese trabajo se analizaron segmentos de señal de 5 minutos correspondientes a un único estado, por lo que posee el problema discutido en el capítulo anterior respecto a la longitud  $L$  de los segmentos. A pesar de esto, la exactitud obtenida por este trabajo está muy por debajo de la que se obtiene con nuestro algoritmo, la cual es directamente aplicable en una situación real.

Por último, si recordamos los datos referenciados en el capítulo 2 respecto a la variabilidad interprofesional, veremos que el índice kappa de Cohen obtenido fue  $\kappa = 0.76$ . Es decir, estamos obteniendo un grado de acuerdo similar al que obtienen los profesionales médicos [57].

## 5.4. Conclusiones

En este capítulo se describió el diseño de un modelo basado en mecanismos de atención para clasificar las fases del sueño en despierto y dormido utilizando señales de HR, provistas por un oxímetro de pulso comercial. El modelo desarrollado está basado en TCN para aprender las características de la señal de HR, que luego serán procesadas mediante *transformers* para procesar las secuencias, aprendiendo las reglas de transición entre fases del sueño y permitiendo su clasificación. La red neuronal fue desarrollada bajo un esquema de evaluación basado en subconjuntos de entrenamiento, validación y testeo. Los resultados obtenidos mediante esta red son similares a los obtenidos en el capítulo anterior, pero con un tiempo de entrenamiento que es un orden de magnitud menor. Esa velocidad de entrenamiento superior constituye una ventaja comparativa importante respecto al algoritmo anterior, ya que facilita el proceso de diseño al acelerar considerablemente los experimentos.

Se mostró que las TCN superan a las CNN o las SLP para extraer características en forma automática en la etapa del aprendizaje de una representación que luego serán procesadas por la parte de la red encargada de la modelización de la secuencia. Estos resultados no hacen otra cosa más que comprobar las ventajas de las redes residuales por sobre las CNN.

Los resultados obtenidos por estas redes son comparables con los desempeños obtenidos por algoritmos que utilizan señales mucho más complejas. Además, al igual que todos los algoritmos desarrollados en este artículo, la base de datos utilizada es considerablemente mayor que las que utilizan otros autores. Adicionalmente, la base de datos contiene pacientes con patologías y se utilizó completamente, incluyendo todos los pacientes sin importar la cantidad de desconexiones y consecuente interpolación de los datos. Por último, es una base de datos adquirida hace aproximadamente 25 años. Por todo esto, se espera que el algoritmo se comporte mejor en señales más estables, obtenidas con tecnología actual y en un entorno más controlado –como suelen ser las bases de datos de los trabajos con los que nos comparamos. Actualmente estamos en tratativas con instituciones de nuestro país para acceder a bases de datos recientes desarrolladas en un centro de excelencia a nivel nacional.

Por todas estas razones, la red propuesta en este capítulo constituye un nuevo avance para el diagnóstico simplificado de patologías del sueño.

El algoritmo desarrollado en este capítulo ha visto demorada su presentación para publicarse en forma de artículo en una revista científica, dado que una institución vinculada a la incubación de empresas de base tecnológica mostró interés en desarrollar esta tecnología para la conformación de una empresa. Por este motivo, al momento actual se están realizando negociaciones para la presentación de una solicitud de patente en Estados Unidos, cuyos inventores serán quien suscribe esta tesis y su director y codirector. Una vez realizada

la presentación de patente, se presentarán los artículos para ser evaluados por una revista científica. En resumen, los desarrollos presentados en este capítulo dieron lugar a:

- Solicitud en la Oficina de Patentes y Marcas Registradas del Gobierno de Estados Unidos, cuyos inventores serán R. Casal, L. E. Di Persia, G. Schlotthauer y Sergio Richter y los derechos de propiedad estarán a cargo de las instituciones UNER, UNL, CONICET y el Centro de Innovación Tecnológica Empresarial y Social (CITES) .
- Casal, Ramiro, Leandro E. Di Persia, y Gastón Schlotthauer. Temporal convolutional networks and transformer for classifying the sleep stage in awake or asleep using pulse oximetry signals. A presentar.

## Capítulo 6

# Acercamiento a trabajos futuros

En más de una ocasión, a lo largo del documento, se mencionó la existencia de un subdiagnóstico generalizado de las patologías del sueño, particularmente del SAOS, como resultado de las limitaciones en el acceso a los estudios del sueño. En los métodos actuales para cribado del SAOS, el subdiagnóstico se ve incrementado debido a una subestimación del AHI. Esto sucede dado que se utiliza el TTR como aproximación del TTS, al cual no se tiene acceso en los métodos simplificados que no registran EEG. El subdiagnóstico que esto genera es reportado incluso por la Clasificación Internacional de Patologías del Sueño, confeccionada por la AASM [9].

También en varias ocasiones se comentó que el objetivo final de la línea de trabajo en la que se enmarca esta tesis es desarrollar un método confiable de cribado de SAOS –aunque extensible a otras patologías del sueño– utilizando únicamente registros de oximetría de pulso. Para ello, la estimación del TTS constituye sólo una fracción del trabajo total a realizar ya que, además, se debe considerar la detección de eventos de apnea e hipopnea, lo que permitirá obtener la estimación del AHI. Ese desarrollo será parte de los trabajos futuros que continuarán a esta tesis. Sin embargo, para dar un cierre más completo a este documento, se decidió incorporar algunos resultados preliminares, haciendo uso de métodos ya existentes para la estimación de los eventos de apnea pero incorporando ahora el TTS estimado a partir de los desarrollos de esta tesis. Además, se incorporan en la discusión las principales estrategias a seguir para el momento de desarrollar un nuevo método, así como también un anticipo de los problemas que se encontrarán.

### 6.1. Incorporación del TTS a un método para diagnóstico de AHI

Con el objetivo de evaluar si la introducción del TTS estimado podría ser útil para corregir el problema de subestimación mencionado, realizamos una serie de experimentos replicando un algoritmo publicado en el artículo *Screening of obstructive sleep apnea with empirical mode decomposition of pulse oximetry* [19] –que ya fue mencionado en el capítulo 2 al revisar el estado del arte– que utiliza las señales provenientes de un oxímetro de pulso para clasificar la severidad del SAOS. En ese trabajo, entre cuyos autores se encuentran tanto el director como el codirector de esta tesis, se presentó un algoritmo basado en descomposición empírica en modos (*empirical mode decomposition*, EMD) para detectar desaturaciones asociadas con eventos de apnea/hipoapnea utilizando únicamente señales de oximetría de pulso. El objetivo fue obtener un índice que se comporte de forma similar al AHI derivado de la PSG, permitiendo un diagnóstico rápido, eficiente y económico del SAOS.

Para detectar las desaturaciones ese trabajo reconoce patrones de onda de acuerdo a los

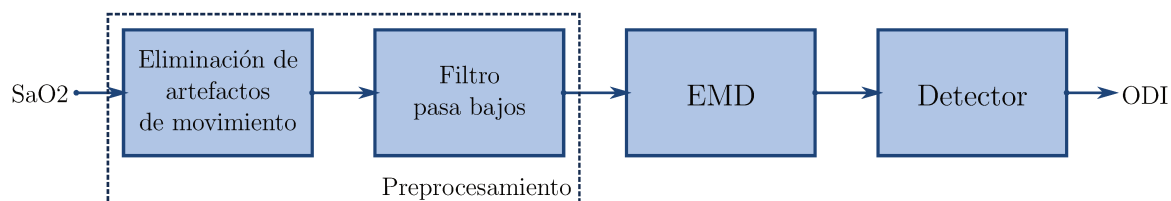


Figura 6.1: Diagrama en bloques del algoritmo desarrollado por Schlotthauer y cols. [19] para cribado de SAOS.

criterios que utilizan los médicos y técnicos en la práctica clínica para realizar el etiquetado manual. Estos criterios ya fueron mencionados en el capítulo 2, pero los recordaremos a continuación para que esta sección sea autocontenida. Las apneas no son consideradas diferentes a las hipoapneas, ya que poseen una fisiopatología similar. Para que un evento sea considerado apnea/hipoapnea, debe satisfacer los siguientes criterios 1 y 2, más el criterio 3:

1. La amplitud de una señal relacionada con la respiración debe presentar un decremento claro ( $>50\%$ ) de su línea de base. La línea de base es definida como la amplitud media de la señal durante respiración y oxigenación estable en los dos minutos precedentes al evento.
2. Una clara reducción de la amplitud en una medida validada de la respiración que no alcance el criterio previo, pero que ocurra con una desaturación mayor o igual a un  $3\%$  o un despertar.
3. La duración del evento debe ser de al menos 10 segundos.

Como es de esperarse, en ese trabajo se intenta inferir el IAH a partir del segundo y tercer criterio. A continuación se describirá brevemente el algoritmo, para luego entender las modificaciones que realizamos y los resultados obtenidos.

## 6.2. Algoritmo propuesto por Schlotthauer y cols. basado en EMD

En la figura 6.1 puede verse un diagrama en bloques del algoritmo. En el algoritmo se realizó un preprocesamiento de la señal de  $SpO_2$  que consiste en dos pasos secuenciales. En primer lugar, se eliminaron las porciones en las que la señal es inválida y se concatenaron los datos. La identificación de las porciones inválidas de la señal para su posterior remoción se realizó mediante la señal complementaria de calidad provista por el oxímetro de pulso. A diferencia de el preprocesamiento realizado en nuestros desarrollos, en ese trabajo no se realizó ningún tipo de interpolación porque producía perturbaciones en el algoritmo de EMD, conduciendo a un peor funcionamiento general. La concatenación, aunque parezca un enfoque demasiado simple, fue la que permitió obtener los mejores resultados.

La señal de  $SpO_2$  posee una resolución de aproximadamente  $1\%$ , por lo que tiene un nivel de cuantización importante. Por ello, en segundo lugar, se utilizó un filtro pasabajos de tipo respuesta finita al impulso (*finite impulse response*, FIR) con una frecuencia de corte en  $0.25$  Hz. Este filtrado reduciría el ruido de cuantización sin afectar la morfología de las apneas, ya que se espera que estos eventos posean una duración mayor a 5 segundos.

Luego de ese preprocesamiento, el algoritmo descompone la señal en 6 modos mediante EMD. El primer modo contiene ruido asociado con la cuantización, por lo que no posee información útil, mientras que los modos 2 a 5 contienen datos relativos a los eventos de



desaturación. El modo 6 y el residuo final contendrían información irrelevante, según los autores. Dado que las desaturaciones se descomponen en varios modos, resulta imposible la utilización de un único modo. Para superar este inconveniente, los autores evaluaron todas las posibles combinaciones de los modos considerados.

Sobre esta señal generada como la suma de modos, se realiza una detección de los eventos de apnea basándose en los criterios clínicos 2 y 3 mencionados. Para ello, cada señal se procesa buscando los extremos. Luego, se obtienen la diferencia en amplitud entre cada extremo local y el siguiente ( $\Delta A$ ) y el correspondiente intervalo de tiempo ( $\Delta T$ ). Además se definen dos umbrales  $\tau_A$  y  $\tau_T$ . De esta forma, si  $\Delta A$  y  $\Delta T$  superan los umbrales correspondientes, el evento se anota como una desaturación. A partir de esto se estima el índice de desaturaciones de oxígeno (*oxygen desaturation index*, ODI), definido como la relación entre el número de eventos de desaturaciones detectados y el TTS:

$$\text{ODI} = \frac{\text{Núm. de desaturaciones}}{\text{TTS}}. \quad (6.1)$$

Como sabemos, el TTS no es accesible para ese algoritmo, por lo que fue reemplazado por el TTR. Aquí es donde entra en juego la modificación realizada en esta tesis, donde evaluaremos la diferencia de desempeño entre los resultados obtenidos utilizando el TTR, los resultados obtenidos utilizando el TTS estimado por los algoritmos desarrollados en esta tesis y, finalmente como referencia, los resultados obtenidos utilizando el TTS real.

### 6.3. Resultados

En el algoritmo es necesario determinar experimentalmente algunos parámetros óptimos con el objetivo de obtener el mejor desempeño. Por un lado, debe encontrarse la mejor combinación de modos a utilizar y, por otro lado, deben encontrarse los umbrales  $\tau_A$  y  $\tau_T$  que mejor permitan estimar las desaturaciones.

Como se comentó anteriormente, Schlotthauer y cols. evaluaron todas las posibles combinaciones de los modos 2 a 5. El mejor resultado se obtuvo generando una señal construida a partir de la suma de los modos 3, 4 y 5. Por una cuestión de tiempo computacional, al replicar los experimentos no realizamos nuevamente esta búsqueda, sino que directamente se utilizó la combinación «óptima» provista por los autores. De esta manera se redujo el costo computacional del experimento, que de por sí es bastante alto, permitiendo la obtención de resultados similares en un tiempo menor.

En cambio, sí repetimos la búsqueda para obtener la mejor combinación posible de los umbrales  $\tau_A$  y  $\tau_T$ , entre otras cosas porque al trabajar con diferentes tiempos como denominador para calcular el ODI –TST estimado, TST real y TRT– es de esperar que se encuentren diferentes  $\tau_A$  y  $\tau_T$  óptimos.  $\tau_A$  se modificó desde 0.1 hasta 5 en pasos de 0.1, mientras  $\tau_T$  se modificó desde 10 hasta 50 en pasos 1. Para cada una de las combinaciones se estimaron los ODI con los diferentes tiempos considerados. Fue necesario repetir estos experimentos tal y cómo se realizó en el artículo, ya que ante alguna pequeña diferencia en la implementación, o incluso también ante la utilización de diferentes datos, podría suceder que los umbrales reportados por el artículo no sean los mismos en nuestro caso –aunque deberían ser similares–.

Para cada una de las posibles combinaciones de parámetros  $\tau_A$  y  $\tau_T$  se calculó el ODI. Luego, para evaluar el mejor resultado, se estableció un umbral en los valores de ODI estimados para cada paciente de manera tal de clasificarlos en dos categorías de acuerdo a si el ODI se encontraba por encima o debajo del umbral. Las categorías en las que se clasificaron los pacientes fueron según su valor de AHI estimado mediante PSG, para lo cual se utilizaron tres valores de referencia: AHI= 5, AHI= 10 y AHI= 15 –tres valores de corte

Método	AHI <sub>thr</sub>	AUC
	5	0.888
EMD	10	0.912
	15	0.923

Tabla 6.1: Resumen de los resultados para cribado de SAOS encontrados en el artículo de Schlotthauer y cols. [19].

Denominador ecuación 6.1	$\tau_A$	$\tau_T$	AUC	$\mathbf{E}_1$
TTR	1.1	19	0.902	6.76
T $\hat{S}$ T - RNN	2.7	18	0.913	4.23
T $\hat{S}$ T - <i>Transformer</i>	2.7	19	0.909	4.31
TST real	2.8	20	0.917	4.05

Tabla 6.2: Corrección de los resultados para cribado de SAOS obtenidos en el artículo de Schlotthauer y cols. [19]. Los resultados que se muestran se obtuvieron para AHI= 15. Se presentan además los parámetros óptimos, el área bajo la curva y el error entre el ODI estimado y el AHI de referencia.

muy usuales en estos cribados de SAOS—. El valor de corte en el ODI se estableció utilizando la curva ROC, de forma de maximizar el AUC al obtener esa clasificación [75]. Los valores obtenidos por el trabajo original se muestran en la tabla 6.1 [19]. El punto de corte más utilizado para clasificar el AHI según su severidad es 15, ya que a partir de dicho valor se considera que la patología tiene un grado moderado. Para AHI= 15, ese trabajo encontró que el mejor desempeño se obtuvo con  $\tau_A = 1.1$  y  $\tau_T = 19$ . Asimismo, el umbral para el ODI fue 18.5.

Estos resultados fueron obtenidos utilizando 996 datos de la ya conocida base de datos SHHS, dividiendo los datos en un subconjunto para encontrar los umbrales óptimos y en otro subconjunto para evaluarlos. Para replicar estos experimentos, nosotros utilizamos parte de los 1250 sujetos que forman parte del subconjunto de testeo, calculando el TTS mediante las redes que ya habían sido entrenadas según se mencionó en el capítulo 4 y en este capítulo. Es esperable que al modificar los datos, los resultados también se vean modificados levemente.

Utilizando 100 sujetos pertenecientes a la base de datos de testeo se establecieron los umbrales óptimos  $\tau_A$  y  $\tau_T$  para cada uno de los tres ODI calculados utilizando TTS estimado, TTS real y TTR. Luego, se utilizaron los parámetros óptimos encontrados en otros 200 sujetos diferentes, también de la base de datos de testeo. En este caso, no sólo calculamos el AUC obtenido al establecer un umbral y clasificar a los sujetos por encima o por debajo de un valor AHI de referencia, sino que además calculamos los errores entre los ODI estimados y el AHI real. Para estos experimentos preliminares, únicamente utilizamos el valor de corte AHI= 15. Además de estimar el AUC tal y cómo se reportó en el artículo de Schlotthauer y cols. [19], en nuestro caso evaluamos también el error absoluto medio entre el ODI estimado y el AHI de referencia, haciendo:

$$\mathbf{E}_1 = \frac{1}{N} \sum_{i=1}^N |\text{AHI}_i - \text{ODI}_i| \quad (6.2)$$

siendo  $N$  la cantidad de sujetos en la base de datos,  $\text{AHI}_i$  el valor de AHI para el  $i$ -ésimo sujeto y  $\text{ODI}_i$  el valor estimado del ODI. En la tabla 6.2 se reportan todos los resultados obtenidos al utilizar los tres posibles denominadores en la ecuación 6.1.

## 6.4. Discusión

De los resultados presentados pueden extraerse una serie de conclusiones. En primer lugar, pareciera que el algoritmo mejora levemente al introducir una estimación del TTS diferente a la utilizada por el algoritmo original. La mejora en el caso del AUC es relativamente pequeña, aunque para el error  $\mathbf{E}_1$  esta diferencia se incrementa un poco. Esta variación entre el AUC y el  $\mathbf{E}_1$  se deben a que el error cometido se «diluye» para calcular el AUC, ya que en ese caso la población se divide en dos clases. Por lo tanto, supongamos que se estimó un ODI= 25, cuando en realidad su valor de AHI era 35. Esta diferencia se verá reflejada en el error  $\mathbf{E}_1$ , pero no lo hará para el caso del AUC ya que a pesar del error, el sujeto sigue perteneciendo a la misma clase.


En segundo lugar, el algoritmo detecta las desaturaciones en base a una serie de parámetros determinados en forma experimental. El denominador, para el caso de utilizar el TTR, siempre produce una sobreestimación del TTS y, luego, una subestimación del índice. Para compensar esto, los parámetros  $\tau_A$  y  $\tau_T$  pueden elegirse de forma que el detector sea más sensible y, consecuentemente, también sobreestime la cantidad de desaturaciones. De esta manera, la sobreestimación del denominador se compensa con una sobreestimación del numerador, haciendo que el error disminuya. Esto se ve claramente reflejado en los valores que toman los umbrales, principalmente  $\tau_A$ . El umbral de amplitud  $\tau_A$  es casi tres veces menor que para los otros métodos que realizan una mejor estimación del TTS. De esta forma, el algoritmo original lograba compensar la falta de una estimación del TTS. Cabe decir que esto se debe en parte a que el algoritmo de Schlotthauer y cols. [19] con el cual trabajamos es un algoritmo para cribado y, por lo tanto, su objetivo no era lograr una buena estimación de los índices, sino lograr una correcta clasificación de la severidad del paciente.

En tercer y último lugar, es necesario mencionar que la comparación entre el AHI y el ODI para construir el error  $\mathbf{E}_1$  no es completamente correcta, por lo que no es realista esperar obtener un error muy bajo de esta manera. Como ya se ha visto, no todo evento de apnea o hipopnea produce una desaturación. Por lo tanto, estimar el AHI a través de la detección de desaturaciones inevitablemente conducirá a un error. Este error puede no ser significativo para algoritmos de cribado, pero pesará más al momento de intentar construir un indicador que se comporte de forma similar al AHI.

Para evitar esto, nuestro grupo de trabajo plantea atacar en el futuro el problema de detección de eventos de apnea e hipopnea mediante redes neuronales con un enfoque similar al utilizado en los métodos presentados en esta tesis para clasificar las fases del sueño. Para ello, se utilizarán las redes diseñadas –tanto la presentada en el capítulo anterior como la de este capítulo– con pequeñas modificaciones, de forma de que acepten como entradas las señales provenientes del oxímetro de pulso y obtengan una salida que será un vector con las etiquetas de apneas e hipopneas. Una vez obtenida una correcta clasificación de apneas e hipopneas, sólo restaría contarlas y dividir el resultado por el TTS calculado por las redes ya desarrolladas. La utilización de estas redes permitiría detectar formas de onda menos evidentes que las ondas diente de sierra buscadas en algoritmos como el de Schlotthauer y cols. [19] y, por lo tanto, lograr un indicador más similar al AHI.

En este sentido, al momento de finalizar esta tesis ya hemos realizado algunos experimentos preliminares obteniendo resultados que no fueron del todo buenos. Existen algunas diferencias entre la tarea de clasificación de fases de sueño y la clasificación en apneas e hipopneas que creemos que son los causantes del pobre desempeño obtenido hasta ahora, y que serán los pasos a atacar en el futuro. En primer lugar, el desbalance entre clases ya presente para clasificar en despierto y dormido se agrava notablemente para este caso. Dado que las hipopneas y las apneas tienen una morfología bastante diferente –independientemente de que en términos de consecuencias en la práctica clínica no se las considere diferentes–, es

habitual considerarlas como dos clases diferentes. Las clases a considerar como salida del clasificador, entonces, serían «apnea», «hipopnea» y «ningún evento». El predominio de la clase «ningún evento» hace que sea mucho más difícil entrenar a las redes neuronales para que eviten la solución trivial. Para abordar este inconveniente, se deberán evaluar diferentes estrategias para contrarrestar el desbalance de clases.

Por otro lado, mientras que para la clasificación de fases de sueño se realiza una clasificación cada 30 segundos, para el caso de las apneas esta clasificación tiene una frecuencia de 1 segundo. Para el algoritmo presentado en el capítulo anterior, basado en RNN, esto no será un problema, ya que para adaptarlo a esta nueva aplicación únicamente habrá que eliminar el bloque de voto mayoritario presente al final. Sin embargo, en el caso del algoritmo presentado en este capítulo, la situación es más compleja, ya que la TCN realiza una reducción de dimensión a lo largo del tiempo, mientras que, por otro lado, incrementa la resolución de la  una resolución de 1 segundo.

Nuestros trabajos a futuro abordarán la clasificación de los eventos de apnea e hipopnea mediante señales provenientes del oxímetro de pulso teniendo todas estas hipótesis en mente.

## 6.5. Conclusiones

En este capítulo se presentaron los resultados preliminares de un experimento realizado y se delinearon los trabajos que continuarán esta tesis. Se utilizó un algoritmo desarrollado por terceros para calcular la variación en el desempeño al introducir el TTS estimado por los algoritmos diseñados en los capítulos anteriores con respecto al uso del TTR, tal y cómo lo hacía el algoritmo original. Los resultados mejoran levemente su desempeño. Sin embargo, dadas las características del algoritmo utilizado, sus parámetros se adaptan de tal forma que pueden compensar la sobreestimación del TTS al utilizar el TTR mediante una sobreestimación del número de eventos de apnea e hipopnea –estableciendo los umbrales óptimos para su detección de manera que sean más sensibles–.

Se planteó la necesidad de desarrollar algoritmos que den con una mejor estimación del número de eventos de apnea e hipopnea. Estos desarrollos constituyen los trabajos futuros de esta tesis, que partirán de las arquitecturas diseñadas en los dos capítulos anteriores y se modificarán para lograr superar las dificultades identificadas: el desbalance de clases, la diferente morfología entre los eventos de «apnea» e «hipopnea» y la diferente resolución en la frecuencia de clasificaciones con respecto a las fases del sueño.

# Capítulo 7

## Conclusiones

En esta tesis doctoral desarrollamos un conjunto de algoritmos con el objetivo de brindar apoyo al diagnóstico de patologías del sueño. Las principales contribuciones realizadas permiten la clasificación del estado de sueño en despierto y dormido a partir de señales de oximetría de pulso, resultado cuya aplicación es de interés para el estudio de numerosas patologías del sueño, en general, y para el SAOS en particular.

A modo de introducción, comenzamos mencionando las prevalencias de las patologías del sueño así como su método de diagnóstico estándar: la PSG. Con motivo de las limitaciones en el acceso a estos estudios para la población, principalmente debido a su costo, laboriosidad y escasez, existe un importante subdiagnóstico en este tipo de patologías. Por esta razón, y como luego mencionamos, existe un creciente interés de la comunidad científica y de la práctica clínica para desarrollar métodos de cribado que permitan abordar este problema. Luego de hacer un breve repaso de las opciones disponibles, establecimos que nuestra propuesta estaría basada en la oximetría de pulso por ser una solución económica y simple que, además, es capaz de representar información relacionada tanto con la dinámica del sueño como con los eventos respiratorios característicos del SAOS, una de las principales patologías del sueño.

En el contexto de un marco teórico, continuamos luego con una exposición minuciosa de los conceptos fisiológicos necesarios para comprender la aplicación. De esta manera, definimos el SAOS y el AHI para determinar su severidad. Describimos, ahora con más detalle, los métodos de diagnóstico según sus diferentes niveles de complejidad, desde la PSG hasta la oximetría de pulso, haciendo hincapié en una de las principales falencias de las alternativas simplificadas para cribado: la incapacidad de estimar el TTS para calcular adecuadamente el AHI. Esta cantidad, entonces, es reemplazada por el TTR, conduciendo a una subestimación del AHI y agravando aún más el problema de subdiagnóstico. Posteriormente, discutimos de forma exhaustiva las características morfológicas y el etiquetado de las fases del sueño y de los eventos de apnea e hipopnea sobre las distintas señales registradas en la PPG. Finalmente, presentamos una recopilación del estado del arte para ambas aplicaciones.

A partir de aquí queda clara la motivación para desarrollar algoritmos que permitan clasificar el estado de sueño en despierto y dormido y estimar el TTS a partir de señales de oximetría de pulso. En este sentido, entonces, presentamos el primer desarrollo de esta tesis: un método para clasificar en forma automática las fases del sueño en despierto y dormido utilizando señales de HR provenientes de un oxímetro de pulso. Este primer algoritmo, basado en técnicas clásicas de aprendizaje maquinal, nos permitió comprobar que las características relacionadas con la teoría de la información y medidas de complejidad, así como sus extensiones al dominio tiempo-frecuencia, son útiles para diferenciar las dos clases de interés. Asimismo, a partir de este desarrollo logramos corroborar que la utilización de señales muy simples y de bajo costo, como lo son las obtenidas por oximetría de pulso, tienen un potencial significativo para abordar la clasificación de las fases del sueño. Si bien el desem-

peño obtenido con este algoritmo luego fue ampliamente superado, su diseño nos permitió conocer en forma más exhaustiva la dinámica de la señal y sus cambios en los estados de sueño despierto/dormido, conocimiento necesario para lograr un adecuada extracción de las características para su posterior clasificación.

En los experimentos llevados a cabo para evaluar el desempeño del algoritmo diseñado quedó claro que el hecho de que cada segmento se clasifique individualmente y en forma aislada resultaba en una gran desventaja, ya que la información de segmentos adyacentes al clasificado aporta información relevante. Además, esta situación impedía reducir la longitud del segmento al nivel recomendado por los consensos de medicina respiratoria. Estas dos relevantes limitaciones resultaron en motivaciones para los siguientes algoritmos. De esta forma, desarrollamos un nuevo método que realice la clasificación en forma secuencial y permita tener «memoria» de las clasificaciones previas y futuras. Esto se logró mediante la creación de una arquitectura basada en un tipo particular de RNN: las GRU. Esta arquitectura constituyó en dos capas de GRU bidireccionales que permitían clasificar las fases del sueño en despierto y dormido utilizando la señales de HR y SpO<sub>2</sub> que provee el oxímetro de pulso. Con este nuevo algoritmo quedó demostrado que con un correcto procesamiento de las señales pueden lograrse resultados notables a pesar de su simplicidad, quedando su desempeño sólo por debajo de los métodos que utilizan EEG. También presentamos una modificación del algoritmo de manera tal que permita una mayor discriminación de las cinco fases de sueño establecidas por la AASM, siendo nuevamente capaz de aprender las reglas de transición a pesar del mayor detalle requerido. Con este resultado se superó aproximadamente en un 7% el desempeño del estado del arte de métodos similares que incluso contienen mayor información. A partir de esta variación, y dado su buen desempeño, es posible pensar en el desembarco en otras aplicaciones tales como monitoreo de somnolencia de conductores o aplicación a dispositivos «vestibles» para monitoreo personal de la salud. Sin embargo, el algoritmo presenta el inconveniente de una baja tasa de aciertos en algunas fases del sueño, principalmente para el caso N1. A pesar de esto, el desempeño general alcanzado tiene un grado de acuerdo equivalente al grado de acuerdo interprofesional obtenido por médicos expertos, por lo que este algoritmo tiene un enorme potencial en el cribado automático de las fases del sueño.

Una limitación de las RNN en este contexto es su naturaleza secuencial, que impide su paralelización y conduce a un proceso de aprendizaje muy lento, generando importantes restricciones de memoria. Por esta razón, luego propusimos una arquitectura basada en mecanismos de atención que permitan superar esta desventaja, obteniendo entrenamientos muchísimo más rápidos con desempeños muy similares. La arquitectura se compone de TCN, que permiten extraer características de la señal de HR que luego fueron procesadas mediante *transformers*. Observamos que las redes de atención también fueron capaces de modelar las dependencias temporales entre muestras de las secuencias de entrada y salida, aprendiendo las reglas de transición entre las clases despierto/dormido y permitiendo su clasificación. A partir de los experimentos realizados, vimos que las TCN superaron a las CNN o las FNN para extraer características en forma automática en la primera etapa de la red. Este comportamiento ya ha sido observado en varios trabajos y fue el motivo del surgimiento de las redes residuales.

Con el desarrollo de estos dos últimos algoritmos logramos estimar automáticamente el TTS con un alto nivel de confiabilidad, lo que resulta de gran utilidad en métodos de cribado de patologías del sueño en general y del SAOS en particular y que constituye el principal aporte de esta tesis doctoral. A modo de análisis preliminar, realizamos experimentos donde incorporamos el TTS a un método de detección de apneas. Los resultados obtenidos sugieren una leve ventaja respecto de considerar el TTR. Sin embargo, creemos que el mayor potencial de estos algoritmos en el cribado de SAOS surgirá a partir de trabajos futuros, que esperamos

realizar como continuación de esta tesis. A lo largo del trabajo aquí realizado pudimos mostrar la capacidad tanto de las RNN como de las redes de atención para una clasificación secuencial de señales provenientes de oximetría de pulso, en este caso para las distintas fases del sueño. Así, es posible pensar en utilizar técnicas similares para desarrollar un detector de apneas o hipopneas. Con una correcta detección de las apneas e hipopneas, y en conjunto con los algoritmos desarrollados, podría lograrse un estimador que se comporte de manera similar al AHI, obteniendo un algoritmo que permita un estudio del SAOS, simplificado pero altamente confiable.

En esta línea, los primeros pasos a seguir son adaptar los algoritmos desarrollados para obtener una clasificación en las clases «apnea», «hipopnea» y «ningún evento» utilizando también señales provenientes de un oxímetro de pulso. Hemos realizado algunos experimentos preliminares que han tenido la utilidad de mostrar los principales desafíos a afrontar en esta tarea. Por un lado, el desbalance de clases se acrecienta notablemente, existiendo un predominio excesivo de la clase «ningún evento» por sobre las demás, por lo que deberá ponerse especial énfasis en las estrategias para combatir este desbalance. Por otro lado, mientras que en la determinación de las fases de sueño se realiza una clasificación cada segmentos de 30 segundos, en la determinación de los eventos de apnea e hipopnea es necesario clasificar para cada una de las muestras. Es decir, una clasificación por segundo. Esto requiere generar una serie de modificaciones en el último algoritmo presentado, ya que en la etapa que utiliza TCN realizamos una reducción a lo largo del tiempo. Es muy probable que las arquitecturas deban ser modificadas de forma de adaptarlas a esta nueva aplicación.

Los algoritmos desarrollados no ven limitada su aplicabilidad al SAOS, sino que pueden tener una importancia en otras patologías del sueño y en otras aplicaciones relacionadas. Esto último es especialmente cierto para la modificación de los algoritmos que permite discriminar más fases de sueño. Como dijimos, estos algoritmos pueden ser de gran utilidad no sólo para el ámbito clínico, sino que resultan muy simples de adaptar para ser utilizados en dispositivos portátiles y para monitoreo personal de la salud. Sería una gran ventaja aplicar algoritmos que originalmente fueron pensados y diseñados con los parámetros de confiabilidad requeridos para el ámbito clínico en el ámbito de los dispositivos de consumo.

Como conclusión general, consideramos que las contribuciones desarrolladas en esta tesis constituyen un avance en las técnicas de cribado de patologías del sueño, permitiendo pensar en la posibilidad de obtener métodos simples, económicos y de amplia aplicabilidad sin pérdidas considerables de desempeño. Los algoritmos diseñados, junto a los desarrollos que emanen de los trabajos futuros sugeridos, nos permiten pensar que alcanzar un diagnóstico eficiente, simple y universal para el SAOS, en particular, y para muchas patologías del sueño, en general, son una posibilidad concreta y relativamente cercana. El incremento en la capacidad y potencia de las nuevas herramientas de aprendizaje maquinal, complementadas con un correcto análisis y procesamiento de señales, permiten, si existe la voluntad social y política necesaria, imaginar una transformación esencial en el acceso universal a la salud.

# Bibliografía

- [1] R. Stickgold, “Sleep-dependent memory consolidation,” *Nature*, vol. 437, no. 7063, pp. 1272–1278, 2005.
- [2] M. M. Ohayon, “Epidemiological overview of sleep disorders in the general population,” *Sleep Medicine Research*, vol. 2, no. 1, pp. 1–9, 2011.
- [3] A. Fido and A. Ghali, “Detrimental effects of variable work shifts on quality of sleep, general health and work performance,” *Medical Principles and Practice*, vol. 17, no. 6, pp. 453–457, 2008.
- [4] C. Fuhrman, B. Fleury, X.-L. Nguyễn, and M.-C. Delmas, “Symptoms of sleep apnea syndrome: high prevalence and underdiagnosis in the french population,” *Sleep medicine*, vol. 13, no. 7, pp. 852–858, 2012.
- [5] V. Kapur, K. P. Strohl, S. Redline, C. Iber, G. O’Connor, and J. Nieto, “Underdiagnosis of sleep apnea syndrome in US communities,” *Sleep and Breathing*, vol. 6, no. 02, pp. 049–054, 2002.
- [6] J. L. Hossain and C. M. Shapiro, “The prevalence, cost implications, and management of sleep disorders: an overview,” *Sleep and Breathing*, vol. 6, no. 02, pp. 085–102, 2002.
- [7] R. S. Leder, G. Schlotthauer, T. Penzel, and R. Jané, “The natural history of the sleep and respiratory engineering track at EMBC 1988 to 2010,” in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pp. 288–291, IEEE, 2010.
- [8] R. B. Berry, R. Brooks, C. E. Gamaldo, S. M. Harding, C. Marcus, and B. Vaughn, “The AASM manual for the scoring of sleep and associated events,” *Rules, Terminology and Technical Specifications, Darien, Illinois, American Academy of Sleep Medicine*, 2012.
- [9] M. J. Sateia, “International classification of sleep disorders: highlights and modifications,” *Chest Journal*, vol. 146, no. 5, pp. 1387–1394, 2014.
- [10] M. R. Mannarino, F. Di Filippo, and M. Pirro, “Obstructive sleep apnea syndrome,” *European journal of internal medicine*, vol. 23, no. 7, pp. 586–593, 2012.
- [11] Y. Sun, Y. Ning, L. Huang, F. Lei, Z. Li, G. Zhou, and X. Tang, “Polysomnographic characteristics of daytime sleepiness in obstructive sleep apnea syndrome,” *Sleep and Breathing*, vol. 16, no. 2, pp. 375–381, 2012.
- [12] P. J. Strollo Jr and R. M. Rogers, “Obstructive sleep apnea,” *New England Journal of Medicine*, vol. 334, no. 2, pp. 99–104, 1996.



- [13] E. Shahar, C. W. Whitney, S. Redline, E. T. Lee, A. B. Newman, F. Javier Nieto, G. T. O'CONNOR, L. L. Boland, J. E. Schwartz, and J. M. Samet, "Sleep-disordered breathing and cardiovascular disease: cross-sectional results of the sleep heart health study," *American journal of respiratory and critical care medicine*, vol. 163, no. 1, pp. 19–25, 2001.
- [14] N. AlGhanim, V. R. Comondore, J. Fleetham, C. A. Marra, and N. T. Ayas, "The economic impact of obstructive sleep apnea," *Lung*, vol. 186, no. 1, pp. 7–12, 2008.
- [15] D. Leger, V. Bayon, J. P. Laaban, and P. Philip, "Impact of sleep apnea on economics," *Sleep medicine reviews*, vol. 16, no. 5, pp. 455–462, 2012.
- [16] A. Rechtschaffen, "A manual of standardized terminology, technique and scoring system for sleep stages of human subjects," *Public Health Service*, 1968.
- [17] K. P. Pang and D. J. Terris, "Screening for obstructive sleep apnea: an evidence-based analysis," *American Journal of Otolaryngology*, vol. 27, no. 2, pp. 112–118, 2006.
- [18] A. Yadollahi and Z. Moussavi, "Apnea detection by acoustical means," in *Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE*, pp. 4623–4626, IEEE, 2006.
- [19] G. Schlotthauer, L. E. Di Persia, L. D. Larrateguy, and D. H. Milone, "Screening of obstructive sleep apnea with empirical mode decomposition of pulse oximetry," *Medical Engineering & Physics*, vol. 36, no. 8, pp. 1074–1080, 2014.
- [20] R. Rolón, L. Larrateguy, L. Di Persia, R. Spies, and H. Rufiner, "Discriminative methods based on sparse representations of pulse oximetry signals for sleep apnea–hypopnea detection," *Biomedical Signal Processing and Control*, vol. 33, pp. 358–367, 2017.
- [21] L.-W. Hang, H.-L. Wang, J.-H. Chen, J.-C. Hsu, H.-H. Lin, W.-S. Chung, and Y.-F. Chen, "Validation of overnight oximetry to diagnose patients with moderate to severe obstructive sleep apnea," *BMC Pulmonary Medicine*, vol. 15, no. 1, p. 24, 2015.
- [22] J.-C. Vázquez, W. H. Tsai, W. W. Flemons, A. Masuda, R. Brant, E. Hajduk, W. A. Whitelaw, and J. E. Remmers, "Automated analysis of digital oximetry in the diagnosis of obstructive sleep apnoea," *Thorax*, vol. 55, no. 4, pp. 302–307, 2000.
- [23] J. V. Marcos, R. Hornero, D. Álvarez, F. del Campo, and C. Zamarrón, "Assessment of four statistical pattern recognition techniques to assist in obstructive sleep apnoea diagnosis from nocturnal oximetry," *Medical Engineering & Physics*, vol. 31, no. 8, pp. 971–978, 2009.
- [24] F. Roche, E. Sforza, D. Duverney, J.-R. Borderies, V. Pichot, O. Bigaignon, G. Ascher, and J.-C. Barthélémy, "Heart rate increment: an electrocardiological approach for the early detection of obstructive sleep apnoea/hypopnoea syndrome," *Clinical Science*, vol. 107, no. 1, pp. 105–110, 2004.
- [25] J. I. Salisbury and Y. Sun, "Rapid screening test for sleep apnea using a nonlinear and nonstationary signal processing technique," *Medical Engineering & Physics*, vol. 29, no. 3, pp. 336–343, 2007.

- [26] B. Raymond, R. Cayton, and M. Chappell, “Combined index of heart rate variability and oximetry in screening for the sleep apnoea/hypopnoea syndrome,” *Journal of Sleep Research*, vol. 12, no. 1, pp. 53–61, 2003.
- [27] R. G. Norman, I. Pal, C. Stewart, J. A. Walsleben, and D. M. Rapoport, “Interobserver agreement among sleep scorers from different centers in a large dataset.,” *Sleep*, vol. 23, no. 7, pp. 901–908, 2000.
- [28] A. Supratak, H. Dong, C. Wu, and Y. Guo, “Deepsleepnet: a model for automatic sleep stage scoring based on raw single-channel eeg,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 11, pp. 1998–2008, 2017.
- [29] H. Phan, F. Andreotti, N. Cooray, O. Y. Chén, and M. De Vos, “Seqsleepnet: end-to-end hierarchical recurrent neural network for sequence-to-sequence automatic sleep staging,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 3, pp. 400–410, 2019.
- [30] L. Fraiwan, K. Lweesy, N. Khasawneh, H. Wenz, and H. Dickhaus, “Automated sleep stage identification system based on time–frequency analysis of a single EEG channel and random forest classifier,” *Computer methods and programs in biomedicine*, vol. 108, no. 1, pp. 10–19, 2012.
- [31] R. Boostani, F. Karimzadeh, and M. Nami, “A comparative review on sleep stage classification methods in patients and healthy individuals,” *Computer methods and programs in biomedicine*, vol. 140, pp. 77–91, 2017.
- [32] T. Penzel, J. W. Kantelhardt, L. Chung-Chang, K. Voigt, and C. Vogelmeier, “Dynamics of heart rate and sleep stages in normals and patients with sleep apnea,” *Neuropsychopharmacology*, vol. 28, no. S1, p. S48, 2003.
- [33] S. Aeschbacher, M. Bossard, T. Schoen, D. Schmidlin, C. Muff, A. Maseli, J. D. Leuppi, D. Miedinger, N. M. Probst-Hensch, A. Schmidt-Trucksäss, *et al.*, “Heart rate variability and sleep-related breathing disorders in the general population,” *The American Journal of Cardiology*, vol. 118, no. 6, pp. 912–917, 2016.
- [34] M. Adnane, Z. Jiang, and Z. Yan, “Sleep–wake stages classification and sleep efficiency estimation using single-lead electrocardiogram,” *Expert Systems with Applications*, vol. 39, no. 1, pp. 1401–1413, 2012.
- [35] Ş. Yücelbaş, C. Yücelbaş, G. Tezel, S. Özşen, and Ş. Yosunkaya, “Automatic sleep staging based on SVD, VMD, HHT and morphological features of single-lead ECG signal,” *Expert Systems with Applications*, vol. 102, pp. 193–206, 2018.
- [36] J. Malik, Y.-L. Lo, and H.-t. Wu, “Sleep-wake classification via quantifying heart rate variability by convolutional neural network,” *Physiological measurement*, vol. 39, no. 8, p. 085004, 2018.
- [37] M. K. Uçar, M. R. Bozkurt, C. Bilgin, and K. Polat, “Automatic sleep staging in obstructive sleep apnea patients using photoplethysmography, heart rate variability signal and machine learning techniques,” *Neural Computing and Applications*, pp. 1–16, 2016.

- [38] Z. Beattie, Y. Oyang, A. Statan, A. Ghoreyshi, A. Pantelopoulos, A. Russell, and C. Heneghan, “Estimation of sleep stages in a healthy adult population from optical plethysmography and accelerometer signals,” *Physiological Measurement*, vol. 38, no. 11, p. 1968, 2017.
- [39] J. Mantua, N. Gravel, and R. Spencer, “Reliability of sleep measures from four personal health monitoring devices compared to research-based actigraphy and polysomnography,” *Sensors*, vol. 16, no. 5, p. 646, 2016.
- [40] T. Young, M. Palta, J. Dempsey, J. Skatrud, S. Weber, and S. Badr, “The occurrence of sleep-disordered breathing among middle-aged adults,” *New England Journal of Medicine*, vol. 328, no. 17, pp. 1230–1235, 1993.
- [41] J. Durán, S. Esnaola, R. Rubio, and Á. Iztueta, “Obstructive sleep apnea–hypopnea and related clinical features in a population-based sample of subjects aged 30 to 70 yr,” *American journal of respiratory and critical care medicine*, vol. 163, no. 3, pp. 685–689, 2001.
- [42] E. O. Bixler, A. N. Vgontzas, T. Ten Have, K. Tyson, and A. Kales, “Effects of age on sleep apnea in men: I. prevalence and severity,” *American journal of respiratory and critical care medicine*, vol. 157, no. 1, pp. 144–148, 1998.
- [43] C. Guilleminault, A. Tilkian, and W. C. Dement, “The sleep apnea syndromes,” *Annual review of medicine*, vol. 27, no. 1, pp. 465–484, 1976.
- [44] A. A. of Sleep Medicine, *International Classification of Sleep Disorders, 3rd ed.* American Academy of Sleep Medicine, Darien, 2014.
- [45] D. J. Eckert, A. S. Jordan, P. Merchia, and A. Malhotra, “Central sleep apnea: pathophysiology and treatment,” *Chest*, vol. 131, no. 2, pp. 595–607, 2007.
- [46] R. Mehra, E. J. Benjamin, E. Shahar, D. J. Gottlieb, R. Nawabit, H. L. Kirchner, J. Sahadevan, and S. Redline, “Association of nocturnal arrhythmias with sleep-disordered breathing: The sleep heart health study,” *American journal of respiratory and critical care medicine*, vol. 173, no. 8, pp. 910–916, 2006.
- [47] H. Sala, C. Nigro, C. Rabec, A. S. Guardia, and M. Smurra, “Consenso argentino de trastornos respiratorios vinculados al sueño,” *Medicina (B Aires)*, vol. 61, pp. 351–63, 2001.
- [48] M. S. Aloia, J. T. Arnedt, J. D. Davis, R. L. Riggs, and D. Byrd, “Neuropsychological sequelae of obstructive sleep apnea-hypopnea syndrome: a critical review,” *Journal of the International Neuropsychological Society: JINS*, vol. 10, no. 5, p. 772, 2004.
- [49] F. Nogueira, E. Borsini, H. Cambursano, S. Marcela, E. Dibur, C. Franceschini, D. Pérez-Chada, L. Larrateguy, and C. Nigro, “Guías prácticas de diagnóstico y tratamiento del síndrome de apneas e hipopneas obstructivas del sueño: Actualización 2019: Sección sueño, oxigenoterapia y tratamientos crónicos domiciliarios. asociación argentina de medicina respiratoria,” *Rev. am. med. respir*, pp. 59–90, 2019.
- [50] J. B. Pietzsch, A. Garner, L. E. Cipriano, and J. H. Linehan, “An integrated health-economic analysis of diagnostic and therapeutic strategies in the treatment of moderate-to-severe obstructive sleep apnea,” *Sleep*, vol. 34, no. 6, pp. 695–709, 2011.

- [51] J. Durán-Cantolla, F. Puertas-Cuesta, G. Pin-Arboledas, J. Santa María Cano, J. Basco-González, R. Ugarte-Líbano, *et al.*, “Consenso nacional sobre el síndrome de apneas-hipopneas del sueño,” *Arch de Bronconeumol*, no. Supl 4, pp. 3–110, 2005.
- [52] B. Oeverland, O. Skatvedt, K. J. Kværner, and H. Akre, “Pulseoximetry: sufficient to diagnose severe sleep apnea,” *Sleep medicine*, vol. 3, no. 2, pp. 133–138, 2002.
- [53] J. Allen, “Photoplethysmography and its application in clinical physiological measurement,” *Physiological Measurement*, vol. 28, no. 3, p. R1, 2007.
- [54] P. A. Kyriacou, “Pulse oximetry in the oesophagus,” *Physiological Measurement*, vol. 27, no. 1, p. R1, 2005.
- [55] Z. Zhang, Z. Pi, and B. Liu, “TROIKA: A general framework for heart rate monitoring using wrist-type photoplethysmographic (PPG) signals during intensive physical exercise,” *Biomedical Engineering, IEEE Transactions*, vol. 62, no. 2, pp. 522 – 531, 2014.
- [56] J. R. Landis and G. G. Koch, “The measurement of observer agreement for categorical data,” *biometrics*, pp. 159–174, 1977.
- [57] H. Danker-hopfe, P. Anderer, J. Zeitlhofer, M. Boeck, H. Dorn, G. Gruber, E. Heller, E. Loretz, D. Moser, S. Parapatics, *et al.*, “Interrater reliability for sleep scoring according to the rechtschaffen & kales and the new aasm standard,” *Journal of sleep research*, vol. 18, no. 1, pp. 74–84, 2009.
- [58] C. W. Whitney, D. J. Gottlieb, S. Redline, R. G. Norman, R. R. Dodge, E. Shahar, S. Surovec, and F. J. Nieto, “Reliability of scoring respiratory disturbance indices and sleep staging,” *Sleep*, vol. 21, no. 7, pp. 749–757, 1998.
- [59] S. Özşen, “Classification of sleep stages using class-dependent sequential feature selection and artificial neural network,” *Neural Computing and Applications*, vol. 23, no. 5, pp. 1239–1250, 2013.
- [60] J. Shi, X. Liu, Y. Li, Q. Zhang, Y. Li, and S. Ying, “Multi-channel eeg-based sleep stage classification with joint collaborative representation and multiple kernel learning,” *Journal of neuroscience methods*, vol. 254, pp. 94–101, 2015.
- [61] A. Flexer, G. Gruber, and G. Dorffner, “A reliable probabilistic sleep stager based on a single EEG signal,” *Artificial intelligence in Medicine*, vol. 33, no. 3, pp. 199–207, 2005.
- [62] H. Phan, F. Andreotti, N. Cooray, O. Y. Chén, and M. De Vos, “Automatic sleep stage classification using single-channel eeg: Learning sequential features with attention-based recurrent neural networks,” in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 1452–1455, IEEE, 2018.
- [63] Y.-L. Hsu, Y.-T. Yang, J.-S. Wang, and C.-Y. Hsu, “Automatic sleep stage recurrent neural classifier using energy features of eeg signals,” *Neurocomputing*, vol. 104, pp. 105–114, 2013.
- [64] R. Casal, L. E. Di Persia, and G. Schlotthauer, “Sleep-wake stages classification using heart rate signals from pulse oximetry,” *Heliyon*, vol. 5, no. 10, p. e02529, 2019.

- [65] S. Mousavi, F. Afghah, and U. R. Acharya, “Sleepegnet: Automated sleep stage scoring with sequence to sequence deep learning approach,” *PloS one*, vol. 14, no. 5, p. e0216456, 2019.
- [66] P. Fonseca, M. M. van Gilst, M. Radha, M. Ross, A. Moreau, A. Cerny, P. Anderer, X. Long, J. P. van Dijk, and S. Overeem, “Automatic sleep staging using heart rate variability, body movements, and recurrent neural networks in a sleep disordered population,” *Sleep*, 2020.
- [67] D. Schlosshan and M. Elliott, “Sleep 3: Clinical presentation and diagnosis of the obstructive sleep apnoea hypopnoea syndrome,” *Thorax*, vol. 59, no. 4, p. 347, 2004.
- [68] I. T. Ling, A. L. James, and D. R. Hillman, “Interrelationships between body mass, oxygen desaturation, and apnea-hypopnea indices in a sleep clinic population,” *Sleep*, vol. 35, no. 1, pp. 89–96, 2012.
- [69] C. Zamarron, P. Romero, J. Rodriguez, and F. Gude, “Oximetry spectral analysis in the diagnosis of obstructive sleep apnoea,” *Clinical Science*, vol. 97, no. 4, pp. 467–473, 1999.
- [70] E. Chiner, J. Signes-Costa, J. M. Arriero, J. Marco, I. Fuentes, and A. Sergado, “Nocturnal oximetry for the diagnosis of the sleep apnoea hypopnoea syndrome: a method to reduce the number of polysomnographies?,” *Thorax*, vol. 54, no. 11, pp. 968–971, 1999.
- [71] B. Xie and H. Minn, “Real-time sleep apnea detection by classifier combination,” *IEEE Transactions on information technology in biomedicine*, vol. 16, no. 3, pp. 469–477, 2012.
- [72] R. E. Rolon, I. E. Gareis, L. D. Larrateguy, L. E. Di Persia, R. D. Spies, and H. L. Ruffiner, “Automatic scoring of apnea and hypopnea events using blood oxygen saturation signals,” *Biomedical Signal Processing and Control*, vol. 62, p. 102062, 2020.
- [73] S. Redline, M. H. Sanders, B. K. Lind, S. F. Quan, C. Iber, D. J. Gottlieb, W. H. Bonekat, D. M. Rapoport, P. L. Smith, J. P. Kiley, *et al.*, “Methods for obtaining and analyzing unattended polysomnography data for a multicenter study,” *Sleep*, vol. 21, no. 7, pp. 759–768, 1998.
- [74] E. J. Nieto, G. T. O’Connor, D. M. Rapoport, and S. Redline, “The sleep heart health study: design, rationale, and methods,” *Sleep*, vol. 20, no. 12, pp. 1077–1085, 1997.
- [75] T. Fawcett, “An introduction to ROC analysis,” *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.
- [76] M. Kuhn, K. Johnson, *et al.*, *Applied predictive modeling*, vol. 26. Springer, 2013.
- [77] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. John Wiley & Sons, 2012.
- [78] S. M. Pincus, “Approximate entropy as a measure of system complexity,” *Proceedings of the National Academy of Sciences*, vol. 88, no. 6, pp. 2297–2301, 1991.
- [79] F. Takens, “Detecting strange attractors in turbulence,” in *Dynamical Systems and Turbulence, Warwick 1980*, pp. 366–381, Springer, 1981.

- [80] L.-S. Xu, K.-Q. Wang, and L. Wang, “Gaussian kernel approximate entropy algorithm for analyzing irregularity of time-series,” in *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, vol. 9, pp. 5605–5608, IEEE, 2005.
- [81] J. F. Restrepo, G. Schlotthauer, and M. E. Torres, “Maximum approximate entropy and  $r$  threshold: A new approach for regularity changes detection,” *Physica A: Statistical Mechanics and its Applications*, vol. 409, pp. 97–109, 2014.
- [82] J. S. Richman and J. R. Moorman, “Physiological time-series analysis using approximate entropy and sample entropy,” *American Journal of Physiology-Heart and Circulatory Physiology*, vol. 278, no. 6, pp. H2039–H2049, 2000.
- [83] A. N. Kolmogorov, “A new metric invariant of transient dynamical systems and automorphisms in Lebesgue spaces,” in *Doklady Akademii Nauk*, vol. 119, pp. 861–864, Russian Academy of Sciences, 1958.
- [84] J.-P. Eckmann and D. Ruelle, “Ergodic theory of chaos and strange attractors,” in *The theory of chaotic attractors*, pp. 273–312, Springer, 1985.
- [85] P. Grassberger and I. Procaccia, “Estimation of the kolmogorov entropy from a chaotic signal,” *Physical review A*, vol. 28, no. 4, p. 2591, 1983.
- [86] C. Granero-Belinchon, S. G. Roux, P. Abry, M. Doret, and N. B. Garnier, “Information theory to probe intrapartum fetal heart rate dynamics,” *Entropy*, vol. 19, no. 12, p. 640, 2017.
- [87] K. H. Chon, C. G. Scully, and S. Lu, “Approximate entropy for all signals,” *IEEE engineering in medicine and biology magazine*, vol. 28, no. 6, pp. 18–23, 2009.
- [88] W. Chen, Z. Wang, H. Xie, and W. Yu, “Characterization of surface EMG signal based on fuzzy entropy,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 15, no. 2, pp. 266–272, 2007.
- [89] C. Bandt and B. Pompe, “Permutation entropy: a natural complexity measure for time series,” *Physical Review Letters*, vol. 88, no. 17, p. 174102, 2002.
- [90] B. Boashash, N. A. Khan, and T. Ben-Jabeur, “Time–frequency features for pattern recognition using high-resolution TFDs: A tutorial review,” *Digital Signal Processing*, vol. 40, pp. 1–30, 2015.
- [91] B. Boashash, *Time-frequency signal analysis and processing: a comprehensive reference*. Academic Press, 2015.
- [92] M. E. Torres, M. M. Anino, and G. Schlotthauer, “Automatic detection of slight parameter changes associated to complex biomedical signals using multiresolution q-entropy,” *Medical Engineering & Physics*, vol. 25, no. 10, pp. 859–867, 2003.
- [93] M. Aboy, R. Hornero, D. Abásolo, and D. Álvarez, “Interpretation of the Lempel-Ziv complexity measure in the context of biomedical signal analysis,” *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 11, pp. 2282–2288, 2006.
- [94] G. Schlotthauer, A. Humeau-Heurtier, J. Escudero, and H. L. Rufiner, “Measuring complexity of biomedical signals,” *Complexity*, vol. 2018, 2018.

- [95] T. F. of the European Society of Cardiology *et al.*, “Heart rate variability: standards of measurement, physiological interpretation, and clinical use,” *Circulation*, vol. 93, pp. 1043–1065, 1996.
- [96] M. Bonnet and D. Arand, “Heart rate variability: sleep stage, time of night, and arousal influences,” *Electroencephalography and Clinical Neurophysiology*, vol. 102, no. 5, pp. 390–396, 1997.
- [97] B. Boashash, L. Boubchir, and G. Azemi, “Time-frequency signal and image processing of non-stationary signals with application to the classification of newborn EEG abnormalities,” in *Signal Processing and Information Technology (ISSPIT), 2011 IEEE International Symposium on*, pp. 120–129, IEEE, 2011.
- [98] I. Guyon and A. Elisseeff, “An introduction to variable and feature selection,” *Journal of Machine Learning Research*, vol. 3, no. Mar, pp. 1157–1182, 2003.
- [99] R. Kohavi and G. H. John, “Wrappers for feature subset selection,” *Artificial Intelligence*, vol. 97, no. 1-2, pp. 273–324, 1997.
- [100] V. Vapnik, *The nature of statistical learning theory*. Springer Science & Business Media, 2013.
- [101] S. Abe, *Support vector machines for pattern classification*, vol. 2. Springer, 2005.
- [102] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [103] R. Casal and G. Schlotthauer, “Sleep detection in heart rate signals from photoplethysmography,” in *2017 XVII workshop on information processing and control (RPIC)*, pp. 1–6, IEEE, 2017.
- [104] N. Böhning, B. Schultheiss, S. Eilers, T. Penzel, W. Böhning, and E. Schmittendorf, “Comparability of pulse oximeters used in sleep medicine for the screening of OSA,” *Physiological Measurement*, vol. 31, no. 7, p. 875, 2010.
- [105] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, p. 436, 2015.
- [106] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” *arXiv preprint arXiv:1406.1078*, 2014.
- [107] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [108] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.
- [109] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, vol. 1. MIT press Cambridge, 2016.
- [110] S. Arlot, A. Celisse, *et al.*, “A survey of cross-validation procedures for model selection,” *Statistics surveys*, vol. 4, pp. 40–79, 2010.
- [111] C. Nadeau and Y. Bengio, “Inference for the generalization error,” in *Advances in neural information processing systems*, pp. 307–313, 2000.

- [112] R. Kohavi *et al.*, “A study of cross-validation and bootstrap for accuracy estimation and model selection,” in *Ijcai*, vol. 14, pp. 1137–1145, Montreal, Canada, 1995.
- [113] S. Yadav and S. Shukla, “Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification,” in *2016 IEEE 6th International conference on advanced computing (IACC)*, pp. 78–83, IEEE, 2016.
- [114] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.
- [115] M. Schuster and K. K. Paliwal, “Bidirectional recurrent neural networks,” *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.
- [116] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [117] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, 2009.
- [118] M. Xiao, H. Yan, J. Song, Y. Yang, and X. Yang, “Sleep stages classification based on heart rate variability and random forest,” *Biomedical Signal Processing and Control*, vol. 8, no. 6, pp. 624–633, 2013.
- [119] R. Casal, L. E. Di Persia, and G. Schlotthauer, “Classifying sleep–wake stages through recurrent neural networks using pulse oximetry signals,” *Biomedical Signal Processing and Control*, vol. 63, p. 102195, 2020.
- [120] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in neural information processing systems*, pp. 5998–6008, 2017.
- [121] S. Bai, J. Z. Kolter, and V. Koltun, “An empirical evaluation of generic convolutional and recurrent networks for sequence modeling,” *arXiv preprint arXiv:1803.01271*, 2018.
- [122] V. Dumoulin and F. Visin, “A guide to convolution arithmetic for deep learning,” *arXiv preprint arXiv:1603.07285*, 2016.
- [123] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 807–814, 2010.
- [124] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.
- [125] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, “Striving for simplicity: The all convolutional net,” *arXiv preprint arXiv:1412.6806*, 2014.
- [126] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *arXiv preprint arXiv:1502.03167*, 2015.
- [127] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, “Improving neural networks by preventing co-adaptation of feature detectors,” *arXiv preprint arXiv:1207.0580*, 2012.



- 
- [128] J. L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” *arXiv preprint arXiv:1607.06450*, 2016.
- [129] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [130] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.
- [131] J. J. Hopfield, “Neural networks and physical systems with emergent collective computational abilities,” *Proceedings of the national academy of sciences*, vol. 79, no. 8, pp. 2554–2558, 1982.

**Doctorado en Ingeniería**  
mención inteligencia computacional, señales y sistemas

Título de la obra:

# **Procesamiento de señales relacionadas con patologías del sueño**

Autor: Ramiro Casal

Lugar: Santa Fe, Argentina

Palabras Claves:

patologías del sueño, ,  
síndrome de apnea obstructiva del sueño,  
clasificación fases del sueño,  
máquinas de soporte vectorial, random forest,  
redes neuronales recurrentes,  
redes convolucionales temporales,  
mecanismos de atención.