



UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA



ELO308: MEMORIA DE TITULACIÓN

“Detección de etapas del sueño en señales EEG
utilizando medidas de complejidad entrópica”.

Profesora Guía: María José Escobar

Profesor Correferente: Matías Zañartu

Profesor Coordinador: Alejandro Suárez S.

Estudiante: Luis Padilla González ROL: 201160081-9

4 de noviembre de 2022

Índice

1. Capítulo 1: Introducción y objetivos	4
1.1. Introducción/Background	4
1.2. Objetivos	6
2. Capítulo 2: Estado del Arte	7
2.1. Pre-procesamiento.	8
2.2. Extracción de Características.	9
2.3. Selección de Características	10
2.4. Clasificación	10
3. Capítulo 3: Alternativas de solución	12
3.1. Alternativas	12
3.2. Criterios de Evaluación	12
3.3. Evaluación de Alternativas	13
3.3.1. EEGLAB [33]	13
3.3.2. MNE-Python [34]	14
3.3.3. BrainVision analyzer [35]	14
3.4. Alternativa Seleccionada	15
4. Capítulo 4: Metodología	16
4.1. Electroencefalograma (EEG)	16
4.2. Medidas entrópicas y no lineales	17
4.2.1. Entropía de Permutación	17
4.2.2. Entropía de Atención	17
4.2.3. Entropía Burbuja	18
4.2.4. Dimensión Fractal de Petrosian	19
4.2.5. Dimensión Fractal de Higuchi	19
4.2.6. Dimensión Fractal de Katz	19
4.3. Clasificadores de aprendizaje supervisado	20
4.3.1. Support Vector Machine (SVM):	20
4.3.2. K-Nearest Neighbors (KNN)	20
4.3.3. Multi Layer Perceptron Neural Network (MLP)	21
4.4. Métricas de evaluación	21

5. Capítulo 5: Experimentos y Resultados.	22
5.1. Datos y Diseño experimental	22
5.2. Método.	22
5.2.1. Características:	22
5.2.2. Clasificadores:	23
5.2.3. Validación Cruzada:	23
5.3. Validación cruzada de características no-lineales	24
5.3.1. Características entrópicas:	24
5.3.2. Características fractales:	25
5.3.3. Características Fractales y Entrópicas:	26
 6. Capítulo 7: Discusión y Conclusiones	 28
6.1. Discusión	28
6.1.1. Desempeño de clasificación	28
6.1.2. Desempeño de clasificación por etapas	28
6.1.3. Limitaciones y futuro trabajo	28
6.2. Conclusiones	29

Índice de figuras

1.	Flujo de trabajo esquema de clasificación automática de etapas de sueño (ASSC)	5
2.	Representación de etapas de sueño con su actividad cerebral asociada	8
3.	Esquema general clasificador RF	11
4.	Red neuronal artificial de tipo perceptrón	11
5.	Flujo de trabajo software MNE	15
6.	Estructura de una neurona	16
7.	a)Ilustración cerebro humano b)Diagrama de corteza cerebral	16
8.	Posicionamiento electrodos sistema 10/20 y 10/10	17
9.	Diferencia entre Entropía de Atención y de Shannon [38]	18
10.	Extracción de las 6 características del canal Fpz-Cz de la persona 'SC4092E0-PSG.edf'	23

Índice de cuadros

1.	Rango Frecuencias de Bandas y eventos EEG	7
2.	Criterio de clasificación AASM	7
3.	Tabla de Criterios de evaluación con sus ponderaciones.	13
4.	Sistema de puntajes	13
5.	Tabla de resultados de evaluación de las alternativas.	15
6.	Número de épocas por cada etapa para las 25 personas	22
7.	MLP: Matriz de confusión 12-fold Validación Cruzada para Entropías	24
8.	KNN: Matriz de confusión 12-fold Validación Cruzada para Entropías	24
9.	SVM: Matriz de confusión 12-fold Validación Cruzada para Entropías	25
10.	MLP: Matriz de confusión 12-fold Validación Cruzada Fractal	25
11.	KNN: Matriz de confusión 12-fold Validación Cruzada Fractal	25
12.	SVM: Matriz de confusión 12-fold Validación Cruzada Fractal	26
13.	MLP: Matriz de confusión 12-fold Validación Cruzada Fractal y Entrópica . . .	26
14.	KNN: Matriz de confusión 12-fold Validación Cruzada Fractal y Entrópica . . .	26
15.	SVM: Matriz de confusión 12-fold Validación Cruzada Fractal y Entrópica . . .	26
16.	Tiempos de cómputo para los 3 experimentos de validaciones cruzadas.	27

1. Capítulo 1: Introducción y objetivos

1.1. Introducción/Background

Dormir es la función primaria del cerebro y juega un rol esencial en las distintas actividades y capacidades de aprendizaje de una persona. Al pasar cerca de un tercio de nuestras vidas durmiendo, condiciones como insomnio, narcolepsia o apnea del sueño son frecuentes y pueden afectar severamente la salud mental y física.

La clasificación de las etapas del sueño es el estándar para analizar el dormir de una persona. Generalmente esto se consigue basado en grabaciones polisomnográficas (PSG), que son adquiridas de pacientes mientras duermen pasando la noche en clínica u hospital. Tradicionalmente, grabaciones PSG incluyen señales de electroencefalograma (EEG), electro-oculograma (EOG), electromiograma (EMG) y electrocardiograma (ECG), visualizadas y clasificadas por expertos usando referencias desarrolladas por Rechtschaffen & Kale en 1986 [1].

Cada grabación PSG se divide en segmentos (épocas) de 20 o 30 segundos, las cuales son subsecuentemente clasificadas como Wakefulness (W), movimiento o no-movimiento rápido de ojos (REM y NREM respectivamente). NREM luego se divide en etapas 1, 2, 3 y 4 (también referidas como S1, S2, S3 y S4) acorde a manuales establecidos por R&K y guías más recientes propuestas por la Academia Americana de Medicina del Sueño (AASM, por sus siglas en inglés) en 2007 [2]. El mayor cambio de los estándares AASM es combinar las etapas NREM S3 y S4 en una única etapa llamada N3 también referida como sueño de ondas lentas (SWS, por sus siglas en inglés).

Debido a que las grabaciones PSG incluyen señales desde múltiples canales y luego son visualmente examinados por una persona experta en el área, la clasificación de etapas de sueño es costosa, sensible a error humano y a veces una tarea tediosa o consumidora de tiempo. La clasificación de sueño de toda una noche suele tomar entre 2-4 horas. Además, en algunos estudios, la tasa de acuerdo entre especialistas para la clasificación de etapas de sueño es menor al 90 % [3].

Considerando estos desafíos, un modelo para clasificar etapas del sueño disminuiría la demanda de tiempo para clínicas, mejorar el porcentaje de precisión en la clasificación y mejorar el diagnóstico de trastornos del sueño.

Las señales EEG son una de las más importantes y frecuentemente usadas para analizar etapas del sueño, ya sea si la clasificación es manual o automática. Varios estudios de investigación han ocupado EEG de múltiples canales. Sin embargo, los equipos necesarios para EEG

de múltiples canales pone limitaciones en el movimiento de la persona, lo cual complica su portabilidad, uso y factibilidad de una evaluación de calidad de las etapas del sueño. Así, un esquema de clasificación automática de etapas del sueño (*Automatic Sleep Stage Classification*, ASSC) empleando distintos canales EEG puede ayudar a reducir las perturbaciones al dormir causadas por los cables de la grabación PSG u otros equipos.

En este trabajo de memoria se extraerán ciertas características (*features*) de la señal EEG para analizar las épocas registradas, sirviendo estas de entrada a un algoritmo de clasificación y así detectar etapas del sueño de forma automática con algún grado de precisión.

Para lograr lo anterior se utilizarán los canales frontal y occipital (Fpz-Cz y Pz-Oz, respectivamente) de señales EEG para clasificar las distintas etapas del sueño. Se trabajará con la base de datos en formato europeo (*european data format*, EDF) Physionet [8]. La implementación se llevará a cabo en lenguaje Python, extrayendo características no-lineales (entrópicas y fractales) de las épocas de 30 segundos para que luego sean ingresadas a la clasificación mediante métodos supervisados. El esquema de clasificación automática se observa en la Figura 1.

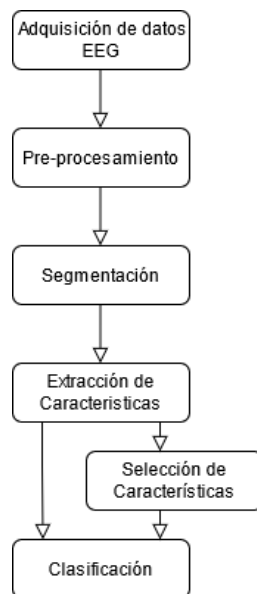


Figura 1: Flujo de trabajo esquema de clasificación automática de etapas de sueño (ASSC)

1.2. Objetivos

- Implementar un esquema de detección automática de etapas del sueño utilizando características no lineales junto a clasificadores no supervisados tales como; K-vecinos más cercanos, Red neuronal de tipo Perceptrón y Máquina de Vector Soporte.
- Realizar una comparación entre las predicciones de las etapas del sueño de cada clasificador con las anotaciones realizadas por especialistas.
- Evaluar mediante validaciones cruzadas cuáles características no-lineales de las señales polisomnográficas son más relevantes para la clasificación y detección de las distintas etapas del sueño.
- Medir y Comparar tiempos de procesamiento para las distintas características a extraer de las señales.

2. Capítulo 2: Estado del Arte

Las formas de onda de una señal EEG pueden ser diferenciadas en distintas bandas: δ , θ , α , β ([1],[2]). Estas bandas pueden proveer información valiosa para diagnosticar, monitorear y manipular características neurológicas. El Cuadro 1 muestra el rango en frecuencia de las señales EEG dividida en bandas junto a otros eventos relacionados a señales EEG del sueño como Spindles y Complejos-K.

Banda Frecuencia	Rango Frecuencias (Hz)
Delta δ	0.5 - 4
Theta θ	4 - 8
Alpha α	8 - 13
Beta β	13 - 30
Spindles	12 - 14
K-Complex	0.5 - 1.5

Cuadro 1: Rango Frecuencias de Bandas y eventos EEG

De acuerdo al manual AASM [2], cada una de las cinco etapas del sueño se clasifican según criterios mostrados en el Cuadro 2.

Etapas de Sueño	Criterio de Clasificación
W (Wake)	La mayoría de las épocas en esta etapa consisten en actividad α , o voltajes bajos de frecuencia mixta (2-7 Hz). Alta actividad EMG
N1	Actividad α ocupa mas del 50 % de la época, mientras actividad θ se hace evidente junto a movimiento circular lento de ojos.
N2	Característica típica de esta etapa es la aparición de Spindles o Complejos-K, junto con algunas actividades de alto voltaje (< 2 Hz)
N3	Etapa caracterizada por actividad δ detectada sobre un 20 % del largo de la época con altos voltajes ($> 75 \mu V$)
REM	Etapa bastante activa. Presenta actividades de bajos voltajes y frecuencias mixtas de ondas diente-sierra (2-7 Hz). Además hay movimiento rápidos de ojos con ausencia o reducida actividad EMG.

Cuadro 2: Criterio de clasificación AASM

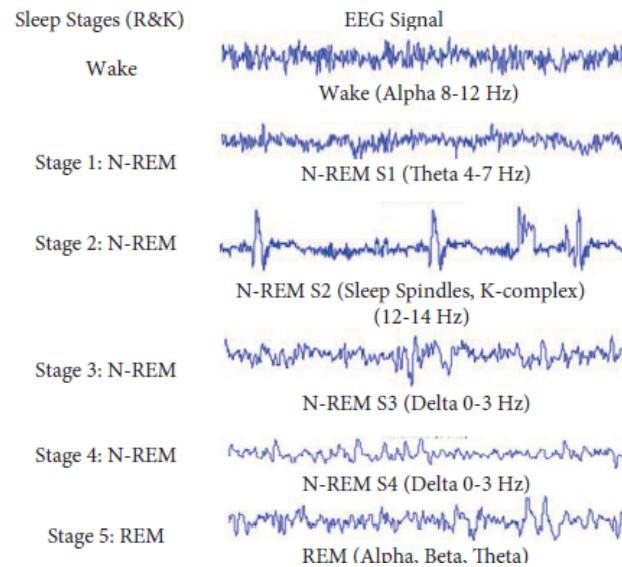


Figura 2: Representación de etapas de sueño con su actividad cerebral asociada

Así, una detección automática de las etapas según el Cuadro 2 debe seguir el flujo de trabajo mostrado en la Figura 1. Cada una de las etapas se discute brevemente a continuación.

2.1. Pre-procesamiento.

La presencia de artefactos o “señales no deseadas” puede llevar a mal interpretar resultados cuantitativos. Algunos artefactos usuales como movimientos oculares, tonos musculares mandibulares, interferencia 50/60 Hz de línea eléctrica o ruido proveniente del instrumento de medición son señales que dependiendo del análisis a utilizar, puede llevar a una mala interpretación de resultados. Así, las grabaciones EEG pueden tener una amplia variedad de artefactos, algunos de origen técnico y otros de origen fisiológico mezcladas con señales cerebrales [10],[11]. Por lo tanto, el uso de métodos de pre-procesamiento antes de la extracción de características es útil para analizar señales EEG sin perder información relevante.

En este contexto, existen varias técnicas de remoción de artefactos. Tales como:

- Filtrado Digital

Filtros digitales selectivos (pasa-bajo, pasa-alto, pasa-banda, elimina-banda), han sido ampliamente usados en la detección y remoción de artefactos. Estos filtros son poderosas herramientas significantes para minimizar artefactos y eliminar ciertas señales en la grabación EEG de frecuencias no deseadas. Filtros digitales de frecuencia selectiva pueden implementarse como filtros IIR o FIR. En electrofisiología, filtros IIR butterworth y elípticos son comunmente aplicados [4].

- Regresión:

Los métodos de regresión permiten aprender el comportamiento de una señal. Si un ruido de color aditivo distorsiona la señal EEG, se usa un método de regresión para aprender esta tendencia (*trend*) y luego modelando el color del ruido, puede ser sustraído de la señal original [12].

- Separación de fuente ciega (BSS):

Otra forma alternativa para remover artefactos son algoritmos de separación de fuente ciega (BSS, por sus siglas en inglés). El análisis de componentes independientes (ICA) [13], es un método bien conocido de BSS el cual produce fuentes cerebrales descorrelacionadas luego de atenuar artefactos. Mediante el cálculo de la energía en las fuentes de señal y aplicando un umbral, se eliminan las señales independientes con menor energía. Finalmente, con una transformada inversa las señales sin ruido se proyectan de regreso al dominio espacio-temporal.

Después de esta etapa de pre-procesamiento, basado en algún criterio mencionado, las grabaciones EEG se segmentan en épocas sucesivas de 30 segundos sin solapamiento, donde a cada una se le llevará a cabo el cómputo de alguna característica a extraer previo a ser clasificada.

2.2. Extracción de Características.

Estas características (*features*) son valores descriptivos que proyectan el contenido de una señal a una propiedad medible. Estas características deben ser lo más independiente posible para evitar recolectar información redundante. Extraer características discriminantes, informativas e independientes es un paso crucial antes de preparar un set de valores para la posterior etapa de clasificación [14]. Se dividen principalmente en cuatro categorías o dominios: Temporal, Frecuencia, Tiempo-Frecuencia y No-lineal. En este trabajo de memoria se utilizarán características no-lineales.

- **No-lineales:** Dado que las señales EEG exhiben características no lineales y dinámicas complejas [15], se utilizan medidas no-lineales como técnicas efectivas para procesar señales EEG. Los métodos no-lineales se dividen en dos categorías:

- Basado en complejidad y entropía:

Los métodos basados en entropía calculan la irregularidad e impureza de una señal en el dominio del tiempo [16]. En otras palabras, a medida que aumente la regularidad de patrones dentro de un largo dado de señal, menor será el valor de entropía de esa

señal. La medida de entropía segun Shannon es:

$$S(X) = - \sum_{i=1}^N P_i \ln(P_i) \quad (1)$$

Donde N es el número de muestras y P_i es la probabilidad de ocurrencia de la i -ésima muestra.

Varias medidas de entropía y complejidad tales como Renyi's Entropy, Sample Entropy, Permutation Entropy, Lempel-Ziv, Multi Scale Entropy (MSE) y Approximate Entropy (ApEn) se han introducido en la literatura [17].

- Basado en Fractales:

Se puede construir una señal de forma ruidosa aplicando repetidamente un operador sobre un elemento geométrico. En otras palabras, en vez de analizar la señal en el dominio del tiempo, estos métodos analizan el comportamiento de su trayectoria en el espacio de fase en términos de auto-similaridad. Existen métodos que determinan las irregularidades de la señal en el dominio del tiempo tales como dimensiones fractales de Katz, Petrosian e Higuchi. Resultados experimentales sugieren que las medidas de comportamiento en el espacio de fase son más precisas que las mismas en el dominio del tiempo. Sin embargo, eso impone un limite computacional más alto [19].

2.3. Selección de Características

El propósito de incluir este bloque es el de obtener un mínimo número de características sin redundancia produciendo mayor precisión. Dado que una grabación EEG de noche completa es larga, este paso es útil cuando distintas características se desean extraer.

2.4. Clasificación

Esta etapa utiliza clasificadores para las características extraídas y predecir la etapa de sueño correspondiente a cada época. Se les enseña a estos clasificadores construir limites lineales/no-lineales para separar vectores de características de diferentes clases. Algunos de los más populares en la literatura son:

- K-Nearest Neighbor (KNN): Es un clasificador local el cual es apropiado para distribuciones de datos multi-modales. En otras palabras, cuando las muestras de cierta clase de datos estan modularmente dispersadas en diferentes ubicaciones del espacio de características, incluso clasificadores flexibles o fuertes no pueden poner limites precisos alrededor de estas localidades. KNN asigna una etiqueta a patrones de entrada basado en el voto de la mayoría de sus k -muestras más cercanas [25].

- Support Vector Machine (SVM): SVM considera un margen alrededor de su hiperplano mientras otros clasificadores convencionales sólo intentan formar una cota, ya sea lineal o no-lineal, entre dos clases. De hecho, SVM intenta maximizar el ancho del margen al mismo tiempo que minimiza el error de clasificación de muestras dentro de esos márgenes [30].
- Random Forest (RF): Propuesto por Breiman [32], consiste en un conjunto de estructuras árbol. Cada árbol individual depende de un vector aleatorio de muestras el cual juega el rol de un clasificador individual. La principal diferencia de RF a otros clasificadores es que la alimentación de muestras de entrada a los árboles se realiza lo más aleatorio posible seguido de una selección bootstrap. Este proceso se repite muchas veces en orden de desensibilizar el efecto de muestras ruidosas o atípicas de la fase de entrenamiento. Finalmente, la salida es determinada mediante votación de las salidas de cada árbol. La estructura general de un clasificador RF se puede ver en la Figura 3.

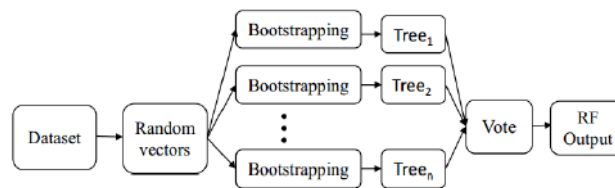


Figura 3: Esquema general clasificador RF

- Artificial Neural Network (ANN): ANN es un sistema de procesamiento artificial de información que comprende un número de procesadores interconectados llamadas neuronas. Se asignan funciones a estas neuronas siguiendo el comportamiento real de neuronas biológicas. La función más empleada de una neurona en diferentes ANN son sigmoides, hiperbólicas y lineales. La arquitectura de una ANN puede contener varias capas escondidas, donde cada una incluye varias neuronas. Un Perceptron multi capa (MLP) es un tipo de red neuronal artificial con alimentación hacia adelante (feedforward) Figura 4 [36].

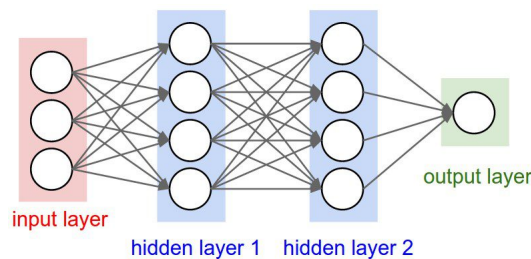


Figura 4: Red neuronal artificial de tipo perceptrón

3. Capítulo 3: Alternativas de solución

3.1. Alternativas

Como se vió en el capítulo anterior, para la solución e implementación de este trabajo de memoria, se requiere de un software capaz de entregar al usuario una amplia flexibilidad de manejo de datos EEG, así como tambien un buen grado de eficiencia computacional para la correcta implementación en el entorno de trabajo adecuado.

EEGLAB muestra ser una buena opción debido a su amigable y simple interfaz con el usuario para procesar datos. MNE-Python ofrece una gran cantidad de rutinas, librerías y paquetes óptimamente implementados. Por otro lado, BrainVision Analyzer combina tanto la implementación de algoritmos eficientes, como una poderosa interfaz gráfica con el usuario.

Para poder seleccionar adecuadamente la mejor alternativa que se ajuste a los requerimientos del trabajo de memoria, se procede a definir los criterios de evaluación con sus respectivas ponderaciones.

3.2. Criterios de Evaluación

Los criterios de evaluación serán los siguientes:

- Software de código abierto: Según el grado de disponibilidad y libertad de uso de la alternativa (OSS por sus siglas en inglés).
- Implementación algoritmos ICA: Variedad de algoritmos, eficiencia computacional e interfaz GUI para la etapa de pre-procesamiento del proyecto.
- Cómputo de complejidad entrópica: Variedades de índices entrópicos para series de tiempo, con el objeto de estimar transiciones entre estados concientes.
- Documentación y Soporte: En base a compatibilidad de sistemas operativos junto con calidad y cantidad de información bien documentada respecto al Software.

Además, cada uno de estos criterios se le asignará distintas ponderaciones según el grado de importancia que tenga en las distintas etapas del proyecto. Más específicamente, debido a que las tres alternativas poseen capacidad de pre-procesamiento, se le asignará mayor peso a la etapa del cómputo de complejidad entrópica, ya que rescata la esencia de este trabajo. Así mismo, lo segundo más importante es la libertad de uso y distribución del Software, ya que también se orienta esta memoria para su uso de código abierto y libre distribución.

Así, las ponderaciones relativas se muestran en el Cuadro 3:

Software de código abierto	25 %
Algoritmos ICA	20 %
Cómputo complejidad entrópica	35 %
Documentación/Soporte	20 %
Total	100 %

Cuadro 3: Tabla de Criterios de evaluación con sus ponderaciones.

3.3. Evaluación de Alternativas

A continuación, se define el sistema de puntajes relativos a que tan óptima es una alternativa respecto al criterio correspondiente. Lo puntajes serán:

Deficiente	Regular	Bueno	Muy bueno
0.1 - 0.25	0.26 - 0.50	0.51 - 0.75	0.76 - 1

Cuadro 4: Sistema de puntajes

Se procede a evaluar las distintas alternativas en base a los criterios de evaluación previamente definidos.

3.3.1 EEGLAB [33]

- Software de código abierto (0.5 puntos): EEGLAB es una herramienta de MATLAB distribuida bajo la licencia libre GNU GPL. Sin embargo, MATLAB es un producto de propiedad de *MathWorks* por lo que los usuario estan restringidos en el uso del servicio.
- Implementación algoritmos ICA (0.45 puntos): Función *runica.m* implementa algoritmo InfoMax ICA. Otros algoritmos como *FastICA* requieren instalación de toolbox adicional.
- Cómputo de complejidad entrópica(0.6 puntos): Existen implementaciones de distintas medidas de complejidad para series de tiempo disponibles en *FileExchange* de *MathWorks*. Entre ellas están; Shannon entropy, Approximate entropy, Sample entropy, Fuzzy entropy, Permutation Entropy y Conditional Entropy. Sin embargo, la capacidad de cómputo y eficiencia de estas implementaciones no está documentada.
- Documentación y Soporte(0.7 puntos): Soporte para sistemas operativos Windows, Linux y MAC OS, además de una muy buena documentación de las distintas dependencias de parte de *Swartz Center for Computational Neuroscience* (SCCN) de la Universidad de California.

3.3.2 MNE-Python [34]

- Software de código abierto (1 punto): Software bajo licencia permisiva BSD. Esta licencia permite a cualquier usuario reusar y redistribuir código, incluso en aplicaciones comerciales.
- Implementación algoritmos ICA(0.8 puntos): MNE-Python implementa la clase *mne.preprocessing.ICA* la cual soporta algoritmos FastICA y la versión extendida de algoritmo InfoMax.
- Cómputo de complejidad entrópica(0.95 puntos): MNE ofrece paquete *EntroPy*, el cual provee al usuario algoritmos eficientemente implementados como; Spectral Entropy, Permutation Entropy, Singular value decomposition Entropy, Approximate Entropy, Sample Entropy, así como también estimadores de complejidad via Lempel-Ziv.
- Documentación y Soporte(0.9 puntos): Soporte para variedad plataformas de como Linux, Unix, Windows, MACOS. Documentación abundante como parte de los principios de la comunidad Python.

3.3.3 BrainVision analyzer [35]

- Software de código abierto(0.1 puntos): *Brain Products* es una compañía privada limitada, teniendo que pagar por una licencia para el uso del software *Analyzer*.
- Implementación algoritmos ICA(0.9 puntos) : Tres versiones de InfoMax: extendida, restringida y polarizada, junto con una versión restringida y otra extendida del algoritmo *FastICA*. Además, soporta una interfaz gráfica para el usuario simple e intuitiva de usar.
- Cómputo de complejidad entrópica(0.1 puntos): No se reporta el uso de estimadores entrópicos para análisis de series de tiempo para los distintos canales de los datos EEG.
- Documentación y Soporte(0.45 puntos): Soporte único para Windows y una buena documentación, pero la gran mayoría de acceso privado debido a las políticas *GmbH*.

El siguiente cuadro resume los puntajes ponderados respectivos a cada alternativa en evaluación:

Criterios de Evaluación	EEGLAB	MNE-Python	Analyzer
Software de código abierto	0.5	1	0.1
Implementación algoritmos ICA	0.45	0.8	0.9
Cómputo de complejidad entrópica	0.6	0.95	0.1
Documentación y Soporte	0.7	0.9	0.45
Puntaje total	56.5 %	92.25 %	33 %

Cuadro 5: Tabla de resultados de evaluación de las alternativas.

3.4. Alternativa Seleccionada

Como se puede ver en el Cuadro 5, la alternativa seleccionada es MNE-Python. Este resultado se ve en gran parte influenciado por la capacidad de cómputo de complejidades algorítmicas que ofrece Python, junto con la característica de ser de plataforma libre. En la Figura 5 se muestran los distintos flujos de trabajo que ofrece el software notando que este trabajo de memoria comprende parte de la rama en azul.

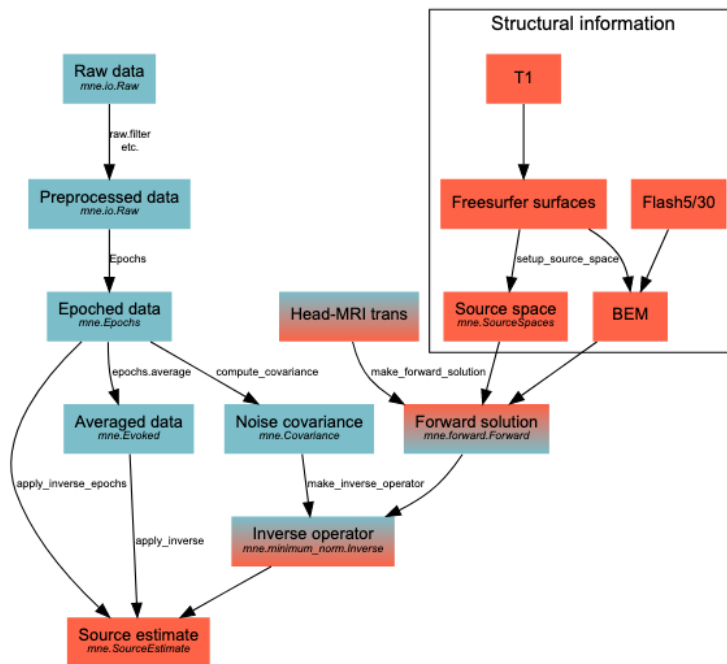


Figura 5: Flujo de trabajo software MNE

4. Capítulo 4: Metodología

4.1. Electroencefalograma (EEG)

El cerebro humano es considerado un sistema dinámico complejo que consiste en millones de neuronas interconectadas por axones y dendritas [9]. Estas neuronas son las responsables de comunicar información hacia y desde el cerebro. La Figura 6 muestra la estructura general de una neurona.

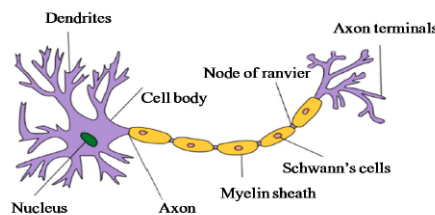


Figura 6: Estructura de una neurona

De acuerdo a [31], el cerebro se puede dividir anatómicamente en tres estructuras primarias: cerebro, cerebelo y tronco encefálico (Fig. 7a). El cerebro, el más largo de estas estructuras, está dividido en dos hemisferios con una superficie exterior llamada corteza cerebral. La corteza cerebral se organiza en cuatro lóbulos: frontal, parietal, temporal y lóbulos occipitales (Fig. 7b).

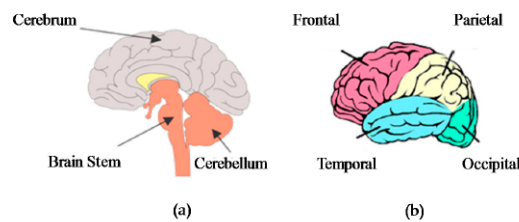


Figura 7: a) Ilustración cerebro humano b) Diagrama de corteza cerebral

Varios métodos se han desarrollado, para medir la señal de actividad generada por el cerebro humano, los cuales incluyen EEG, Magnetoencefalografía (MEG), *functional Magnetic Resonance Imaging* (f-MRI) y *Positron Emission Tomography* (PET). Entre ellas, EEG es una de las señales biomédicas con más fuerte significancia y un gran valor práctico para aplicaciones en clínica neurológica [5]. EEG es un método no invasivo para medir actividad eléctrica en la corteza cerebral y ha sido ampliamente usada desde su descubrimiento por Berger en 1929 [6]. Estas mediciones se logran colocando múltiples electrodos en el cuero cabelludo posicionados según estandar internacional SI 10/20 [7]. Este sistema se extendió al 10/10 con más electrodos

debido al desarrollo de hardware EEG multicanal. Como se muestra en la Figura 8 los círculos negros indican las posiciones en el sistema 10/20 y los grises las posiciones de electrodos en el sistema 10/10.

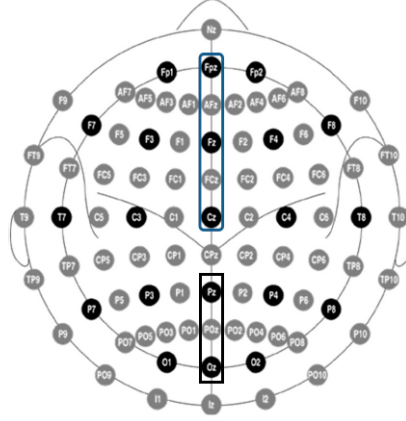


Figura 8: Posicionamiento electrodos sistema 10/20 y 10/10

Para una serie de tiempo, a mayor sea la probabilidad de ir generando nuevos patrones, mayor será su irregularidad en el tiempo. En este sentido la entropía es usada para evaluar el grado de aleatoriedad de la serie de tiempo. En consecuencia, es posible aplicar medidas entrópicas o no-lineales (fractales) para extraer características relevantes previo a ser clasificadas utilizando métodos de aprendizaje supervisado como K vecinos más cercanos (KNN), una red neuronal de tipo Perceptron multi-capas (MLP) o una Máquina de Vector Soporte (SVM).

4.2. Medidas entrópicas y no lineales

4.2.1 Entropía de Permutación La entropía de permutación de orden $n \geq 2$ se define como :

$$H(n) = - \sum p(\pi) \log_2 p(\pi) \quad (2)$$

Donde la suma corre sobre todas las $n!$ permutaciones π de orden n . Esto representa la información contenida en comparar n valores consecutivos de la serie de tiempo. Así, la entropía se acota $0 \leq H(n) \leq \log n!$ donde el límite inferior se alcanza para secuencias de valores crecientes o decrecientes, y el límite superior para un sistema completamente aleatorio (secuencia independiente e idénticamente distribuida) donde todas las posibles permutaciones aparecen con la misma probabilidad [37].

4.2.2 Entropía de Atención La Entropía de Atención es calculada en tres principales pasos: (1) Definir patrones claves; (2) Calcular los intervalos entre dos patrones claves adyacentes; (3) Calcular Entropía de Shannon de los intervalos.

Formalmente, dada una serie finita X , primero se define el patrón clave Ω . En segundo lugar, se calcula los intervalos $I^\Omega = \{v | v = j - i\}$ para cualquier sub-serie dada u_i, u_k y u_j de X los cuales satisfacen que u_i y u_j coinciden (*match*) en el patrón Ω , pero u_k no coincide en Ω para cualquier $i < k < j$. Finalmente, se calcula entropía de Shannon sobre I^Ω como entropía de Atención siendo este trabajo de memoria la primera vez en utilizar la medida en señales EEG para clasificación de etapas de sueño [38]. Los tres pasos anteriores se ejemplifican en la Figura 9 comparandolos con el cálculo de la entropía de Shannon.

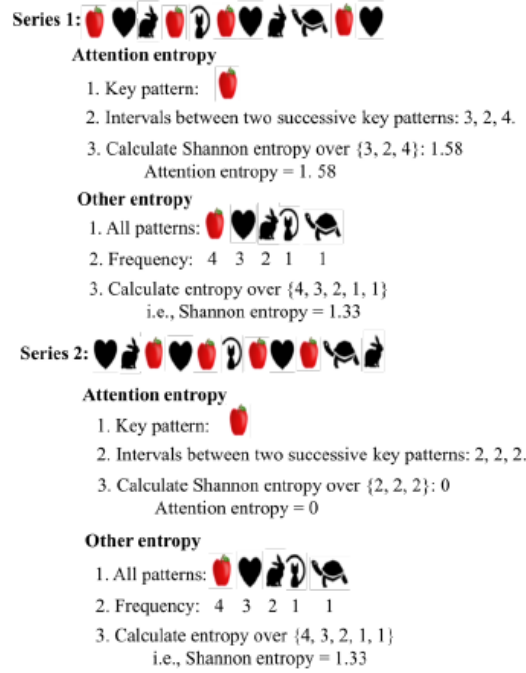


Figura 9: Diferencia entre Entropía de Atención y de Shannon [38]

4.2.3 Entropía Burbuja Esta entropía se basa estimando el esfuerzo requerido en el proceso de permutación en una dimension embebida $m > 1$. Los vectores que requieren un mismo esfuerzo se agrupan en nuevos grupos de vectores y se les calcula la Entropía Renyi. Se utiliza *bubble sort* para contar el número de *swaps* necesarios para ordenar el vector en orden acendente. Luego se calculan los *swaps* necesarios para ordenar cada vector y se calcula la entropía *Rényi* de esta distribución (H_{swaps}^m). La entropía Burbuja (bEn), viene dada por:

$$bEn = (H_{swaps}^{m+1} - H_{swaps}^m) / \log(m + 1/m - 1) \quad (3)$$

Donde el factor de normalización viene dado por la diferencia de máximas entropías [18].

4.2.4 Dimensión Fractal de Petrosian Para una serie de tiempo X , se define la dimensión fractal de Petrosian (PFD) como:

$$P = \frac{\log N}{\log N + \log \frac{N}{N+0.4N_\delta}} \quad (4)$$

Donde N es el largo de la serie X y N_δ es el número de cambios de signo en la derivada de la señal [19].

4.2.5 Dimensión Fractal de Higuchi La dimensión fractal D de Higuchi, se obtiene considerando un set finito de observaciones tomadas a intervalos regulares:

$$X(1), X(2), X(3), \dots, X(N).$$

A partir de esta serie, se contruye una nueva serie X_k^m definida como:

$$X_k^m; X(m), X(m+k), X(m+2k), \dots, X(m + [\frac{N-m}{k}]k)$$

con $(m = 1, 2, \dots, k)$, y donde $[\cdot]$ representa la notación de Gauss, esto es, el entero mas grande. m y k indican el tiempo inicial y el tiempo de intervalo respectivamente. Luego, para un intervalo de tiempo k , se obtienen k nuevos sets de series de tiempo. Así, Higuchi [39], define el largo de la curva asociada a cada serie de tiempo X_k^m como:

$$L_m(k) = \frac{1}{k} \left(\sum_{i=1}^{[\frac{N-m}{k}]} (X(m+ik) - X(m+(i-1)k)) \right) \left(\frac{N-1}{[\frac{N-m}{k}]k} \right) \quad (5)$$

Donde el término:

$$\frac{N-1}{[\frac{N-m}{k}]k}$$

Representa factor de normalización. Higuchi toma el valor promedio $\langle L(k) \rangle$ de las longitudes asociadas a las series de tiempo dadas por la ecuación L_m . Así, si el valor promedio sigue una ley de potencia $\langle L(k) \rangle \propto k^{-D}$, entonces la curva es fractal con dimensión D .

4.2.6 Dimensión Fractal de Katz El método de Katz calcula la suma y el promedio de las distancias euclidianas entre puntos sucesivos en una muestra (L y a , respectivamente). También se obtiene la distancia máxima entre el primer punto y cualquier otro punto de la muestra (d). Así la dimensión fractal de Katz (K_{fd}) de la muestra es:

$$K_{fd} = \frac{\log(\frac{L}{a})}{\log(\frac{d}{L}) + \log(\frac{L}{a})} \quad (6)$$

4.3. Clasificadores de aprendizaje supervisado

Para la clasificación se utilizarán tres clasificadores utilizados en el estado del arte: *Support Vector Machine*, *K-Nearest Neighbors* y *Multi Layer Perceptron Neural Network*, los cuales se detallan a continuación.

4.3.1 Support Vector Machine (SVM): En este trabajo se elige una Máquina de Vector Soporte (SVM, por sus siglas en inglés), por su simplicidad y robustez, así como también por ser ampliamente usada para la clasificación de etapas del sueño [23]. Una Máquina de Vector Soporte en general se aplica para clasificación binaria bajo técnicas de aprendizaje supervisado. El objetivo es encontrar un hiperplano óptimo con el set de datos de entrenamiento para luego validar el set de datos mediante el set de testeo [20]. En el caso de clasificación multiclase, siguiendo el enfoque de Nakamura, *et al* [21], se conectan varios clasificadores binarios en serie, y cada clasificador binario resuelve la clasificación de una clase contra las demás. Así, los siguientes clasificadores realizan lo mismo ignorando la clase previamente clasificada.

Como kernel de la máquina, se utiliza la función Gaussiana de base radial (“rbf”, por sus siglas en inglés), ya que ha mostrado mejor desempeño en comparación a kernels lineales o polinómicos en el contexto de clasificación de etapas del sueño [22]. Para la fase de entrenamiento es necesario ajustar el parámetro de regularización C y el coeficiente del kernel γ . Específicamente, C es un parámetro de compensación entre regularización y exactitud que influyen en el comportamiento de la selección del vector soporte y γ es un factor importante para controlar el kernel “rbf” en la transmisión de datos a un nuevo hiperespacio.

4.3.2 K-Nearest Neighbors (KNN) El método KNN [24] estima el valor de densidad de probabilidad en algún punto x basado en la distancia entre x y de su k -ésimo vecino más cercano. Una distancia larga indica que la densidad es pequeña y vice versa. Comparado con otros métodos, KNN tiene la ventaja de ser una técnica no paramétrica que evita completamente supuestos sobre densidades de probabilidad, y es usualmente considerada cuando no hay un previo conocimiento sobre la distribución de los datos.

Desde el punto de vista de clasificación, es fácil de interpretar, de cálculos rápidos, apto para set de datos extensos, resultados competitivos y predicciones útiles para set de datos con ruido. Es por lo anterior que se utiliza un clasificador KNN en este trabajo con métrica Manhattan, ya que obtiene mejores desempeños para la clasificación de etapas del sueño en comparación a métricas Euclidianas o de Chebyshev [27].

4.3.3 Multi Layer Perceptron Neural Network (MLP) Se utiliza un tipo de red neuronal rectificadora para la clasificación de tipo Perceptrón Multi-Capa (ver Figura 4). Para un set de entrada x , la función rectificadora se describe como:

$$f(x) = \max(0, x) \quad (7)$$

Una red rectificadora ha mostrado ser capaz de optimizar el desempeño sin necesidad de pre-entrenamientos no supervisados o de datos sin etiqueta. Neuronas rectificadoras tienen un mejor desempeño comparados a neuronas tipo sigmoide y tangente hiperbólicas [26].

4.4. Métricas de evaluación

Se evalúa el rendimiento de los modelos en base a una validación cruzada de 12 iteraciones. Las métricas de evaluación son: exactitud (*overall accuracy*) , *Precision*, Sensibilidad (*Recall*) y puntaje F_1 :

$$Accuracy = \frac{T_P + T_N}{T_P + F_P + T_N + F_N} \quad (8)$$

$$Precision = \frac{T_P}{T_P + F_P} \quad (9)$$

$$Recall = \frac{T_P}{T_P + F_N} \quad (10)$$

$$F_1 = \frac{T_P}{T_P + 0,5(F_P + F_N)} \quad (11)$$

Donde, por sus siglas en inglés, T_P es el numero de verdaderos positivos, T_N son verdaderos negativos, F_P falsos positivos y F_N falsos negativos.

5. Capítulo 5: Experimentos y Resultados.

5.1. Datos y Diseño experimental

La implementación se realiza en Python 3.8.2 y computador personal Dell con 2.3 GHz, procesador AMD Ryzen 7 y 16Gb de RAM. Se utiliza la base de datos Pyshionet con contiene del dormir de 197 grabaciones polysomnográficas (PSG) durante dos noches completas. Los archivos en formato ‘.edf’ contienen señales EEG, EOG, EMG y marcadores de eventos [8]. Los correspondientes hypnogramas (“*ground truth*”) fueron etiquetados manualmente por puntajes W, N1, N2, N3, REM o ‘?’ (desconocido) de técnicos bien entrenados acorde al manual de Rechtschaffen and Kales [1].

En este trabajo se utilizan los dos canales frontal y occipital; Fpz-Cz y Pz-Oz, respectivamente (ver Figura 8) de los datos PSG de 25 personas contenidos en la base de datos. La cantidad de épocas de cada etapa se puede ver en el Cuadro 6.

Etapa del Sueño	W	S1	S2	S3/4	REM	Total
Épocas	1258(5.36 %)	2239(9.53 %)	11802(50.25 %)	3107(13.23 %)	5079(21.63 %)	23485

Cuadro 6: Número de épocas por cada etapa para las 25 personas

Las grabaciones se cortan a los intervalos de sueño efectivo para disminuir el desbalance de clase. Luego cada canal se segmenta en épocas de 30 segundos a una frecuencia de muestreo de 100 Hz. Las características son extraídas con la ayuda de los paquetes en Python *Antropy* [28] y *EntropyHub* [29], los cuales proveen una variedad de algoritmos para calcular métricas de complejidad del estado del arte a series de tiempo tales como entropías: *Approximate*, *Fuzzy*, *Sample*, *Multi Scale*. Luego, sin más pre-procesamiento de los datos, para cada secuencia de la forma $x(i) : 1 \leq i \leq N$, con $N = 3000$ muestras, se calculan:

- Entropías de: **Permutación**, **Burbuja** y **Atención**.
- Dimensiones Fractales de: **Petrosian**, **Higuchi** y **Katz**

Las características son estandarizadas a media cero y varianza unitaria (Figura 10).

5.2. Método.

5.2.1 Características: Para los dos canales previamente mencionados, se utilizarán tres vectores de características extraídas como entradas a cada uno de tres clasificadores. Es decir:

- Vector Características Entrópicas de tamaño (23485,6) $\xrightarrow{\text{input}}$ KNN, MLP, SVM.

- Vector Características Fractales de tamaño $(23485,6) \xrightarrow{\text{input}}$ KNN, MLP, SVM.
- Vector Características Entrópicas y Fractales de tamaño $(23485,12) \xrightarrow{\text{input}}$ KNN, MLP, SVM.

Lo anterior entregando un total de nueve modelos para la clasificación, los cuales serán evaluados mediante Validación Cruzada.

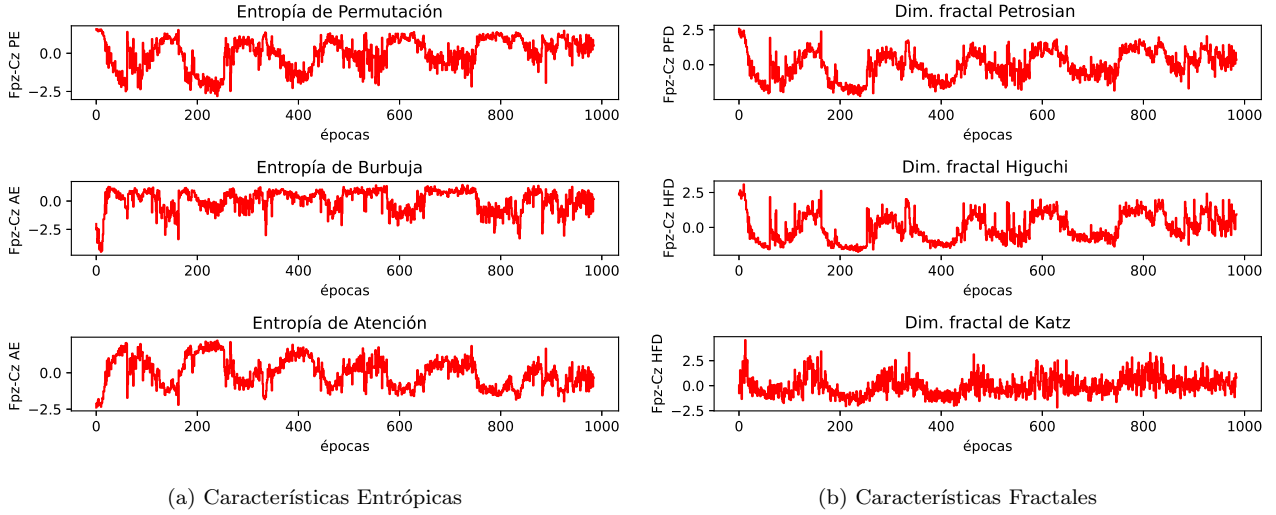


Figura 10: Extracción de las 6 características del canal Fpz-Cz de la persona 'SC4092E0-PSG.edf'

5.2.2 Clasificadores: Para la clasificación red neuronal de tipo Perceptrón MLP, se utilizan dos capas ocultas de tamaño 20 y 5 unidades ocultas respectivamente. Función de activación rectificadora '*relu*', solver de tipo gradiente descendiente estocástico '*sgd*' con una tasa de aprendizaje '*adaptativo*' y parámetro de regulación $\alpha = 10^{-4}$.

Para la etapa de clasificación, en el caso de K-vecinos más cercanos (KNN) se elige $K=100$ con métrica de distancia Manhattan. Para clasificador Máquina de Vector Soporte (SVM), se utiliza kernel función de base radial '*rbf*', $C = 0,8$, $\gamma = 1/(n_{features} \cdot X.var())$ donde $n_{features}$ indica el número de características extraídas para cada validación cruzada y $X.var()$ corresponde a la varianza de los datos de entrenamiento.

5.2.3 Validación Cruzada: En orden de observar el desempeño de las distintas características no-lineales extraídas para las 25 personas, se realizan 3 validaciones cruzadas de 12 iteraciones (*12-fold Cross Validation*). La elección de 12 es tal que para cada iteración se cuente con un mínimo de épocas correspondiente a 2 personas como set de testeo y el resto para el

entrenamiento. La validación cruzada se realizará para cada clasificador en cada uno de los siguientes casos: **1)** Extracción sólo de características entrópicas, **2)** Extracción sólo de características fractales y **3)** Extracción de características entrópicas y fractales.

El desempeño de cada clasificación se evalúa en términos de índices ampliamente usados: puntaje de exactitud total (Acc) dada por la ecuación 8 y puntaje macro-F1 dado por el promedio de todos los puntajes F1 de cada clase dados por la ecuación 11. También, el desempeño de cada etapa se evalúa en términos de *Recall* (Re) y *Precision* (Pre) dados por las ecuaciones 10 y 9 respectivamente.

5.3. Validación cruzada de características no-lineales

A continuación se muestran los resultados para cada uno de los tres experimentos previamente mencionados.

5.3.1 Características entrópicas: Extrayendo únicamente las tres características entrópicas: Permutación, Burbuja y Atención, los Cuadros 7, 8 y 9 muestran las matrices de confusión para las validaciones cruzadas en los clasificadores MLP, KNN y SVM respectivamente. El tiempo total de cómputo es de aproximadamente 29.2 minutos.

Acc= 65.32 % Macro-F1= 53.71 %								
	W	N1	N2	N3	REM	Pre %	Re %	F1 %
W	488	377	76	9	322	43.72	38.36	40.87
N1	282	565	509	2	881	33.55	25.23	28.8
N2	144	241	9447	830	1140	74.77	80.05	77.32
N3	22	22	1146	1891	26	69.09	60.86	64.72
REM	180	479	1457	5	2958	55.53	58.24	56.85

Cuadro 7: MLP: Matriz de confusión 12-fold Validación Cruzada para Entropías

Acc= 65.75 % Macro-F1= 50.05 %								
	W	N1	N2	N3	REM	Pre %	Re %	F1 %
W	395	227	100	0	550	39.46	31.05	34.76
N1	269	210	541	2	1217	30.17	9.38	14.31
N2	101	70	9781	693	1157	74.86	82.88	78.66
N3	18	17	1171	1851	50	72.65	59.58	65.46
REM	218	172	1473	2	3214	51.94	63.28	57.05

Cuadro 8: KNN: Matriz de confusión 12-fold Validación Cruzada para Entropías

Acc= 66.42 % Macro-F1= 50.78 %								
	W	N1	N2	N3	REM	Pre %	Re %	F1 %
W	388	308	90	3	483	36.06	30.50	33.05
N1	298	260	494	2	1185	28.14	11.61	16.44
N2	141	127	9808	571	1155	76.00	83.10	79.40
N3	26	18	1217	1802	44	75.78	58.00	65.71
REM	223	211	1295	0	3350	53.89	65.96	59.31

Cuadro 9: SVM: Matriz de confusión 12-fold Validación Cruzada para Entropías

5.3.2 Características fractales: Extrayendo únicamente las tres características fractales: Petrosian, Higuchi y Katz, los Cuadros 10, 11 y 12 muestran las matrices de confusión para las validaciones cruzadas en los clasificadores MLP, KNN y SVM respectivamente. El tiempo total de cómputo es de aproximadamente 12.5 minutos.

Acc= 67.52 % Macro-F1= 54.76 %								
	W	N1	N2	N3	REM	Pre %	Re %	F1 %
W	597	219	100	1	355	44.82	46.93	45.85
N1	382	377	564	12	904	43.53	16.84	24.28
N2	124	88	9699	797	1094	75.01	82.18	78.43
N3	43	10	1217	1769	14	68.59	56.94	62.22
REM	186	172	1297	0	3424	59.13	67.41	63.00

Cuadro 10: MLP: Matriz de confusión 12-fold Validación Cruzada Fractal

Acc= 67.1 % Macro-F1= 53.01 %								
	W	N1	N2	N3	REM	Pre %	Re %	F1 %
W	379	240	100	0	553	52.49	29.80	38.01
N1	178	390	576	3	1092	44.57	17.42	25.05
N2	49	92	9749	735	1177	74.58	82.61	78.39
N3	32	20	1287	1747	21	70.30	56.23	62.48
REM	84	133	1360	0	3502	55.19	68.95	61.31

Cuadro 11: KNN: Matriz de confusión 12-fold Validación Cruzada Fractal

Acc= 67.49 % Macro-F1= 54.05 %								
	W	N1	N2	N3	REM	Pre %	Re %	F1 %
W	509	257	85	1	420	49.18	40.02	44.13
N1	268	403	589	7	972	42.69	18.00	25.32
N2	76	114	9977	597	1038	73.93	84.54	78.88
N3	37	19	1445	1587	19	72.40	51.08	59.90
REM	145	151	1400	0	3383	58.00	66.60	62.01

Cuadro 12: SVM: Matriz de confusión 12-fold Validación Cruzada Fractal

5.3.3 Características Fractales y Entrópicas: Extrayendo las seis características en total, los cuadros 13, 14 y 15 muestran las matrices de confusión para las validaciones cruzadas en los clasificadores MLP, KNN y SVM respectivamente. El tiempo total de cómputo es de aproximadamente 31 minutos.

Acc= 69.85 % Macro-F1= 57.62 %								
	W	N1	N2	N3	REM	Pre %	Re %	F1 %
W	691	245	75	7	254	48.59	54.32	51.30
N1	473	336	573	15	842	28.19	15.00	19.59
N2	78	225	9812	651	1036	78.60	83.14	80.80
N3	20	17	941	2122	7	75.52	68.30	71.73
REM	160	369	1083	15	3452	61.74	67.97	64.71

Cuadro 13: MLP: Matriz de confusión 12-fold Validación Cruzada Fractal y Entrópica

Acc= 68.80 % Macro-F1= 54.56 %								
	W	N1	N2	N3	REM	Pre %	Re %	F1 %
W	417	222	97	0	536	54.58	32.78	40.96
N1	198	309	554	1	1177	42.04	13.80	20.78
N2	51	56	9816	680	1199	76.98	83.17	79.96
N3	22	17	1059	1979	30	74.37	63.69	68.62
REM	76	131	1225	1	3646	55.34	71.79	62.5

Cuadro 14: KNN: Matriz de confusión 12-fold Validación Cruzada Fractal y Entrópica

Acc= 70.18 % Macro-F1= 57.26 %								
	W	N1	N2	N3	REM	Pre %	Re %	F1 %
W	549	317	88	2	316	51.50	43.16	46.96
N1	291	388	544	2	1014	33.65	17.33	22.88
N2	72	172	10019	529	1010	77.95	84.89	81.27
N3	31	18	1057	1982	19	78.81	63.79	70.51
REM	123	258	1145	0	3553	60.10	69.95	64.65

Cuadro 15: SVM: Matriz de confusión 12-fold Validación Cruzada Fractal y Entrópica

Finalmente, los tiempos totales de cómputo para cada uno de los 3 experimentos se resumen en el cuadro 16.

Características	Tiempo [s]
Entrópicas	1749.6
Fractales	752.1
Entrópicas y Fractales	1862.4

Cuadro 16: Tiempos de cómputo para los 3 experimentos de validaciones cruzadas.

6. Capítulo 7: Discusión y Conclusiones

6.1. Discusión

6.1.1 Desempeño de clasificación

El modelo de clasificación que muestra el mejor desempeño general es el de extracción de las seis características no-lineales junto a un clasificador de tipo Máquina de Vector Soporte, con una exactitud del 70.18 % (Cuadro 15). Sin embargo en el mismo caso de todas las características extraídas, los demás clasificadores (MLP y KNN) no reportan resultados con diferencias significativas ($\approx 1\%$).

Por otro lado, considerando sólo las características fractales, se obtiene para los tres clasificadores exactitudes de $\approx 67\%$ lo cual, si bien, no son diferencias significativas, aún son buenos desempeños con un costo en tiempo de procesamiento relativamente bajo (Cuadro 16). En este sentido, la extracción de características fractales pueden ser útiles en aplicaciones de tiempo real sobre la detección de etapas del sueño.

6.1.2 Desempeño de clasificación por etapas

De todas las validaciones cruzadas para los tres experimentos se puede observar que la etapa N1 es con la que más se confunden los clasificadores, llegando a lo más a un puntaje $F_1 = 25,05\%$ para el modelo con características sólo fractales junto a clasificador KNN (N1-F1 Cuadro 11). Esto principalmente por la similitud de las características en frecuencia de los patrones EEG entre estas etapas. Ambas etapas están caracterizadas por voltajes bajos de frecuencia mixta (2-7 Hz) junto a actividad α . Otra principal razón de esto es debido al inherente desbalance de clases en este problema de clasificación dándole más peso a aquellas con mayor cantidad de épocas (Figura 6).

Por otro lado, las etapa con mejor desempeño en la clasificación resulta ser etapa N2 con $F_1 = 81,27\%$ en el caso de todas las características con clasificador SVM (N2-F1 Cuadro 15). Esto principalmente por el desbalance de clases mencionado donde N2 contiene el mayor porcentaje de etapas en una noche de sueño. Esto puede ser particularmente útil para identificar Spindles o Complejos-K presentes en esta etapa (Figura 2), los cuales juegan un rol importante en plasticidad sináptica o mecanismos de consolidación de memoria [40].

Finalmente, la posibilidad de que expertos hayan asignado erróneamente las etiquetas a esta etapa también puede contribuir a este problema.

6.1.3 Limitaciones y futuro trabajo

Es importante hacer notar que como modelos supervisados de aprendizaje aún es posible mejorar el desempeño para la clasificación en el futuro. La robustez de los modelos estudiados

aún debe ser testeada en distintas bases de datos junto a un detallado ajuste de los distintos parámetros e hiperparámetros para cada modelo. Además, sólo se han utilizado datos de personas saludables y aún queda evaluar rendimiento en pacientes con desordenes del sueño. El problema de clasificación de múltiples etapas para datos con clases desbalanceadas presenta el principal problema en la clasificación. En el futuro, una buena estrategia de sobre-muestreo previo a la clasificación puede mejorar significativamente los resultados obtenidos, así como también agregar las características asociadas a la señal de electro-oculograma (EOG).

6.2. Conclusiones

- En este trabajo se ha implementado un esquema de clasificación multiclase para las etapas del sueño utilizando los dos canales con más actividad relacionadas al sueño. Se ha mostrado el desempeño de distintos modelos de clasificación extrayendo características no-lineales del estado del arte de las señales EEG junto a clasificadores de aprendizaje supervisado.
- La clasificación de las tres características fractales de Petrosian, Higuchi y Katz muestran un buen desempeño en costo de tiempo computacional con puntajes de exactitud aceptables, comparado a la clasificación de características entrópicas.
- La extracción de las tres características fractales junto a las tres entrópicas de: Permutación, Burbuja y Atención, tiene un mejor desempeño al utilizar un clasificador de tipo Máquina de Vector Soporte en términos de exactitud así como también un buen puntaje relacionado a las etapas pertenecientes a la fase NREM.

Referencias

- [1] A. Rechtschaffen, A. Kales, “A Manual of Standardized Terminology, Techniques and Scoring System for Sleep Stages of Human Subjects.”, Los Angeles, CA: BIS/BRI, University of California; 1986.
 - [2] RB. Berry, R. Budhiraja, DJ. Gottlieb, D. Gozal , C. Iber , VK. Kapur, CL. Marcus , R. Mehra, S. Parthasarathy, SF. Quan , S. Redline, KP. Strohl , SL. Davidson Ward , MM. Tangredi “Rules for Scoring Respiratory Events in Sleep: Update of the 2007 AASM Manual for the Scoring of Sleep and Associated Events. Deliberations of the Sleep Apnea Definitions Task Force of the American Academy of Sleep Medicine.”, *Sleep Breath*, 19(4), pp. 1273-1277, Dec. 2015.
 - [3] F. Chapotot, G. Becq, “Automated sleep-wake staging combining robust feature extraction, artificial neural network classification, and flexible decision rules.”, *Int J Adapt Control Signal Process*, 24(5), pp. 409-423, May. 2010.
 - [4] Proakis, J. G., Manolakis, D. G., & Proakis, J. G. (1992). *Digital signal processing: Principles, algorithms, and applications*. New York: Macmillan
 - [5] Liu, Y.; Yan, L.; Zeng, B.; Wang, W. Automatic sleep stage scoring using Hilbert–Huang transform with BP neural network. In *Proceedings of the 4th International Conference on Bioinformatics and Biomedical Engineering (iCBBE)*, Chengdu, China, 18–20 June 2010; pp. 1–4.
 - [6] Patil, S.S.; Pawar, M.K. Quality advancement of EEG by wavelet denoising for biomedical analysis. In *Proceedings of the International Conference on Communication, Information & Computing Technology (ICCICT)*, Mumbai, India, 19–20 October 2012; pp. 1–6.
 - [7] Cic, M.; Šoda, J.; Bonkovic, M. Automatic classification of infant sleep based on instantaneous frequencies in a single-channel EEG signal. *Comput. Biol. Med.* 2013, 43, 2110–2117.
 - [8] B Kemp, AH Zwinderman, B Tuk, HAC Kamphuisen, JJL Oberyé. Analysis of a sleep-dependent neuronal feedback loop: the slow-wave microcontinuity of the EEG. *IEEE-BME* 47(9):1185-1194 (2000).
- Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., ... & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation* [Online]. 101 (23), pp. e215–e220.

- [9] Ahmad, R.F.; Malik, A.S.; Kamel, N.; Amin, H.; Zafar, R.; Qayyum, A.; Reza, F. Discriminating the different human brain states with EEG signals using fractal dimension: A nonlinear approach. In Proceedings of the IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA), Kuala Lumpur, Malaysia, 25–25 November 2014; pp. 1–5.
- [10] Widmann, A.; Schröger, E.; Maess, B. Digital filter design for electrophysiological data—A practical approach. *J. Neurosci. Methods* 2015, 250, 34–46.
- [11] Motamedi-Fakhr, S.; Moshrefi-Torbati, M.; Hill, M.; Hill, C.M.; White, P.R. Signal processing techniques applied to human sleep EEG signals—A review. *Biomed. Signal Process. Control* 2014, 10, 21–33.
- [12] R.J. Croft, J.S. Chandler, R.J. Barry, N.R. Cooper, A.R. Clarke “EOG correction: a comparison of four methods.”, *Psychophysiology*, 42(1), pp. 16- 24, Jan. 2005.
- [13] MB. Hamaneh, N. Chitravas, K. Kaiboriboon, S.D. Lhatoo, ; K.A. Loparo, “Automated Removal of EKG Artifact From EEG Data Using Independent Component Analysis and Continuous Wavelet Transformation.”, *IEEE Transactions on Biomedical Engineering*, 61(6), pp. 1634-1641, 2014.
- [14] C. M. Bishop, “Pattern Recognition and Machine Learning”, Springer- Verlag New York, 1st Edition, 2006.
- [15] C.F.V. Latchoumane, J. Jeong, “Quantification of brain macrostates using dynamical nonstationarity of physiological time series.”, *IEEE Trans. Biomed.Eng*, 58(4), pp. 1084-1093, Apr. 2011
- [16] M. Sabeti, S. Katebi, R. Boostani, “Entropy and complexity measures for EEG signal classification of schizophrenic and control participants.”, *Artificial Intelligence in Medicine*, 243, pp. 263-274, 2009.
- [17] Rodríguez-Sotelo, Jose L., Alejandro Osorio-Forero, Alejandro Jiménez-Rodríguez, David Cuesta-Frau, Eva Cirugeda-Roldán, and Diego Peluffo. 2014. “Automatic Sleep Stages Classification Using EEG Entropy Features and Unsupervised Pattern Analysis Techniques.” *Entropy* 16, no. 12: 6573-6589.
- [18] G. Manis, M. Aktaruzzaman and R. Sassi, “Bubble Entropy: An Entropy Almost Free of Parameters” in *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 11, pp. 2711-2718, Nov. 2017, doi: 10.1109/TBME.2017.2664105.

- [19] R. Esteller, G. Vachtsevanos, J. Echauz and B. Litt, “A comparison of Waveform Fractal Dimension Algorithms”, IEEE transactions on circuits and systems , 48 (12), pp. 177-183, Feb. 2001.
- [20] Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273–297.
- [21] Nakamura T, Adjei T, Alqurashi Y, Looney D, Morrell M J and Mandic D P 2017 Complexity science for sleep stage classification from EEG Int. Joint Conf. on Neural Networks pp 4387–94
- [22] Zhihong Cui and Xiangwei Zheng. 2017. A sleep stage classification algorithm based on radial basis function networks. In Proceedings of the 2nd International Conference on Crowd Science and Engineering (ICCSE'17). Association for Computing Machinery, New York, NY, USA, 56–60. <https://doi.org/10.1145/3126973.3126976>
- [23] Basha, A.J., Balaji, B.S., Poornima, S. et al. Support vector machine and simple recurrent network based automatic sleep stage classification of fuzzy kernel. J Ambient Intell Human Comput 12, 6189–6197 (2021). <https://doi.org/10.1007/s12652-020-02188-4>
- [24] Zhao, Puning & Lai, Lifeng. (2020). Analysis of KNN Density Estimation. <https://doi.org/10.48550/arXiv.2010.00438>
- [25] S. Gunes, K. Polat, S. Yosunkaya, “Efficient Sleep Stage Recognition System Based on EEG Signal Using K-means Clustering Based Feature Weighting.”, Expert Systems with Applications, 37, pp. 7922-7928, 2010.
- [26] 18. X. Glorot, A. Bordes, and Y. Bengio, “Deep Sparse Rectifier Neural Networks”, Aistats, vol. 15, pp. 315–323, 2011.
- [27] Qureshi, Shahnawaz & Karrila, Seppo & Vanichayobon, Sirirut. (2018). Human Sleep Scoring Based on K-Nearest Neighbors. TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES. 26. 2803-2819. 10.3906/elk-1805-12.
- [28] Antropy: open-source Python package for computing entropy metrics of EEG time-series. Raphael Vallat, <https://github.com/raphaelvallat/antropy>.
- [29] Matthew W. Flood and Bernd Grimm, EntropyHub: An Open-Source Toolkit for Entropic Time Series Analysis, PLoS One 16(11):e0259448 (2021), DOI: 10.1371/journal.pone.0259448

- [30] C. Cortes, V. Vapnik, "Support-vector networks.", *Machine Learning*, 20(3), pp. 273-297, Sept. 1995
- [31] Khalifa, W.; Salem, A.; Roushdy, M.; Revett, K. A survey of EEG based user authentication schemes. In *Proceedings of the 8th International Conference on Informatics and Systems (INFOS)*, Cairo, Egypt, 14–16 May 2012.
- [32] L. Breiman, "Random forests", *Machine Learning*, 45 (1), pp. 5-32, 2001.
- [33] A. Delorme & S. Makeig (2004) EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics.
- [34] A. Gramfort, M. Luessi, E. Larson, D. Engemann, D. Strohmeier, C. Brodbeck, R. Goj, M. Jas, T. Brooks, L. Parkkonen, M. Hämäläinen, *MEG and EEG data analysis with MNE-Python*, *Frontiers in Neuroscience*, Volume 7, 2013, ISSN 1662-453X.
- [35] BrainVision Analyzer (Version 2.2.0) [Computer software]. (2019). Gilching, Germany: Brain Products GmbH.
- [36] Nurma, Intan & Rosadi, Rudi & Purwani, Sri & Suryani, Mira. (2018). Multi-Layer Perceptron for Sleep Stage Classification. *Journal of Physics: Conference Series*. 1028. 012212. 10.1088/1742-6596/1028/1/012212.
- [37] Bandt C, Pompe B. Permutation entropy: a natural complexity measure for time series. *Phys Rev Lett*. 2002 Apr 29;88(17):174102. doi: 10.1103/PhysRevLett.88.174102. Epub 2002 Apr 11. PMID: 12005759.
- [38] Yang, Jiawei & Choudhary, G. & Rahardja, Susanto. (2020). Classification of Interbeat Interval Time-series Using Attention Entropy. *IEEE Transactions on Affective Computing*. 1. 10.1109/TAFFC.2020.3031004.
- [39] T. Higuchi, Approach to an irregular time series on the basis of the fractal theory, *Physica D: Nonlinear Phenomena*, Volume 31, Issue 2, 1988, Pages 277-283, ISSN 0167-2789, [https://doi.org/10.1016/0167-2789\(88\)90081-4](https://doi.org/10.1016/0167-2789(88)90081-4).
- [40] Antony JW, Schönauer M, Staresina BP, Cairney SA. Sleep Spindles and Memory Reprocessing. *Trends Neurosci*. 2019 Jan;42(1):1-3.