

UNIVERSIDAD DEL VALLE DE GUATEMALA  
Facultad de Ingeniería



**Aplicación de ayuda para pilotos a detección temprana de fatiga por medio de modelos de *Machine Learning*: Detección por medio de ritmo cardíaco**

Trabajo de graduación presentado por Julio Roberto Herrera Sabán  
para optar al grado académico de Licenciado en Ingeniería en Ciencia de  
la Computación y Tecnologías de la Información

Guatemala,

2024



UNIVERSIDAD DEL VALLE DE GUATEMALA  
Facultad de Ingeniería



**Aplicación de ayuda para pilotos a detección temprana de fatiga por medio de modelos de *Machine Learning*: Detección por medio de ritmo cardíaco**

Trabajo de graduación presentado por Julio Roberto Herrera Sabán  
para optar al grado académico de Licenciado en Ingeniería en Ciencia de  
la Computación y Tecnologías de la Información

Guatemala,

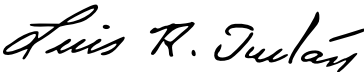
2024




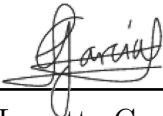
Vo.Bo.:

(f)   
M.A. Luis Roberto Furlán Collver

Tribunal Examinador:

(f)   
M.A. Luis Roberto Furlán Collver

(f)   
MSc. Douglas Leonel Barrios Gonzalez

(f)   
MBIA Lynette García Pérez

Fecha de aprobación: Guatemala, 25 de noviembre de 2024.



Con orgullo presento este trabajo de graduación como la culminación de mi etapa universitaria. Este proyecto refleja mi capacidad de aprendizaje, investigación y aplicación de los conocimientos adquiridos a lo largo de estos años en la Universidad del Valle de Guatemala.

Este Trabajo de Graduación es parte de un esfuerzo colaborativo con compañeros de la carrera, quienes, junto conmigo, buscamos desarrollar un sistema que pueda contribuir a la sociedad a través de la tecnología. Como alguien que ha tenido que conducir largas distancias, sé lo peligroso que es sentirse fatigado al volante, llegando incluso a cerrar los ojos con el vehículo en movimiento. También reconozco lo difícil que es aceptar el estado de cansancio y, aun así, forzarse a continuar. Crear un sistema de detección de somnolencia para conductores es una idea que ha sido explorada por muchos, pero no existe un estándar consolidado. Por ello, contribuir a este campo me pareció una oportunidad valiosa.

Personalmente, siempre he tenido interés en el tema del sueño, los patrones de descanso y los cambios fisiológicos asociados, por lo que seleccionar el ritmo cardíaco como variable para la detección de somnolencia fue un desafío que me resultó atractivo. Desde el inicio, el proyecto estaba pensado para integrarse con otros modelos de predicción, ya que sabía que el ritmo cardíaco influye en las fases del sueño, aunque no tenía claro en qué medida. Durante el desarrollo del proyecto, los resultados no fueron lo suficientemente precisos como para crear un sistema funcional y utilizable, lo que me llevó a revisar continuamente el enfoque, buscar más datos y tratar de perfeccionar el método de predicción. Sin embargo, al final, recordé que este proyecto no estaba diseñado para funcionar de manera aislada, y que lo más importante era contribuir al avance de una tecnología que aún no está completamente establecida en la sociedad.

La finalización de este trabajo si bien representa a mi persona, no hubiera sido posible sin la colaboración de mis compañeros de universidad, al Ing. Douglas y al Ing. Furlán por su asesoría durante el proyecto, a mi familia que me llenó de ánimos durante el transcurso de la carrera, a mis amigos de la universidad por hacer de esta una etapa de mi vida muy amena y a la Fundación Juan Bautista Guitérrez por darme la oportunidad de estudiar en la Universidad del Valle de Guatemala.

Julio Roberto Herrera Sabán.



<b>Prefacio</b>	<b>v</b>
<b>Lista de figuras</b>	<b>x</b>
<b>Lista de tablas</b>	<b>xi</b>
<b>Resumen</b>	<b>xiii</b>
<b>Abstract</b>	<b>xv</b>
<b>1. Introducción</b>	<b>1</b>
<b>2. Objetivos</b>	<b>3</b>
2.1. Objetivo general	3
2.2. Objetivos específicos	3
<b>3. Justificación</b>	<b>5</b>
<b>4. Marco teórico</b>	<b>7</b>
4.1. Métodos de detección de fatiga en conductores	7
4.1.1. Detección por observación ocular, PERCLOS y otros métodos	7
4.1.2. Detección por frecuencia cardíaca (HR) y variabilidad (HRV)	7
4.1.3. Implementación de sensores en vehículos	8
4.2. Personalización de los modelos de detección	8
4.3. Tecnologías de sensores para monitoreo de fatiga	9
4.3.1. Tipos de sensores para medición de HR	9
4.3.2. Sensores PPG en <i>wearables</i> y volantes	9
4.4. Herramientas de simulación y experimentos controlados	10
4.5. Latencia de sueño y prueba de mantenimiento de la vigilia (MWT)	10
4.6. Fases del sueño y su relación con la fatiga	10
4.6.1. Clasificación de las fases del sueño	10
4.6.2. Actividad del sistema nervioso autónomo durante el sueño	11
4.7. Factores de salud y estilo de vida en conductores	12
<b>5. Antecedentes</b>	<b>13</b>
<b>6. Alcance</b>	<b>15</b>

<b>7. Herramientas y modelos</b>	<b>17</b>
7.1. <i>Datasets</i>	17
7.1.1. Sleep Heart Health Study (SHHS)	17
7.1.2. PMData - Datos de frecuencia cardíaca y sueño obtenidos a través del <i>smart-watch</i> Fitbit	18
7.2. Modelos	19
7.2.1. LSTM	19
7.2.2. AutoGluon	20
7.3. Ética y seguridad	20
7.3.1. Selección ética de variables	20
7.3.2. Implicaciones legales y responsabilidad del uso de datos	21
7.3.3. Consideraciones éticas en la implementación de la Prueba de Mantenimiento de la Vigilia	21
7.3.4. Limitaciones éticas y propuestas para estudios futuros	21
<b>8. Metodología</b>	<b>23</b>
8.1. Procesamiento de los <i>datasets</i>	23
8.1.1. Procesamiento del <i>dataset</i> PMData	23
8.1.2. Preprocesamiento del <i>dataset</i> SHHS	30
8.1.3. Procesamiento en conjunto de PMData y SHHS	44
8.2. Modelo LSTM	47
8.2.1. Preparación de datos para LSTM	47
8.2.2. Estructura del modelo LSTM	48
8.2.3. Evaluación del modelo LSTM	50
8.3. Modelo tabular de AutoGluon	51
8.3.1. Preparación de datos para el modelo tabular de AutoGluon y análisis PCA	51
8.3.2. Entrenamiento del modelo tabular de AutoGluon	55
8.3.3. Evaluación del modelo tabular de AutoGluon	56
8.4. Modelo forecast de AutoGluon	60
8.4.1. Preparación de datos para el modelo forecast de AutoGluon	60
8.4.2. Entrenamiento del modelo forecast de AutoGluon	60
8.4.3. Evaluación del modelo forecast de AutoGluon	62
8.5. Discusión sobre los resultados de los modelos	63
8.6. Sistema en tiempo real	65
8.6.1. Hardware	65
8.6.2. Software para la obtención del ritmo cardíaco	65
8.6.3. Servidor que ejecuta el modelo	67
<b>9. Conclusiones</b>	<b>71</b>
<b>10.Recomendaciones</b>	<b>73</b>
<b>11.Bibliografía</b>	<b>75</b>
<b>12.Glosario</b>	<b>79</b>

---

## Lista de figuras

---

8.1. Histograma de la columna <code>BPM_count</code> del <i>dataset</i> <code>PMData</code> .	24
8.2. Histograma de la columna <code>seconds_passed</code> del <i>dataset</i> <code>PMData</code> .	25
8.3. Histograma de la columna <code>day_seconds</code> del <i>dataset</i> <code>PMData</code> .	25
8.4. Conteo de cada categoría de sueño donde 0 es despierto, 1 es sueño ligero y 2 es sueño profundo, para el <i>dataset</i> <code>PMData</code> .	25
8.5. Histograma de la columna <code>age</code> del <i>dataset</i> <code>PMData</code> .	26
8.6. Histograma de la columna <code>BPM</code> del <i>dataset</i> <code>PMData</code> .	26
8.7. Histogramas de las columnas <code>fatigue</code> , <code>sleep_quality</code> y <code>stress</code> del <i>dataset</i> <code>PMData</code> .	27
8.8. Mapa de calor ( <i>heat map</i> ) de las correlaciones de las variables del <i>dataset</i> <code>PMData</code> .	27
8.9. Gráfica de ritmo cardíaco con tendencia a bajar.	28
8.10. Gráfica de ritmo cardíaco constante.	28
8.11. Gráfica de ritmo cardíaco con tendencia a subir.	29
8.12. Gráfica de ritmo cardíaco fluctuante 1.	29
8.13. Gráfica de ritmo cardíaco fluctuante 2.	30
8.14. Gráfica de ECG del participante 17 de SHHS, sin filtro.	31
8.15. Gráfica de ECG del participante 17 de SHHS, sin filtro y con acercamiento.	32
8.16. Gráfica de ECG del participante 10 de SHHS con filtro de paso alto y filtro de paso bajo.	32
8.17. Gráfica de ECG del participante 13 de SHHS con filtro de paso alto y filtro de paso bajo.	33
8.18. Gráfica de ECG del participante 17 de SHHS, derivada.	33
8.19. Gráfica de ECG del participante 17 de SHHS, derivada y con acercamiento.	33
8.20. Gráfica de ECG del participante 17 de SHHS, elevada al cuadrado.	34
8.21. Gráfica de ECG del participante 17 de SHHS, elevada al cuadrado y con acercamiento.	34
8.22. Gráfica de ECG del participante 17 de SHHS, integrada.	35
8.23. Gráfica de ECG del participante 17 de SHHS, integrada y con acercamiento.	35
8.24. Gráfica de ECG del participante 17 de SHHS, detección de picos.	35
8.25. Gráfica de ECG del participante 17 de SHHS, detección de picos con acercamiento.	36
8.26. Gráfica de BPM del participante 17 de SHHS.	36
8.27. Gráfica de BPM del participante 17 de SHHS, con acercamiento.	37
8.28. Gráfica de ECG del participante 7 de SHHS, sin filtro.	37
8.29. Gráfica de BPM del participante 7 de SHHS.	38
8.30. Gráfica de ECG del participante 20 de SHHS, sin filtro.	38
8.31. Gráfica de BPM del participante 20 de SHHS.	38
8.32. Histograma de la columna <code>age</code> del <i>dataset</i> <code>SHHS</code> .	39
8.33. Histograma de la columna <code>BPM_count</code> del <i>dataset</i> <code>SHHS</code> .	39
8.34. Histograma de la columna <code>time</code> del <i>dataset</i> <code>SHHS</code> .	39

8.35. Conteo de la columna <code>SleepStage</code> del <i>dataset</i> SHHS.	40
8.36. Conteo de la columna <code>BPM</code> del <i>dataset</i> SHHS.	40
8.37. Conteo de la columna <code>fatigue</code> del <i>dataset</i> SHHS.	41
8.38. Conteo de la columna <code>sleep_quality</code> del <i>dataset</i> SHHS.	41
8.39. Conteo de la columna <code>stress</code> del <i>dataset</i> SHHS.	41
8.40. Gráfica de ritmo cardíaco con picos a 0.	42
8.41. Gráfica de ritmo cardíaco en escenario no relacionado con somnolencia en conductores.	42
8.42. Gráfica de ritmo cardíaco con anomalías en los valores finales.	43
8.43. Gráfica de ritmo cardíaco de SHHS con acercamiento.	43
8.44. Gráfica de ritmo cardíaco de una sesión correcta de SHHS.	44
8.45. Distribución de las fases de sueño <code>SleepStage</code> en el <i>dataset</i> completo.	44
8.46. Gráfica de caja <i>box plot</i> de la distribución de BPM por fase de sueño ( <code>SleepStage</code> ).	46
8.47. Mapa de calor ( <i>heat map</i> ) de las correlaciones de las variables del <i>dataset</i> total.	47
8.48. Arquitectura del modelo LSTM.	49
8.49. Matriz de confusión de las predicciones del modelo LSTM.	51
8.50. Gráfica de barras de los porcentajes de varianza explicada por cada componente principal (PCA).	52
8.51. Gráfica de barras de las cargas máximas absolutas por variable en al menos un componente principal (ordenada).	53
8.52. Matriz de confusión de las predicciones del modelo tabular de AutoGluon.	57
8.53. Gráfica de la importancia, desviación estándar y p-value de las características para el modelo tabular de AutoGluon.	58
8.54. Gráfica de la carga absoluta de cada una de las variables en los componentes principales PCA_4 y PCA_5 (ordenada).	60
8.55. Matriz de confusión de las predicciones del modelo de forecast de AutoGluon.	63
8.56. Gráficas de comparación para precisión, recall y F1-score entre los modelos utilizados.	64
8.57. Pantalla de inicio del cliente para conectar con el <i>smartwatch</i> Xiaomi Mi Band 5.	66
8.58. Pantalla que muestra el ritmo cardíaco y sin alerta.	66
8.59. Pantalla que muestra el ritmo cardíaco y alerta de somnolencia	67
8.60. Gráfica de los pesos que se aplican en la función de predicción.	68

---

## Lista de tablas

---

8.1. Métricas de clasificación para las distintas clases del modelo LSTM.	51
8.2. Porcentaje de varianza explicada por cada componente principal (PCA).	53
8.3. Cargas máximas absolutas por variable en al menos un componente principal.	54
8.4. Especificaciones del sistema utilizado para el entrenamiento de AutoGluon.	56
8.5. Resultados del entrenamiento del modelo tabular de AutoGluon.	56
8.6. Resultados de las métricas de evaluación del modelo tabular de AutoGluon.	56
8.7. Reporte de clasificación del modelo tabular de AutoGluon.	56
8.8. Importancia de características ( <i>leaderboard</i> ) para el modelo tabular de AutoGluon.	58
8.9. Carga de cada una de las variables en los componentes principales PCA_4 y PCA_5.	59
8.10. Resultados del entrenamiento del modelo forecast de AutoGluon.	62
8.11. Reporte de clasificación del modelo forecast de AutoGluon.	62
8.12. Comparación de métricas de clasificación y soporte para los tres modelos utilizados.	64
8.13. Frecuencia de toma de datos de la Mi Band 5.	65



Este proyecto tiene como objetivo desarrollar un sistema para la detección de somnolencia o fatiga basado en el monitoreo de la frecuencia cardíaca (BPM) y otros parámetros fisiológicos simples relacionados con la fatiga. Se emplearon técnicas de IA y aprendizaje automático para entrenar tres modelos distintos: un modelo *Long Short-Term Memory* (LSTM) utilizando TensorFlow, un modelo tabular de *AutoGluon* y un modelo forecast de *AutoGluon*. El entrenamiento y la validación se realizaron utilizando datos preexistentes de estudios del sueño, específicamente los conjuntos de datos *Sleep Heart Health Study* (SHHS) y *PMData*. Aunque los modelos desarrollados no están diseñados para ser implementados de forma independiente, este estudio proporciona resultados sobre el uso potencial de estos datos para sistemas más robustos..

Utilizando el 20% de los datos para evaluación, el modelo con mejor desempeño fue el **WeightedEnsemble Forecast** de *AutoGluon*, con un F1-score de 0.92 para la clase 0 (Despierto), 0.58 para la clase 1 (Sueño ligero) y 0.96 para la clase 2 (Sueño profundo). Tanto el **WeightedEnsemble Tabular** de *AutoGluon* como el modelo LSTM mostraron resultados bajos en las predicciones de la clase 1, con un F1-score de 0.43 para el *Tabular* y 0.07 para el LSTM. Los modelos fueron probados en un sistema de monitoreo en tiempo real de la frecuencia cardíaca, donde no lograron predecir la somnolencia con antelación o mientras ocurría. Los resultados sugieren que la frecuencia cardíaca, aunque limitada como indicador único, puede aportar información complementaria para la detección de fatiga.



This project aims to develop a system for detecting drowsiness or fatigue based on heart rate (BPM) monitoring and other simple physiological parameters related to fatigue. Artificial intelligence and machine learning techniques were employed to train three different models: a Long Short-Term Memory (LSTM) model using TensorFlow, a Tabular model from AutoGluon, and a forecast model from AutoGluon. The training and validation processes were conducted using pre-existing data from sleep studies, specifically the Sleep Heart Health Study (SHHS) and PMData datasets. Although the developed models are not designed to be deployed independently, this study provides insights into the potential use of these data for more robust systems.

Using 20% of the data for evaluation, the best-performing model was the **WeightedEnsemble** Forecast from AutoGluon, achieving an F1-score of 0.92 for class 0 (Awake), 0.58 for class 1 (Light Sleep), and 0.96 for class 2 (Deep Sleep). Both the **WeightedEnsemble** Tabular from AutoGluon and the LSTM model showed lower performance in predicting class 1, with an F1-score of 0.43 for the Tabular model and 0.07 for the LSTM. The models were tested in a real-time heart rate monitoring system, where they failed to predict drowsiness in advance or as it occurred. The results suggest that heart rate, while limited as a standalone indicator, can provide complementary information for fatigue detection.



# CAPÍTULO 1

---

## Introducción

---

La fatiga es una causa importante de accidentes vehiculares, esta afecta tanto a conductores de vehículos particulares como a los pilotos de vehículos pesados, que conducen largas rutas y durante periodos prolongados, usualmente por la noche cuando está permitida la circulación de estos vehículos. Es importante mencionar que en Centroamérica los viajes de distancias largas son comunes, y la infraestructura de países como Guatemala los hace más complicados. A esto se suma el tráfico presente al momento de transitar y las presiones para cumplir tiempos de entrega, factores que incrementan la fatiga de los conductores en estos viajes. Por lo tanto, es crucial controlar su fatiga, ya que es algo que eventualmente sucederá.

El ritmo cardíaco o frecuencia cardíaca refleja la actividad del corazón y puede ser un indicador del estado físico y emocional de una persona. Por lo tanto, cuando se experimenta fatiga de cualquier tipo, el ritmo cardíaco tiende a disminuir y a volverse más irregular, dado que el cuerpo intenta mantenerse alerta. Este comportamiento del ritmo cardíaco puede manifestarse en patrones que pueden ser identificados de manera temprana mediante el monitoreo en tiempo real y alertar al conductor para que tome las medidas necesarias, evitando situaciones peligrosas debido al riesgo de conducir con fatiga.

La inteligencia artificial es un campo en constante evolución que se ha aplicado en diversas áreas para optimizar procesos y mejorar la productividad. En este trabajo, se evalúa el uso de técnicas de Inteligencia Artificial (IA) para detectar la fatiga en conductores a través del análisis del ritmo cardíaco, con el objetivo de identificar patrones tempranos que ayuden a prevenir accidentes vehiculares. Además, el sistema está diseñado para integrarse con un modelo de detección de fatiga basado en reconocimiento facial y un sistema de interacción por voz con el conductor. Se espera que estos hallazgos contribuyan al desarrollo de soluciones más seguras y eficientes para el monitoreo de la fatiga en el ámbito del transporte vehicular.



### 2.1. Objetivo general

Este proyecto tiene como objetivo principal desarrollar un sistema de predicción de somnolencia basado en la frecuencia del ritmo cardíaco, utilizando técnicas de aprendizaje automático para analizar su influencia y evaluar su viabilidad.

### 2.2. Objetivos específicos

- Entrenar los modelos de inteligencia artificial con un soporte de al menos 100,000 ejemplos para cada clase que se desea predecir, que sean datos confiables y que reflejen la relación entre el ritmo cardíaco y las fases de sueño.
- Validar la utilidad del sistema mediante el desarrollo y evaluación de modelos predictivos que permitan identificar la somnolencia, definida en este proyecto como la fase de sueño ligero (N1), asegurando que al menos uno demuestre un desempeño viable y práctico para la detección de la clase objetivo.
- Desarrollar un sistema que reciba datos de un sensor de ritmo cardíaco y que garantice predicciones en tiempo real, manteniendo un intervalo máximo de actualización de 5 segundos entre cada resultado, asegurando así una respuesta ágil en la detección de somnolencia.



Si bien existen sistemas de detección de fatiga en el mercado y la tecnología permite realizar esta tarea mediante sensores u otros *hardware*, las aplicaciones actuales aún no ofrecen una solución accesible y adaptable a diversas necesidades. No se ha consolidado una herramienta confiable y ampliamente adoptada que permita a personas o empresas monitorear y gestionar la fatiga de manera eficiente, asegurando el bienestar de los pilotos y la integridad del producto, especialmente en escenarios de conducción prolongada.

La fatiga es un factor relevante en los accidentes comerciales según la Administración Federal de Seguridad de Autotransportes (FMCSA), representando el 13% de los accidentes graves que involucran vehículos comerciales [7]. Esta cifra corresponde a los Estados Unidos, donde opera la FMCSA, pero es útil como referencia para estimar la cantidad de casos que podrían presentarse en otros países, como Guatemala, donde se registraron 8,218 accidentes en el año 2023, según el Programa de las Naciones Unidas para el Desarrollo (PNUD) en Guatemala [27].

Los síntomas más comunes que una persona puede mostrar cuando experimenta fatiga al conducir son:

- Cabecear mientras se conduce o tener microsueños, que son breves periodos de sueño que duran menos de 15 segundos.
- Cansancio en los ojos y aumento del parpadeo.
- Reacción lenta ante las condiciones cambiantes de la carretera, como el comportamiento de otros conductores o la presencia de peatones.
- Dificultad para mantener un carril o una velocidad constante.

[23]

Dada esta información, se considera que es posible y conveniente desarrollar soluciones que incluyan:

- Con el *hardware* y los sensores adecuados, registrar y digitalizar todo el viaje del conductor para monitorear sus síntomas de fatiga.
- Utilizar modelos de *Machine Learning* para detectar patrones de fatiga en tiempo real.

- Desarrollar una aplicación que pueda generar alertas inmediatas al conductor si se detectan síntomas de fatiga.

Seguir estas bases y cumplir con los objetivos del proyecto ayuda a asegurar el bienestar de los pilotos mientras realizan sus trabajos, mantiene las ganancias de la empresa al evitar la pérdida o daños del producto y de los vehículos, y también contribuye indirectamente a la seguridad de los conductores particulares, debido a la disminución de accidentes de tráfico por fatiga.

Como humanos podemos notar que una persona se está quedando dormida y, según la ocasión, existen distintos márgenes de tiempo entre esta detección humana y el siniestro vehicular. Si como humanos podemos detectar cuándo los conductores se están quedando dormidos, es posible entrenar modelos de **Inteligencia Artificial (IA)** para encontrar estos patrones. Entre los patrones visuales, los más comunes son la frecuencia de parpadeo, el cierre de los ojos y el bostezo. Entre los patrones fisiológicos se encuentran el ritmo cardíaco, la respiración y la temperatura corporal. Los sistemas para analizar este tipo de comportamientos humanos durante los episodios de somnolencia han sido últimamente muy investigados, dado que representan un buen complemento dentro de los vehículos inteligentes. En el año 1998, ya se exploraba el uso de un modelo de redes neuronales que calculaba el cierre lento de los ojos o **PERcentage of eye CLOSure (PERCLOS)** (*PERcentage of eye CLOSure*) a partir de videos; sin embargo, también se aplicó un modelo que tomaba datos directamente de la conducta de manejo, como mantenerse en el carril, movimientos o aceleración del vehículo [11].

## 4.1. Métodos de detección de fatiga en conductores

### 4.1.1. Detección por observación ocular, PERCLOS y otros métodos

Si bien la detección ocular es uno de los métodos más comunes, también se ha intentado utilizar enfoques de medidas psicológicas. Sin embargo, estas suelen ser intrusivas para el conductor, ya que requieren ser colocadas en la cabeza o el cuerpo. Además, otras investigaciones que utilizan algún tipo de entrada subjetiva por parte del conductor pueden llegar a distraerlo de su actividad principal y, por lo tanto, no ser tan efectivas en su medición. Por esta razón, en “*IOT Based Real-Time Drowsy Driving Detection System for the Prevention of Road Accidents*” optaron por utilizar **PERcentage of eye CLOSure (PERCLOS)** como índice de medición de la somnolencia, ya que no es intrusivo para el conductor. Sin embargo, sí se utilizó una bocina para alertar al conductor cuando se detectaba la somnolencia [12].

Otros métodos suelen ser utilizados para la detección de fatiga y fases de sueño, tanto métodos de observación visual como comportamientos (bostezos o parpadeos), la frecuencia cardíaca y la variabilidad del ritmo cardíaco, la temperatura corporal, y, dependiendo de la actividad que se esté realizando, comportamientos que pueden indicar somnolencia o si una persona está dormida o no, como los movimientos mediante un sensor acelerómetro cuando el sueño ocurre en un ambiente controlado, el uso del cursor y escritura si se está usando una computadora, o patrones de manejo si se está conduciendo.

### 4.1.2. Detección por frecuencia cardíaca (HR) y variabilidad (HRV)

La frecuencia cardíaca – **Heart Rate (HR)** y la variabilidad de la frecuencia cardíaca – **Heart Rate Variability (HRV)** no son lo mismo. HR se refiere al número de latidos del corazón por minuto y se expresa en **Beats Per Minute (BPM)**, mientras que HRV se refiere a las fluctuaciones en los intervalos de tiempo entre los latidos del corazón, es decir, la variación en la duración entre un latido y el siguiente. Algunos calculan la fatiga por medio de HR y otros por medio de HRV [31].

Los datos provenientes del ritmo cardíaco han sido considerados para la predicción de fatiga, ya que el *hardware* permite que no sea intrusivo, que se obtengan datos en tiempo real y que se aproveche

la naturaleza secuencial del ritmo cardíaco, permitiendo así capturar cambios graduales en el estado del conductor a lo largo del tiempo. Esto proporciona una visión continua y dinámica sobre la fatiga acumulada. Existen diferentes estudios que demuestran que, efectivamente, la frecuencia cardíaca disminuye por fatiga mientras se conduce. Un ejemplo de esto es el estudio “*Heart Rate Change While Drowsy Driving*”, donde observaron a partir de seis pilotos que el ritmo cardíaco (HR) promedio bajó un  $9.3\% \pm 7.4\%$  en al menos 298 minutos, y de ese tiempo, en promedio, 27 minutos fueron de conducción con fatiga [15]. Otro estudio relevante es “*Early detection of falling asleep at the wheel: A Heart Rate Variability approach*”, donde, con diez voluntarios, notaron que la variabilidad de la frecuencia cardíaca (HRV) en el rango de muy baja frecuencia – **Very Low Frequency (VLF)** decreció consistentemente y significativamente minutos antes de los eventos de caída de sueño [6]. Esto nos demuestra que es conveniente monitorear la HRV, ya que nos puede dar indicios, incluso minutos antes, de que la persona ya no está en condiciones de seguir conduciendo, lo que permitiría evitar accidentes de forma anticipada.

### 4.1.3. Implementación de sensores en vehículos

Si el ritmo cardíaco no fuera útil para detectar la fatiga, las grandes compañías no estarían incorporando este enfoque en sus nuevos automóviles inteligentes. Compañías como Ford, Toyota, Hyundai, Tesla y Harman (de Samsung) actualmente desarrollan y usan sensores para monitorear la frecuencia cardíaca y detectar el estado del conductor antes y después de un accidente, incluso llegando a realizar acciones preventivas, como frenado, debido al acceso al vehículo que tiene el sistema. En el caso de Hyundai, el sistema en desarrollo de Hyundai Mobis utiliza la postura del conductor, la frecuencia cardíaca e incluso ondas neuronales para detectar conducción con fatiga o embriaguez. Ellos dicen que su cabina desempeña el rol de “centro de examinación de salud en movimiento” [13].

## 4.2. Personalización de los modelos de detección

En una mayor variedad de individuos, pueden existir variables que no se ajusten a un modelo específico. El uso de **Machine Learning (ML)** da la posibilidad de tomar en cuenta otras características del conductor para determinar un modelo preciso para distintas poblaciones. En “*Real-time Nonintrusive Detection of Driver Drowsiness*”, se menciona que, aunque no hubo diferencia entre la disminución de **Heart Rate (HR)** en un sujeto masculino y uno femenino, se identificó que cada individuo tiene un patrón de **Heart Rate Variability (HRV)** diferente. Por ello, se recomienda crear criterios de detección personalizados [43]. Además, en “*Intelligent wearable heart rate sensor implementation for in-vehicle infotainment and assistance*”, se consideran síntomas relacionados con el infarto de miocardio y el paro cardíaco, aunque aquí lo hacen principalmente para evitar estos ataques durante la conducción [34].

El tipo de datos a utilizar y las relaciones entre ellos también determinarán el modelo a utilizar. **Support Vector Machines (SVM)**, **Hidden Markov Model (HMM)** y **Convolutional Neural Network (CNN)** han sido algunos de los modelos utilizados, como se menciona en “*10 - Driver drowsiness detection using heart rate and behavior methods: A study*” [41].

## 4.3. Tecnologías de sensores para monitoreo de fatiga

### 4.3.1. Tipos de sensores para medición de HR

El tipo de valor a obtener del sensor de ritmo cardíaco no debería variar entre sensores, ya que lo que se busca es obtener un número que indique la frecuencia, es decir, la cantidad de pulsaciones en un tiempo determinado. Sin embargo, existen varios tipos de sensores, como el de electrodos, sensor de presión, tubo de aire, capacitancia electrostática, ópticos y de ondas de radio. Todos son adecuados para medir *Heart Rate (HR)* en reposo, pero es importante considerar que en movimiento, con turbulencias del vehículo y otras condiciones, algunos de estos sensores pierden precisión. El que mejor se adapta a este entorno es el sensor de ondas de radio, que también permite la medición sin contacto. Algunos estudios indican que obtener mediciones exactas con este tipo de sensor podría ser complicado, pero se ha demostrado que depende de la implementación del sensor. En el artículo “*Development of the in-vehicle pulse sensor. About the pulse rate estimate technique from a microwave sensor*”, se colocó el sensor de ondas en el asiento del conductor para detectar la parte baja de su espalda [33]. En las implementaciones de esta tecnología en Ford, también se utiliza esta posición para colocar los sensores, con seis sensores distribuidos en la parte del asiento que hace contacto con la espalda del conductor [36].

### 4.3.2. Sensores PPG en *wearables* y volantes

Hoy en día, los *smartwatches* son dispositivos comunes que incluyen la función de medición del ritmo cardíaco utilizando sensores fotoeléctricos basados en *Fotopleletismografía (PPG)*. Estos sensores permiten medir parámetros fisiológicos de manera no invasiva mediante una combinación de luces LED y un fotodetector, que registran cambios en el volumen de sangre en los vasos durante cada ciclo cardíaco. El funcionamiento de la *Fotopleletismografía (PPG)* se basa en dos componentes principales: una fuente de luz que penetra el tejido de la piel y un fotodetector que mide la luz reflejada, proporcional a las variaciones en el volumen sanguíneo. Las fuentes de luz utilizadas incluyen *Infrared Light Emitting Diode (IR-LED)*, que se enfocan en el flujo de sangre en tejidos profundos como los músculos, y LEDs verdes, los más comunes, que se utilizan para medir la absorción de oxígeno en la oxihemoglobina y desoxihemoglobina. Dependiendo de la cantidad de luz reflejada, que varía con cada pulsación, se puede calcular con precisión el ritmo cardíaco [5].

Algunos de los puntos débiles de estos sensores son que pueden verse afectados por el movimiento del cuerpo, el contacto con los dedos o alguna afección en la piel. En “*A Review of Heartbeat Detection Systems for Automotive Applications*” también utilizaron este tipo de sensor con luz verde, como los de los *smartwatches*, y mencionan que si se coloca en el volante para detectar con la palma del conductor, puede limitar los movimientos. Los tipos de sensores de ritmo cardíaco en la cabina del vehículo pueden clasificarse como portátiles y no portátiles; los *smartwatches* son portátiles y no resultan intrusivos para el conductor, siendo ya una tendencia que ha llevado a detener las investigaciones de sensores de ondas de radio de Ford mencionados anteriormente. Aun así, usar un *smartwatch* para medir *Heart Rate (HR)* y aprovecharlo para la detección de fatiga al conducir puede presentar varias complicaciones, como la calidad del sensor, ya que existen muchas marcas y algunas pueden tener mejor precisión que otras; la pérdida de exactitud debido al entorno, ya que no hay muchas pruebas con este tipo de sensores en vehículos; que el conductor se lo quite o se quede sin batería; o problemas con la señal o conexión *bluetooth*. Arakawa recomienda la combinación de la detección mediante *Heart Rate (HR)* y la detección de expresiones faciales [2].

## 4.4. Herramientas de simulación y experimentos controlados

Para realizar estudios o experimentos, no es necesario que un conductor se quede dormido mientras conduce de verdad. La mayoría de los estudios utilizan simuladores de conducción, de los cuales hay una gran cantidad y son de fácil acceso para la recolección de datos y pruebas. Esto se evidencia en el *paper* “*Drowsy Driver Detection Through Facial Movement Analysis*”, donde, además, cabe resaltar que utilizaron un modelo para detectar el cierre de los ojos, bostezos y movimientos faciales. Mediante *eye tracking* también se detectó el movimiento de la cabeza, lo cual proporciona información adicional. Estos datos se ingresaron a un modelo de aprendizaje, como **Adaboost** y la regresión multinomial *ridge*, obteniendo una precisión superior al 90 % en promedio para todos los sujetos, utilizando un simulador de conducción [40].

## 4.5. Latencia de sueño y prueba de mantenimiento de la vigilia (MWT)

Se le conoce como latencia de sueño (*sleep latency*) al tiempo que una persona tarda en entrar en la primera fase del sueño. Aunque este es un término técnico, no se puede definir un tiempo exacto para la latencia de sueño, ya que varía según las condiciones de cada individuo. Aun así, se estima que la latencia de sueño oscila entre 10 y 20 minutos. La latencia de sueño es una medición común en estudios sobre el sueño, ya que refleja la somnolencia o la dificultad para dormir [24]. En un estudio de 2010, se determinó que la latencia de sueño calculada mediante la prueba de latencia múltiple de sueño – **Multiple Sleep Latency Test (MSLT)** es una característica clave para predecir los riesgos de colisión en conductores. Sin embargo, este es solo uno de los múltiples indicadores de somnolencia. En el estudio “*Maintenance of Wakefulness Test scores and driving performance in sleep disorder patients and controls*” se clasificó a los participantes en cuatro grupos según los resultados de la prueba de mantenimiento de la vigilia (MWT): patológico (0-19 minutos de latencia), intermedio (20-33 minutos), alerta (34-40 minutos) y control (>30 minutos). Como resultado, observó que el grupo patológico tuvo mayor incidencia de cruce de líneas inapropiado durante la prueba de conducción en un simulador [26].

La prueba de mantenimiento de vigilia – **Maintenance of Wakefulness Test (MWT)**, mide la capacidad de una persona para mantenerse despierta en un entorno tranquilo, durante sus horas normales de vigilia. Esta prueba generalmente se realiza entre 1.5 y 3 horas después de haber despertado de su período principal de sueño. **Maintenance of Wakefulness Test (MWT)** actúa como contraparte de **Multiple Sleep Latency Test (MSLT)**, donde en la MWT la persona intenta mantenerse despierta, mientras que en la MSLT la persona intenta quedarse dormida [37].

## 4.6. Fases del sueño y su relación con la fatiga

### 4.6.1. Clasificación de las fases del sueño

Las fases del sueño se refieren a los distintos patrones de ondas cerebrales y otras funciones fisiológicas que se presentan durante el sueño. Estas ocurren de forma ordenada y, en conjunto, conforman un ciclo de sueño que suele durar entre 90 y 110 minutos en promedio. Por lo tanto, durante un sueño prolongado, pueden darse varios ciclos.

Las fases del sueño se dividen en dos principales: la fase **Rapid Eye Movement (REM)** y la **Non-Rapid Eye Movement (NREM)**, donde la REM se caracteriza por las variaciones en el equilibrio entre el tono simpático y parasimpático. Las fases NREM suelen tener menor actividad cerebral.

Las fases NREM pueden clasificarse como **N1 (Sueño Ligero)**, **N2 (Sueño profundo)** y **N3 (Sueño de ondas lentas)**.

Respecto a la fase de sueño ligero, esta fase es la primera que ocurre después del tiempo de latencia de sueño. Suele darse a medida que el cuerpo se relaja. En esta fase, es común oscilar entre estar despierto y dormido, y una de sus características principales es la reducción del ritmo cardíaco y la relajación muscular. En cuanto a las ondas cerebrales, se producen ondas *theta* de mayor amplitud, las cuales son más lentas que las ondas *alfa* del estado despierto y relajado [17].

Las fases del sueño suelen medirse a través de un **Electroencefalograma (EEG)** que detecta la actividad cerebral y su conexión con otros sistemas, como la actividad ocular y muscular, mediante características de las ondas como la frecuencia y amplitud. Esta suele ser la forma más confiable de detectar las fases del sueño y la somnolencia previa. La clasificación más utilizada se realiza por medio del **Electroencefalograma (EEG)**, la **Electrooculografía (EOG)** y la **Electromiografía (EMG)**. Esto, junto con sensores de actividad cardíaca y respiratoria, se conoce como **Polisomnografía (PSG)**. Aunque este método es el estándar, tiene sus limitaciones y la complejidad de su uso en los sujetos, por lo que se han investigado alternativas, como el monitoreo de **Heart Rate Variability (HRV)** causada por la actividad del sistema nervioso autónomo, que aumenta con la profundidad del sueño y se obtiene a través de sensores cardíacos de **Electrocardiografía (ECG)** [28].

#### 4.6.2. Actividad del sistema nervioso autónomo durante el sueño

El sistema nervioso autónomo lo define Waxenbaum como: “un componente del sistema nervioso periférico que regula los procesos fisiológicos involuntarios, como la frecuencia cardíaca, la tensión arterial, la respiración, la digestión y la excitación sexual. Contiene tres divisiones anatómicas distintas: **Sistema Nervioso Simpático (SNS)**, **Sistema Nervioso Parasimpático (PNS)** y el **Sistema Nervioso Entérico (ENS)**”. También se menciona que “La activación del **Sistema Nervioso Simpático (SNS)** conduce a un estado general de actividad y atención elevadas: la respuesta de ‘lucha o huida’. En este proceso, aumentan la presión sanguínea y la frecuencia cardíaca... El **Sistema Nervioso Parasimpático (PNS)** promueve los procesos de ‘reposo y digestión’; disminuyen la frecuencia cardíaca y la presión sanguínea” [42].

Es común pensar que al empezar a dormirse el ritmo cardíaco baja, y esto es cierto; sin embargo, no siempre es el caso, ya que puede verse afectado por el sistema nervioso autónomo. En “*Assessing the Drowsiness of Drivers*” se menciona que al inicio del sueño, en algunas personas, la frecuencia cardíaca aumenta o cambia muy poco y que ambas divisiones, **Sistema Nervioso Simpático (SNS)** y **Sistema Nervioso Parasimpático (PNS)**, se inhiben en cierta medida. La frecuencia cardíaca resultante depende de cuál de las dos influencias era dominante antes de su inhibición [16].

La frecuencia cardíaca puede dar indicios de qué división del sistema nervioso autónomo estaba activa mientras un conductor se quedaba dormido al conducir. La actividad de autorregulación o resistencia suele elevar la variación de la frecuencia cardíaca. Actividades como resistirse a estar despierto cuando hay somnolencia, o como en el experimento “*Heart Rate Variability Reflects Self-Regulatory Strength, Effort, and Fatigue*” donde se muestra un aumento de **Heart Rate Variability (HRV)** durante una actividad de alto esfuerzo de autorregulación, como comer zanahorias y resistirse a galletas, en comparación con una de bajo esfuerzo, como comer galletas en lugar de zanahorias [32]. Esto también puede darse cuando el conductor sufre de un síndrome de fatiga crónica, donde un aumento de la frecuencia cardíaca y una reducción de **Heart Rate Variability (HRV)** sugieren un predominio del **Sistema Nervioso Simpático (SNS)** [4].

## 4.7. Factores de salud y estilo de vida en conductores

También hay que considerar que, afecciones directas al ritmo cardíaco harán que estas variaciones, dadas por medio del **Sistema Nervioso Simpático (SNS)**, estén fuera de lo esperado. Síndromes como la obesidad, que afecta a una gran cantidad de conductores de vehículos comerciales pesados, hasta un 90 % según la investigación “*Unravelling the Complexity of Irregular Shiftwork, Fatigue and Sleep Health for Commercial Drivers and the Associated Implications for Roadway Safety*”, se asocia con la somnolencia, aunque no existan trastornos del sueño. Factores como la disminución de la actividad física y el consumo de alimentos calóricos ricos en grasa se asocian con la apnea obstructiva del sueño. Otras enfermedades, como la hipertensión, que afecta hasta un 40 % de los conductores de vehículos comerciales, están relacionadas con la apnea obstructiva del sueño a través del **Sistema Nervioso Simpático (SNS)**, que eleva la frecuencia cardíaca y la presión arterial en respuesta a activaciones simpáticas repetitivas y estresantes. En general, los conductores de vehículos pesados suelen sufrir problemas de salud debido a la naturaleza de su entorno laboral, por el tiempo prolongado que pasan en carretera y las barreras para llevar un estilo de vida saludable, lo que contribuye a la fatiga [20].

Los enfoques basados en la medición del ritmo cardíaco en cualquier ámbito han tomado relevancia en los últimos años, debido a la comercialización de métodos accesibles y no invasivos incluidos en dispositivos *wearables*.

El uso de **Fotopleletismografía (PPG)** tiene ventajas y limitaciones. Entre sus ventajas está que los dispositivos son más sencillos y de bajo costo, lo que ha permitido su comercialización. Son de fácil colocación en el usuario, ya que solo es necesario ubicarlos de manera superficial sobre la piel, y también se ha demostrado que, al comparar las señales extraídas de **Fotopleletismografía (PPG)** y **Electrocardiografía (ECG)**, los registros de PPG ofrecen un excelente potencial para sustituir a los registros de ECG [5]. Entre sus principales limitaciones, se puede mencionar que monitorean el cambio dinámico del volumen sanguíneo en un solo punto, como el torso, la muñeca o el dedo. Si se quisiera emplear un sistema multisensor, se enfrentaría el desafío de la sincronización, lo que comprometería la comodidad del usuario. La necesidad de contacto con la piel es otra limitante, que dificulta la medición en caso de heridas o en pieles con pigmento oscuro, incluidas las personas con tatuajes. Además, se debe considerar que la colocación del dispositivo no interrumpa el flujo sanguíneo. Por último, la señal captada por los sensores **Fotopleletismografía (PPG)** suele verse afectada por el movimiento [38].

La detección de somnolencia en conductores es un tema relevante debido al impacto significativo de la fatiga en la seguridad vial. Diversos estudios han demostrado que la fatiga es responsable de un porcentaje considerable de accidentes vehiculares, lo que ha motivado tanto a investigadores como a compañías automotrices a desarrollar sistemas para su detección temprana. Existen múltiples enfoques que incluyen análisis de patrones oculares, conductuales y señales fisiológicas como la frecuencia cardíaca y la variabilidad de frecuencia cardíaca. Durante la búsqueda de literatura relevante, se identificaron varios estudios e implementaciones relacionadas que proporcionan una base importante para este proyecto:

- **A Drowsy Driver Detection System for Heavy Vehicles:** En este estudio realizado en 1998 se combinaron varios métodos, como el análisis *PERcentage of eye CLOSure (PER-CLOS)* (Porcentaje de Cierre de Ojos) y la conducta de manejo, para detectar somnolencia en conductores. Sin embargo, no se incluyó la medición del ritmo cardíaco en este enfoque [11].
- **Early Detection of Falling Asleep at the Wheel: A Heart Rate Variability Approach:** Esta investigación demostró que es posible predecir la somnolencia minutos antes de que el conductor se quede dormido, utilizando *Heart Rate Variability (HRV)*. Este enfoque destaca por su capacidad de anticipación, lo que lo hace útil para prevenir accidentes de manera temprana [6].
- **Sleep Stage Classification from Heart-Rate Variability Using Long Short-Term Memory Neural Networks:** Este estudio propuso el uso de redes neuronales *Long Short-Term Memory (LSTM)* para clasificar las fases del sueño a partir de *Heart Rate Variability (HRV)*. El modelo alcanzó una precisión del  $77,00 \pm 8,90\%$  y un coeficiente  $k$  de Cohen de  $0,61 \pm 0,15$ . Sin embargo, el rendimiento disminuyó en subpoblaciones mayores de 50 años, resaltando la importancia de personalizar los modelos para distintos grupos demográficos [28].
- **Using Wearable ECG/PPG Sensors for Driver Drowsiness Detection Based on Distinguishable Pattern of Recurrence Plots:** Este estudio exploró el uso de sensores portátiles *Electrocardiografía (ECG)* y *Fotopleismografía (PPG)* para detectar somnolencia en conductores, utilizando patrones de recurrencia (*Recurrence Plots, RP*) combinados con *Convolutional Neural Network (CNN)* para analizar *Heart Rate Variability (HRV)*. Los gráficos *ReLU-RP* mejoraron la precisión en un rango de 6% a 17% para ECG y de 4% a 14% para PPG, destacando la viabilidad de estos sensores en entornos reales. Sin embargo, los autores reconocen que la muestra utilizada fue pequeña, lo que podría limitar la generalización de los

resultados. Subrayan la necesidad de realizar pruebas adicionales con *datasets* más amplios y diversos para validar plenamente la efectividad del sistema y asegurar su aplicabilidad en diferentes contextos [18].

- **Detection of Driver Drowsiness Using Wavelet Analysis of Heart Rate Variability and a Support Vector Machine Classifier:** Este estudio se enfoca en la detección de somnolencia mediante *Heart Rate Variability (HRV)*, utilizando señales directas de *Electrocardiografía (ECG)*. Implementaron transformadas wavelet para analizar las señales no estacionarias y un clasificador *Support Vector Machines (SVM)* para diferenciar entre estados alerta y somnoliento, alcanzando una precisión del 95%. Sin embargo, este enfoque, como el anterior, se basa en el uso de ECG, que implica una señal más compleja y difícil de obtener de manera no intrusiva en aplicaciones reales. En contraste, el presente proyecto utiliza directamente la frecuencia cardíaca en *Beats Per Minute (BPM)*, extraída de dispositivos como *smartwatches*, lo que facilita una implementación más práctica y no invasiva para la detección en tiempo real. Aunque los métodos basados en HRV son más detallados, el uso de BPM simplifica la recolección de datos en situaciones de conducción prolongada [19].
- **Machine Learning and Deep Learning Techniques for Driver Fatigue and Drowsiness Detection: A Review:** Este estudio ofrece una revisión exhaustiva sobre las técnicas utilizadas para la detección de somnolencia en conductores mediante aprendizaje automático – *Machine Learning (ML)* y aprendizaje profundo – *Deep Learning (DL)*. En la Tabla 8 de dicho estudio, se presentan sistemas híbridos que combinan características biológicas y comportamentales con resultados variados. Se destaca que los enfoques basados en imágenes tienden a alcanzar mayores niveles de precisión en comparación con otros métodos. Por ejemplo, se menciona que un sistema que utiliza tanto características biológicas como comportamentales (Referencia 109) obtuvo una precisión de solo 57.1%. Esto subraya la importancia de utilizar métodos complementarios y resalta la oportunidad de mejorar la precisión con sistemas híbridos que integren múltiples fuentes de información [1].
- **Computer vision-based approach to detect fatigue driving and face mask for edge computing device:** En este trabajo se presenta un sistema híbrido que combina análisis facial con monitoreo de ritmo cardíaco para la detección de somnolencia. El sistema utiliza un umbral simple para BPM, considerando valores menores o iguales a 67 como indicativos de fatiga y mayores a 68 como estado alerta. Sin embargo, se observó que análisis más profundos del ECG, como el tamaño de los picos, ofrecen información detallada sobre el estado de somnolencia, información que se pierde al trabajar únicamente con BPM como dato agregado [29].
- **Heart Rate Variability-Based Driver Drowsiness Detection and Its Validation With EEG,** En este estudio donde participaron 34 individuos en un simulador, las predicciones detectadas a través de un proceso de control estadístico multivariable (MSPC) que detecta anomalías en *Heart Rate Variability (HRV)*, son comparados con los datos basados en *Electroencefalograma (EEG)* y validados por un especialista en sueño, detectaron 12 de 13 episodios pre-N1 (*Sueño Ligero*) antes del inicio del sueño. Es decir, se lograron 12 de 13 detecciones con diferentes rangos antes de entrar en un episodio de sueño, lo que representa una tasa de falsos positivos de 1.7 veces por hora. Aun así, se demuestra que es bastante efectivo, pero es mejor validar utilizando varios métodos [9].

Estos estudios proporcionan un marco conceptual relevante, pero también resaltan desafíos que este proyecto busca abordar, como la necesidad de generalizar los modelos de detección para diversas poblaciones y la integración eficiente de múltiples fuentes de datos en tiempo real. Este proyecto no solo se enfoca en el uso de la frecuencia cardíaca para detectar somnolencia, sino también en la implementación de modelos que anticipen la fatiga antes de que represente un riesgo crítico para la seguridad vial.

Este proyecto se enfoca únicamente en la detección de somnolencia mediante el uso de la frecuencia cardíaca. No incluye el uso de otros métodos como monitoreo visual o análisis **Electroencefalograma (EEG)**, por lo que se recomienda complementar este sistema con otras técnicas más precisas para mejorar la detección. Los modelos resultantes de este proyecto no son aptos para ser aplicados en un contexto de conducción real.

En este proyecto no se buscó desarrollar modelos fuertemente optimizados o completamente funcionales para la detección precisa de somnolencia o fases de sueño. En su lugar, el enfoque se centró en investigar y explorar la relación entre las variaciones en la frecuencia cardíaca y otras características, como la edad, la fatiga percibida y la calidad del sueño, con respecto a la detección de somnolencia. Se buscó principalmente evaluar el alcance y la capacidad de predicción de los modelos cuando se utilizan únicamente características de fácil acceso y no invasivas, como el ritmo cardíaco.

El nivel más alto de interacción con el usuario es dar el resultado de la predicción; no se busca hacer sonar una alarma o realizar alguna acción para “despertar” al usuario. El sistema aquí presentado busca dar esa predicción a un sistema más amigable al usuario, que puede ser una aplicación web, una aplicación en el teléfono celular o un sistema integrado al vehículo, siempre y cuando pueda tenerse el entorno en el que se ejecuta el modelo.

En este proyecto no se pretende llevar a cabo pruebas exhaustivas de Mantenimiento de la Vigilia – **Maintenance of Wakefulness Test (MWT)**, tanto para la recolección de datos como para la validación del modelo, debido a que este tipo de pruebas son complejas de realizar y requieren la aprobación de un comité de ética, así como el consentimiento explícito de los participantes. Por lo tanto, este proyecto se enfoca en el uso de datos existentes y en la evaluación de los modelos sin recurrir a este tipo de experimentación controlada.

Los modelos entrenados pueden no generalizar bien a todas las poblaciones, ya que se han basado en datos previamente recolectados y no en una muestra representativa de la población en general. Por tanto, se recomienda hacer ajustes adicionales antes de aplicarlo en casos específicos.

Aunque los modelos se entrenaron para la detección de somnolencia, su aplicabilidad a otros contextos (como la fatiga laboral o la evaluación del sueño en condiciones no relacionadas con la conducción) no ha sido validada.



## 7.1. Datasets

En este proyecto se utilizaron dos *datasets* para el entrenamiento y validación de los modelos: *Sleep Heart Health Study (SHHS)* y *PMData*. El SHHS es el conjunto de datos principal, proporcionando la mayoría de la información utilizada en los análisis, incluye métricas confiables del sueño y ritmo cardíaco, dado que es un estudio de laboratorio. Sin embargo, este solo incluye sujetos mayores de 40 años. Por otro lado, *PMData* aporta datos complementarios recolectados por dispositivos *smartwatch*, lo que añade diversidad al conjunto global.

Aunque ambos *datasets* no fueron creados específicamente en el contexto ideal para nuestro proyecto, es decir, no provienen de pruebas de mantenimiento de la vigilia – *Maintenance of Wakefulness Test (MWT)*, sí representan casos de personas en fases de sueño. Esto podría incluir, dentro de las muchas muestras, situaciones análogas a la somnolencia que se experimenta en la conducción, de las cuales algunas pueden aportar más que otras. Estos datos abarcan un amplio rango de situaciones y siguen siendo valiosos para modelar distintos escenarios de somnolencia, en lugar de limitar el entrenamiento solo a casos de conductores fatigados. De esta manera, se considera que tener diversidad en los datos es beneficioso, permitiendo que el modelo generalice mejor y se adapte a diferentes condiciones.

### 7.1.1. Sleep Heart Health Study (SHHS)

El *Sleep Heart Health Study (SHHS)* es un conjunto de datos extenso, desarrollado por el *National Sleep Research Resource (NSRR)*. El *dataset* tiene un enfoque principal en la relación entre el sueño y la salud cardiovascular, recopilando información relevante para estudios epidemiológicos a gran escala. Dado su valor y confidencialidad, el acceso está restringido y solo se permite mediante solicitud aprobada por el NSRR [22].

## Participantes

El SHHS incluye datos de 5,804 participantes, seleccionados para representar diferentes perfiles demográficos de la población estadounidense. Cada individuo tiene entre 1 y 2 sesiones registradas, donde se recogen medidas de sueño y salud cardíaca. La falta de múltiples visitas limita los análisis longitudinales a nivel individual, pero la gran muestra permite realizar estudios transversales robustos.

## VARIABLES RELEVANTES

Este *dataset* ofrece una larga cantidad de variables, incluyendo medidas respiratorias, niveles de oxígeno, movimiento corporal, fases de sueño y datos de frecuencia cardíaca obtenidos mediante **Electrocardiografía (ECG)**. Para este proyecto, nos centramos específicamente en las señales cardíacas y las fases del sueño, aunque los datos están limitados al contexto de las sesiones específicas registradas en laboratorio.

### 7.1.2. PMData - Datos de frecuencia cardíaca y sueño obtenidos a través del *smartwatch* Fitbit

Este es un *dataset* público proveído por **Simula**, una organización de investigaciones tecnológicas que contiene varios *datasets* de acceso público, entre ellos “PM2Data”, que contiene un conjunto de datos de registros de vida de 16 personas durante 5 meses utilizando *Fitbit*. Este *dataset* incluye datos relacionados con actividades deportivas, signos vitales y registros de sueño, entre otros valores como información general de cada participante y reportes de bienestar como estrés, calidad de sueño y fatiga [39]. A continuación, veremos más a profundidad el significado de cada variable en relación con este proyecto y la razón por la cual se eligió cada característica para componer un *dataset* de entrenamiento.

#### Datos a través de *Fitbit*

*Fitbit* es una marca de relojes inteligentes o *smartwatches* que forma parte de la familia de productos de *Google*. El modelo específico utilizado para el *dataset* PMData fue el *Fitbit Versa 2*, del cual se proveen datos sobre calorías, distancia, pasos y ritmo cardíaco.

El *dataset* contiene distintos archivos para cada participante, incluyendo información de ejercicio, minutos sedentarios, minutos de actividad ligera, moderada e intensa; también sobre la frecuencia cardíaca y las fases de sueño. El archivo de frecuencia cardíaca contiene el dato crudo del BPM, la fecha y hora de la toma, y además un valor que indica la confianza de la obtención. La variable **confidence** tiene un valor de 1 y 2 cuando los datos se obtuvieron con movimiento o ruido en el sensor óptico, y un valor de 3 cuando el registro de la frecuencia cardíaca se obtuvo en condiciones óptimas [30]. El archivo con información de sueño contiene registros diarios, detallando las fases de sueño, que pueden clasificarse en: despierto, ligero, profundo y REM; y otro registro más simple que contiene solo las categorías: despierto, inquieto y dormido. Cada registro de sueño tiene un código de información que va de 0 a 3: 0 significa que hubo suficiente información para generar el registro de sueño; 1, que hay insuficiente información de frecuencia cardíaca; 2, que el periodo de sueño fue muy corto (menos de 3 horas); y 3, que hubo un error en el servidor. Además, cada registro tiene una bandera que indica si el sueño fue el principal del día o no [8]. En cada registro de sueño se tiene una lista de fases, que contiene la fecha y hora de inicio de la fase, el tipo de fase y la duración en segundos de la misma.

## Datos a través de PMSys

Por cada participante se tienen tres archivos de registros de datos subjetivos obtenidos a través de una aplicación llamada PMSys:

- **srpe**: Contiene datos sobre actividades deportivas.
- **injury**: Contiene datos sobre lesiones.
- **wellness**: Contiene datos subjetivos de bienestar, como el nivel de fatiga, estado de ánimo, duración y calidad del sueño, así como el nivel de estrés. El conjunto de estos valores se encuentra junto a su respectiva fecha y hora del registro.

## Información general de los participantes

Si bien hay otras dos carpetas con información, estas tratan sobre alimentación, por lo que no se tomarán en cuenta para el *dataset* a crear. Por último, existe un archivo que contiene una vista general de todos los participantes que muestra su edad, peso, género, frecuencia cardíaca máxima y si es una persona de tipo A o B. Las personas de tipo A son más impulsivas e inquietas, mientras que las personas de tipo B son más pacientes y manejan mejor el estrés. Se cree que las personas de tipo A pueden tener más probabilidades de sufrir una enfermedad cardíaca [35].

## 7.2. Modelos

Los modelos utilizados se seleccionaron considerando la naturaleza secuencial del ritmo cardíaco. En primer lugar, se implementa un modelo *Long Short-Term Memory* (LSTM), diseñado específicamente para capturar dependencias temporales y patrones a lo largo del tiempo. Además, se emplea *AutoGluon* en sus variantes *Tabular* y *Forecast*, con el propósito de realizar clasificaciones sobre secuencias temporales en distintos escenarios. Aunque el objetivo del proyecto no es una comparación estricta entre estos modelos, dado que cada uno maneja los datos de manera distinta —incluyendo variaciones en el procesamiento previo de las entradas y salidas—, el uso de múltiples enfoques añade robustez y validez a los resultados, permitiendo explorar distintas perspectivas en la predicción de somnolencia.

### 7.2.1. LSTM

El modelo *Long Short-Term Memory* (LSTM) es un tipo de red neuronal recurrente —*Recurrent Neural Network* (RNN) diseñado para trabajar con datos secuenciales. Este modelo destaca por su capacidad para aprender dependencias a largo plazo al mitigar problemas como el *Vanishing gradient*. *TensorFlow* es una biblioteca de aprendizaje profundo desarrollada por Google, ampliamente utilizada para construir modelos LSTM por su flexibilidad y escalabilidad [10].

El LSTM se utilizará para procesar secuencias temporales del ritmo cardíaco, buscando patrones que permitan identificar estados relacionados con la somnolencia. La capacidad del modelo para retener información a lo largo del tiempo lo convierte en una elección adecuada para estos datos longitudinales.

## 7.2.2. AutoGluon

*AutoGluon* es una biblioteca de código abierto diseñada para automatizar el desarrollo de modelos de aprendizaje automático, simplificando tanto las tareas de clasificación en datos tabulares como las predicciones en secuencias temporales. Su capacidad para optimizar hiperparámetros automáticamente y probar múltiples arquitecturas la convierte en una herramienta adecuada para este proyecto [3].

El uso de AutoGluon permite simplificar el proceso de desarrollo de modelos y explorar enfoques alternativos al LSTM, añadiendo diversidad metodológica sin perder precisión en los resultados.

### AutoGluon tabular

El módulo *TabularPredictor* de *AutoGluon* es versátil y puede utilizarse tanto para tareas de clasificación como de regresión. En este proyecto, se ha utilizado específicamente para la clasificación, ya que estamos interesados en asignar a cada observación una de las tres clases definidas (despierto, sueño ligero o sueño profundo). Esta alternativa al modelo LSTM nos permitió comparar el rendimiento y validar la robustez de los resultados en un contexto de clasificación supervisada.

### AutoGluon forecast para series temporales

El módulo *TimeSeriesPredictor* de *AutoGluon* se empleó para realizar predicciones basadas en secuencias temporales. A diferencia del enfoque de clasificación, este predictor genera valores continuos que reflejan la evolución temporal de los datos. Aunque los valores predichos no se limitan a categorías discretas, se pueden interpretar y ajustar para representar las clases definidas en el problema de predicción.

## 7.3. Ética y seguridad

### 7.3.1. Selección ética de variables

Durante el desarrollo de este proyecto, y particularmente en la selección de características fisiológicas adicionales para complementar el análisis del ritmo cardíaco, se consideró que múltiples variables pueden influir en el ritmo cardíaco. Sin embargo, el uso de datos relacionados con la salud del usuario exige medidas estrictas de confidencialidad y seguridad, para proteger la privacidad del individuo. Aunque el conjunto de datos SHHS incluye numerosas variables, se priorizó la selección de información básica del usuario, evitando la inclusión de datos sensibles que pudieran revelar condiciones sobre su salud. De hecho, el ritmo cardíaco en sí ya representa un dato sensible; por tanto, su uso en una aplicación requiere el consentimiento explícito del usuario, habitualmente a través de los términos y condiciones de la misma.

Otras variables seleccionadas, aunque no necesariamente médicas, también podrían afectar la confidencialidad del usuario, especialmente si el sistema desarrollado se utiliza en un contexto laboral. Por ejemplo, el hecho de que un conductor informe que realiza su trabajo en estado de fatiga, con pocas horas de sueño de baja calidad o en condiciones de alto estrés, podría tener implicaciones en su empleo. En general, las variables que no representan un riesgo para la privacidad de la salud del usuario tienden a ser características básicas, como la edad, el género o la altura. No obstante, se decidió conservar variables relacionadas con el estado de fatiga y sueño actual del usuario, dado que, si se gestionan adecuadamente para proteger su privacidad, pueden proporcionar información valiosa para el modelo.

### 7.3.2. Implicaciones legales y responsabilidad del uso de datos

Como se explicó en la sección de alcance de este proyecto, el sistema desarrollado se limita a ofrecer una predicción del estado de sueño del usuario. Incluso el manejo de este resultado en un sistema más amplio podría representar un riesgo de privacidad. Por ejemplo, si se considera que el usuario está en estado de somnolencia mientras conduce y se registra una alerta, esta información podría tener consecuencias legales en caso de que el conductor se vea involucrado en algún incidente.

### 7.3.3. Consideraciones éticas en la implementación de la Prueba de Mantenimiento de la Vigilia

En los alcances del proyecto se estableció que no se realizarían pruebas exhaustivas de *Maintenance of Wakefulness Test (MWT)*, ya que estas pruebas requieren condiciones controladas de privación de sueño que pueden comprometer la seguridad y el bienestar del participante. La MWT implica inducir al usuario a un estado de somnolencia, lo cual exige su consentimiento explícito debido a los riesgos asociados a su salud física y mental. Este tipo de prueba debería realizarse, idealmente, en un entorno supervisado para reducir cualquier posible riesgo.

### 7.3.4. Limitaciones éticas y propuestas para estudios futuros

En estudios futuros, cualquier intento de realizar pruebas MWT deberá cumplir con protocolos éticos rigurosos y, de ser posible, contar con supervisión médica para garantizar la seguridad de los participantes en todo momento. Mantener el respeto por la privacidad y el bienestar de cada individuo es fundamental, asegurando que las pruebas de privación de sueño se realicen en condiciones que minimicen los riesgos y protejan a los usuarios.

Además, la confidencialidad y protección de los datos personales en estos estudios deben priorizarse mediante prácticas como la anonimización de los datos para evitar la identificación directa de los participantes, la encriptación tanto en almacenamiento como en transmisión para proteger la información frente a accesos no autorizados, y el cumplimiento de normativas de protección de datos, como el *General Data Protection Regulation (GDPR)*, que establecen pautas sobre el uso y la gestión de información sensible.



## 8.1. Procesamiento de los *datasets*

En el desarrollo de este proyecto, inicialmente se utilizó el *dataset* `PMData`, que proporcionaba variables relevantes para el contexto de los conductores, aunque presentaba una cantidad limitada de registros. Posteriormente, se obtuvo acceso al *dataset* `Sleep Heart Health Study (SHHS)`, el cual se adaptó para ser integrado con las variables seleccionadas de `PMData`. En esta sección se detalla la selección inicial de características de `PMData` y el procesamiento de `SHHS`, buscando una integración de ambos conjuntos.

### 8.1.1. Procesamiento del *dataset* `PMData`

#### Limpieza de datos y selección de características de `PMData`

Este *dataset* contiene variables que, en primera instancia, no queremos mantener para nuestro contexto de conductores vehiculares, ya que estas están relacionadas con el ejercicio y la actividad física. Nuestra atención se centra en los archivos `heart_rate.json`, `sleep.json`, `sleep_score.csv` y `wellness.csv` de cada participante que son los que contienen la información relacionada a las sesiones de sueño de los participantes, además del archivo `participant-overview.xlsx` que contiene datos generales de todos los participantes.

Desde `participant-overview.xlsx`, se eligieron las variables que no tienen que ver con la actividad física, obteniendo el ID del participante dentro del *dataset*, la altura, el género y si es una persona de tipo A o B. De `heart_rate.json` se toman todos los valores, la fecha y hora, el valor de la frecuencia cardíaca (BPM) y la confianza de la toma. Cabe aclarar que el archivo `resting_heart_rate.json` tiene la misma estructura, pero es una versión reducida que solo contiene los valores de la frecuencia cardíaca en reposo.

Del archivo `sleep.json` se toma el ID de la sesión (`logId`), la variable `infoCode` que indica la confiabilidad de la sesión y la lista de fases de sueño con su fecha y hora de inicio. Sobre el archivo

`sleep_score.csv` solo se toma el ID de la sesión y el valor de ritmo cardíaco en reposo para esa sesión.

Por último, sobre el archivo `wellness.csv`, se mantienen las columnas de la fecha y hora de la toma sobre los datos de fatiga, ánimo, estrés, calidad del sueño anterior y duración en horas del último sueño.

## Transformaciones de PMData

El objetivo es terminar con un *dataset* de un solo archivo que contenga, en cada fila, una muestra del ritmo cardíaco. La estrategia es unir todas las características por participante, luego realizar un análisis de cada uno y, finalmente, unir los datos de todos los participantes.

Para cada participante:

1. Se descartan las tomas registradas como “de poca confianza”, priorizando aquellas con mayor nivel de confianza en las fases completas de sueño.
2. Iteramos sobre las fases de sueño extraídas de `sleep.json`, conservando solo aquellas sin errores de servidor y que contienen datos suficientes.
3. Para cada fase de sueño, se agregan los registros de BPM que coinciden con el intervalo de inicio y final de la fase, seleccionando únicamente los de mayor confianza.
4. Si la fase de sueño es la primera de la sesión, se agregan de 30 minutos de registros en estado despierto al comienzo de la sesión.
5. Las sesiones que contienen únicamente un tipo de nivel de sueño se descartan.
6. Los datos estáticos del participante se extraen del archivo `participant-overview.xlsx` y se agregan coincidiendo con el ID del participante.
7. Se incorporan los datos de `wellness.csv` coincidiendo por fecha y hora. Si no hay coincidencia, se utiliza el valor promedio de toda la columna para completar los datos faltantes.

## Análisis de datos procesados PMData

Se creó un reporte sobre todo el conjunto de datos de todos los participantes usando *Pandas Profiling* de *Python*, una herramienta que genera automáticamente un informe exploratorio detallado. Con esto podemos observar información que no es clara a simple vista. Algunos de estos son:

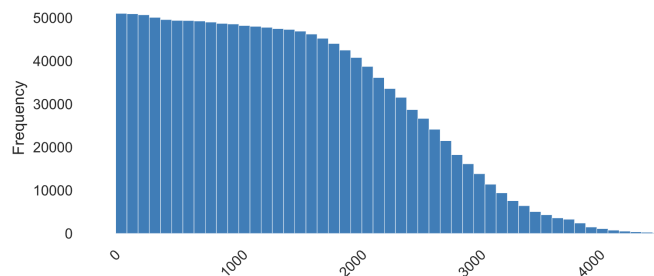


Figura 8.1: Histograma de la columna `BPM_count` del *dataset* PMData.

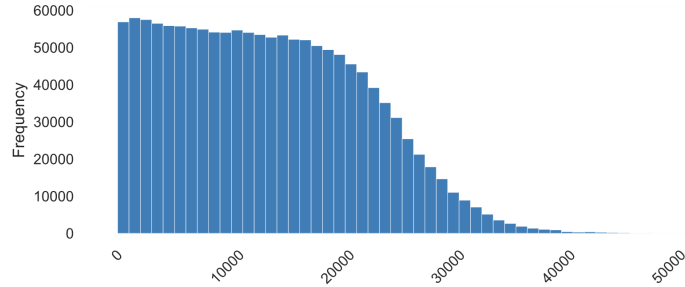


Figura 8.2: Histograma de la columna `seconds_passed` del *dataset* PMData.

Los histogramas [8.1](#) y [8.2](#) permiten observar la cantidad de registros para cada sesión identificada por `logId`. En el primer histograma, la columna `BPM_count` actúa como un contador que incrementa con cada toma de BPM durante la sesión, lo que refleja la densidad de mediciones realizadas. En el segundo histograma, `seconds_passed` muestra la distribución del tiempo en segundos transcurrido en cada `logId`. Esta representación permite analizar la duración y periodicidad de las sesiones, además de la distribución de la cantidad de mediciones.

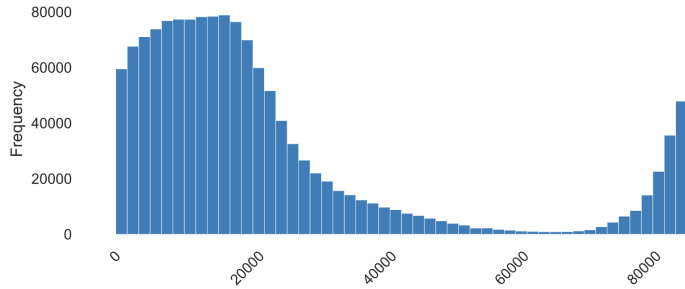


Figura 8.3: Histograma de la columna `day_seconds` del *dataset* PMData.

El histograma [8.3](#) muestra la distribución de la variable `day_seconds`, la cual representa el segundo del día en el que se tomaron las mediciones. La mayor concentración de registros se encuentra al inicio y al final del día, indicando que la mayoría de las mediciones ocurrieron cerca de la medianoche. Esto sugiere que las personas estaban en un contexto de sueño voluntario, es decir, acostándose de manera consciente. Además, esto refuerza que los datos no reflejan situaciones de somnolencia forzada, como en [Maintenance of Wakefulness Test \(MWT\)](#), donde se hacen esfuerzos por mantenerse despierto.

Value	Count	Frequency (%)
1	773318	55.3%
2	465528	33.3%
0	159544	11.4%

Figura 8.4: Conteo de cada categoría de sueño donde 0 es despierto, 1 es sueño ligero y 2 es sueño profundo, para el *dataset* PMData.

En [8.4](#), el mayor porcentaje corresponde a fases de sueño ligero, la cual está compuesta por las fases con las etiquetas “light” y “restless”. Esta distribución puede no seguir la lógica de las fases

de sueño, donde la fase **N1 (Sueño Ligero)** tiene la menor duración. Posteriormente, observando las gráficas de tiempo de las sesiones de sueño, se observa que sí existe una gran cantidad de tiempo en fase de sueño ligero.

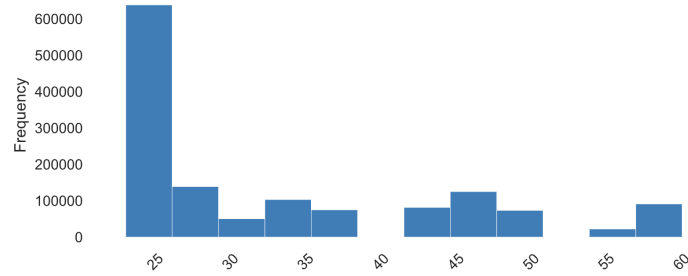


Figura 8.5: Histograma de la columna `age` del *dataset* PMData.

El histograma **8.5** muestra la distribución por edad de los participantes, destacando que este *dataset* incluye un mayor número de personas jóvenes, especialmente en el rango de 25 años. A diferencia del *Sleep Heart Health Study (SHHS)*, que se enfoca en adultos mayores de 40 años, aquí se observa una mayor diversidad de edades, con algunos participantes en edades más avanzadas. La presencia de individuos más jóvenes puede influir en los patrones de sueño y en la detección de fatiga.

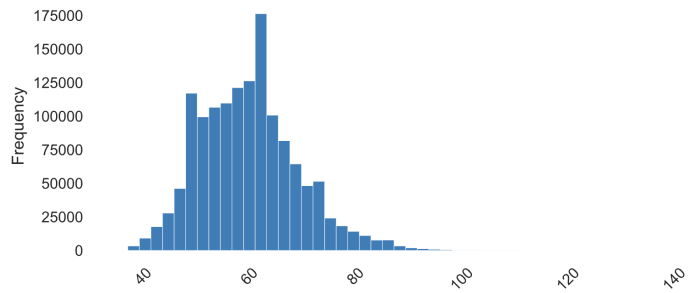


Figura 8.6: Histograma de la columna `BPM` del *dataset* PMData.

El histograma **8.6** muestra la distribución de la frecuencia cardíaca (BPM) en el *dataset*. Se observa un pico notable alrededor de 60 BPM, lo cual es consistente con la frecuencia de reposo típica en adultos. A medida que los BPM aumentan por encima de 80, la frecuencia disminuye gradualmente, lo que refleja la menor ocurrencia de estados más activos o estrés. También se observa una cola hacia valores más altos, que podrían indicar eventos específicos o actividades físicas ocasionales. Este tipo de distribución es útil para identificar rangos normales y detectar posibles anomalías en el ritmo cardíaco.

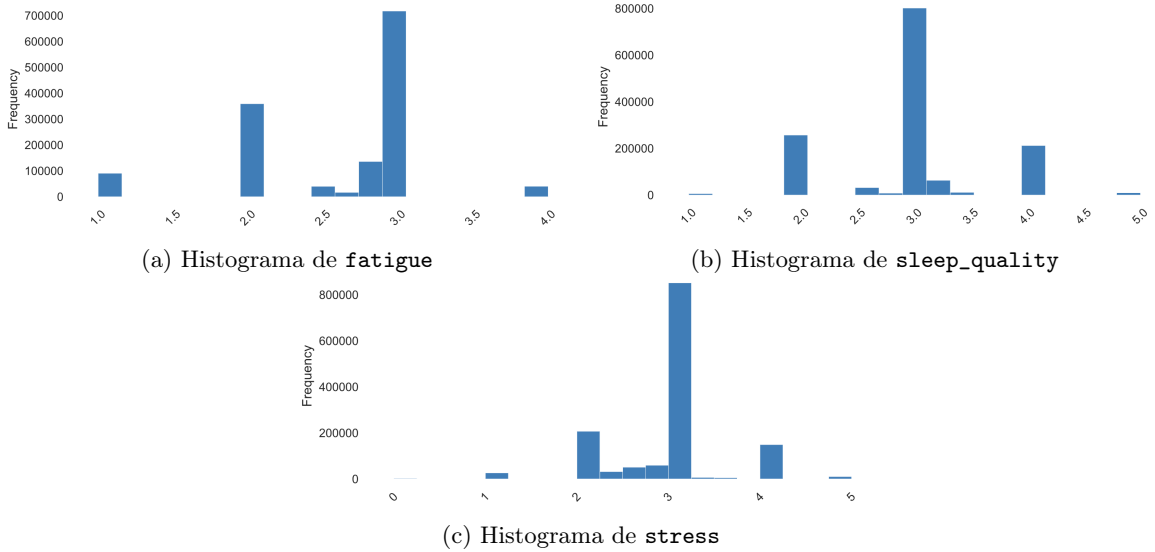


Figura 8.7: Histogramas de las columnas **fatigue**, **sleep\_quality** y **stress** del *dataset* PMData.

Estos tres histogramas [8.7a](#), [8.7b](#) y [8.7c](#) representan las variables subjetivas de fatiga, calidad del sueño y estrés, evaluadas en una escala del 1 al 5 por los participantes. La mayoría de las respuestas se concentran en valores medios, lo que sugiere que los participantes tienden a evaluar su estado de forma equilibrada o neutral. Esta distribución puede afectar los datos al reducir la variabilidad en las respuestas, dificultando la detección de patrones extremos o diferenciados. También implica que los modelos podrían tener dificultades para encontrar correlaciones claras con estas variables si no hay suficiente dispersión en las puntuaciones.

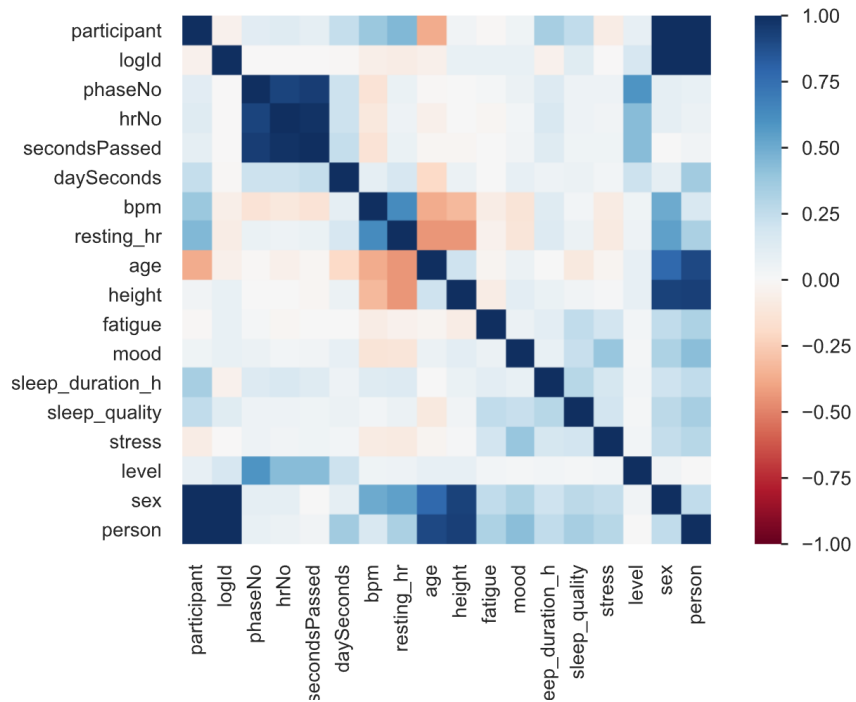


Figura 8.8: Mapa de calor (*heat map*) de las correlaciones de las variables del *dataset* PMData.

El mapa de calor [8.8](#) muestra la matriz de correlación entre las variables del *dataset*, revelando cómo están relacionadas entre sí. Las correlaciones más fuertes se destacan con colores más oscuros. Por ejemplo, variables como `BPM_count` y `seconds_passed` muestran una fuerte relación positiva, lo cual tiene sentido dado que ambas miden progresión temporal en un mismo registro. En cambio, variables como `sex` y `person` tienen correlaciones bajas o nulas con otras, lo que sugiere que podrían tener menor impacto predictivo o redundancia para ciertos modelos.

### Análisis de secuencias de ritmo cardíaco (BPM)

El análisis del comportamiento del ritmo cardíaco (BPM) durante el sueño permite observar cómo varía a lo largo de diferentes fases. A continuación, se presentan gráficas que ilustran distintos comportamientos del BPM, diferenciando las fases de vigilia (nivel 0), sueño ligero (nivel 1) y sueño profundo (nivel 2) mediante colores específicos.

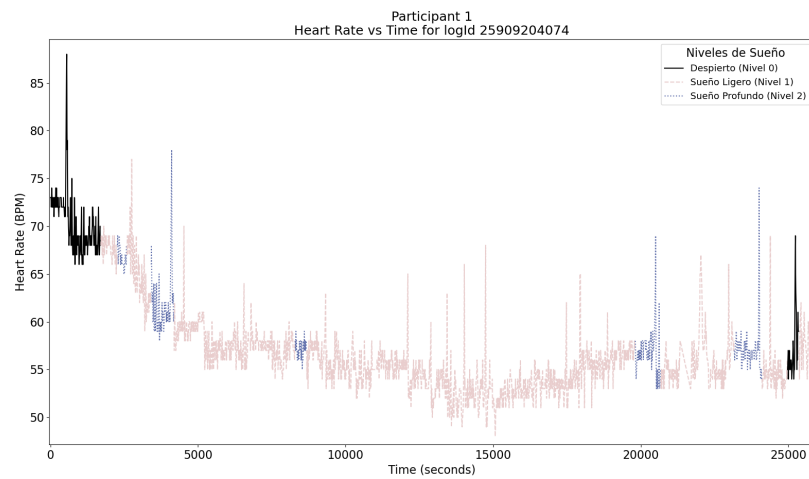


Figura 8.9: Gráfica de ritmo cardíaco con tendencia a bajar.

En la Figura [8.9](#), se observa una tendencia a la baja del BPM al entrar en sueño ligero, lo que representa un patrón común asociado con la relajación progresiva del cuerpo al iniciar el sueño.

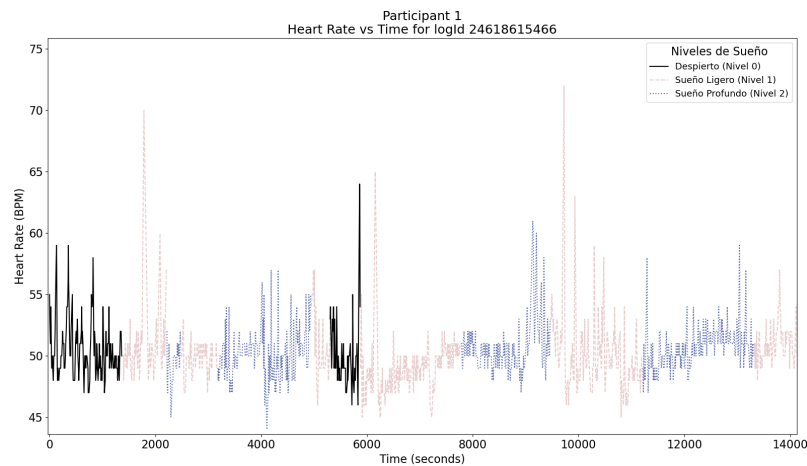


Figura 8.10: Gráfica de ritmo cardíaco constante.

En la Figura 8.10 se aprecia que el BPM permanece relativamente constante al cambiar de vigilia a sueño ligero, lo que indica que no siempre se observa una caída inmediata del ritmo cardíaco durante este cambio de fase.

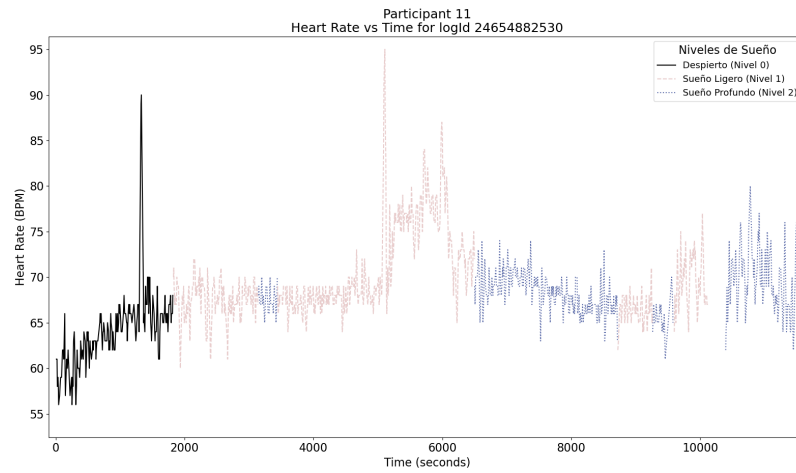


Figura 8.11: Gráfica de ritmo cardíaco con tendencia a subir.

En contraste, la Figura 8.11 muestra una tendencia al alza en el BPM, lo cual puede sugerir la activación del sistema nervioso simpático, como en casos donde una persona se duerme en un estado de alerta, por ejemplo, al intentar mantenerse despierta durante actividades prolongadas, como conducir.

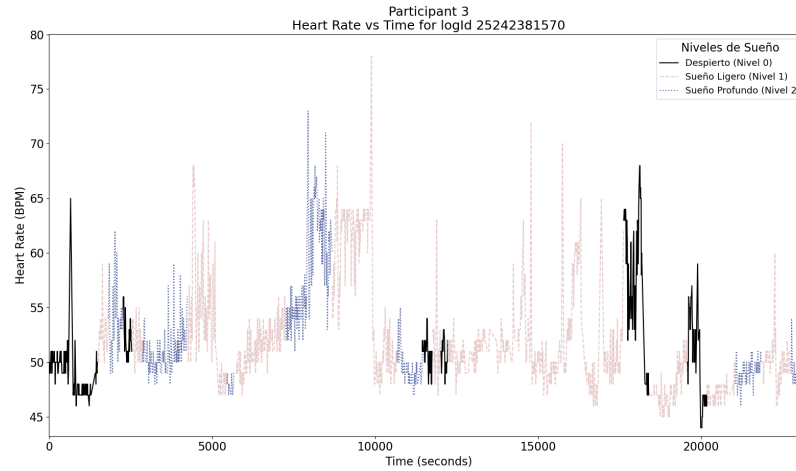


Figura 8.12: Gráfica de ritmo cardíaco fluctuante 1.

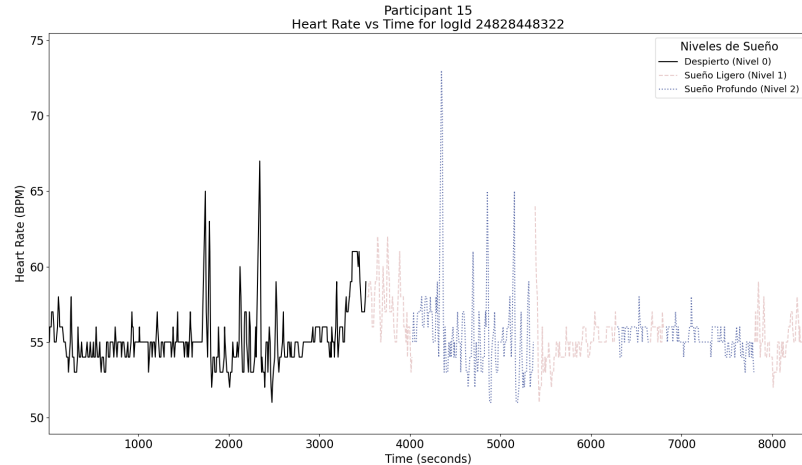


Figura 8.13: Gráfica de ritmo cardíaco fluctuante 2.

En las Figuras 8.12 y 8.13, el primer cambio de fase de vigilia a sueño ligero ocurre durante una subida del BPM. Sin embargo, en la Figura 8.12, se observa una ligera caída en el BPM antes del cambio de fase, mientras que en la Figura 8.13, el BPM se mantiene constante antes de aumentar.

Además, un patrón recurrente es la presencia de un pequeño pico en el BPM justo antes del cambio a sueño ligero. Este comportamiento se puede notar en varias gráficas, como las de las Figuras 8.9, 8.12, 8.11 y 8.13, lo que sugiere que la transición hacia el sueño podría estar acompañada de una breve activación del sistema nervioso.

Este análisis resalta que, aunque los patrones comunes del BPM incluyen descensos y estabilidad en las transiciones de fase, también hay casos de fluctuaciones y aumentos. Estos patrones muestran la importancia de estudiar el ritmo cardíaco de forma secuencial para captar estas dinámicas y comprender cómo podrían impactar en la somnolencia en diferentes contextos.

### 8.1.2. Preprocesamiento del *dataset* SHHS

#### Limpieza de datos y selección de características de SHHS

Este *dataset* es considerablemente grande, con un total de 1,848 características distribuidas en archivos y un volumen total de 356 GB de datos. A diferencia de PMData, donde las mediciones de los participantes se extienden a múltiples sesiones, en SHHS los participantes fueron sometidos a 1 o 2 visitas, durante las cuales se recopilaban señales de Electrocardiografía (ECG) y otros datos clínicos. Esto resulta en una gran cantidad de participantes con solo 1 o 2 sesiones de toma de ritmo cardíaco.

La selección de variables dependió de las que ya se habían elegido en el *dataset* PMData, por lo que se buscaron las variables que representan lo mismo, algunas teniendo un nombre que parecía incluso arbitrario, por lo que se buscaron por medio de su descripción. Se excluyeron solamente las variables de “ánimo” y “frecuencia cardíaca en reposo”, por lo que se decidió no incluirlas al momento de unir ambos *datasets* posteriormente.

Para facilitar el manejo de este extenso volumen de datos, se utilizó la biblioteca *PyHealth* de *Python*, que simplifica el procesamiento y análisis de datos relacionados con la salud. Esta biblioteca proporciona herramientas para el preprocesamiento de señales fisiológicas como el Electrocardiografía (ECG), permitiendo extraer características relevantes de manera más eficiente. En lugar de acceder

manualmente a los archivos en bruto, *PyHealth* automatiza gran parte del flujo de trabajo, ofreciendo funciones que cargan, procesan y organizan las señales y otras variables clínicas.

*PyHealth* permite cargar y procesar eficientemente las señales de **Electrocardiografía (ECG)**. Basta con especificar la ruta de los datos, tras lo cual se recorren automáticamente todos los participantes y sus respectivas grabaciones de polisomnografía. Para unificar los datos de **Electrocardiografía (ECG)** con las etapas de sueño, se cuadran los intervalos entre las tomas, que son siempre de 30 segundos, y cada archivo empieza de manera sincronizada con la sesión.

## Transformación de la señal de ECG a BPM

El siguiente proceso describe cómo transformar una señal de **Electrocardiografía (ECG)** en datos de frecuencia cardíaca o latidos por minuto (**Beats Per Minute (BPM)**). Antes de este análisis, es necesario eliminar cualquier ruido presente en la señal, el cual puede ser provocado por señales externas o movimientos del paciente, por lo que se aplican filtros de paso alto y paso bajo [25]. Para la detección de picos e intervalos RR, se realizan también otras operaciones que facilitan el manejo de la señal, tales como aplicar una derivada para incrementar la amplitud, elevar la señal al cuadrado para eliminar los picos negativos y, finalmente, integrar para resaltar los picos [21].

Entre las señales de **Fotopleletismografía (PPG)** presentes en el archivo en formato EDF, también se encuentra una señal de H.R. (frecuencia cardíaca), la cual se identificó tras realizar el siguiente proceso. Sin embargo, se optó por conservar la frecuencia cardíaca obtenida a través de este procedimiento. Al hacer comparaciones, se observó que ambas señales presentaban los mismos patrones, confirmando que el proceso de transformación de ECG a BPM fue preciso. La diferencia principal es que la señal del archivo EDF presenta una mayor cantidad de valores atípicos en cero, mientras que los valores de BPM transformados a través de nuestro proceso tienen menor pérdida.

- **Lectura y carga de la señal de ECG:** La señal de **Electrocardiografía (ECG)** se lee desde un archivo **European Data Format (EDF)** utilizando la biblioteca **pyedflib**, que permite acceder a los canales de datos del **Electrocardiografía (ECG)**. Se identifica el canal de **Electrocardiografía (ECG)** y se extrae la señal correspondiente, además de obtener la frecuencia de muestreo (*sampling frequency*) de la señal. En las Figuras 8.14 y 8.15 se observa la gráfica de la señal original de **Electrocardiografía (ECG)**.

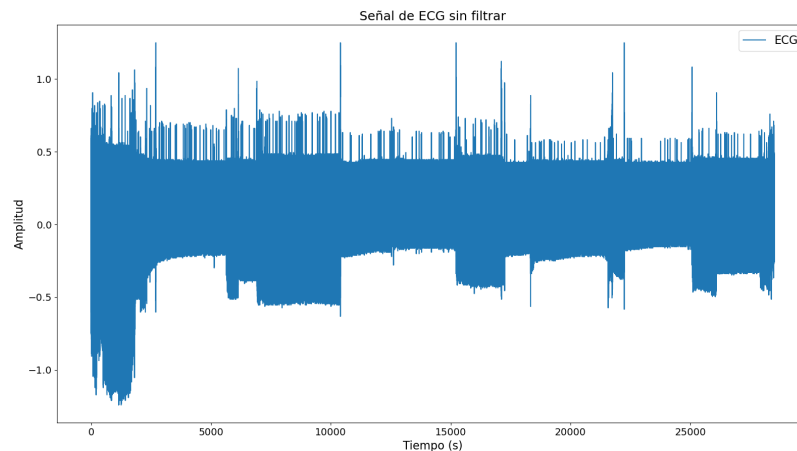


Figura 8.14: Gráfica de ECG del participante 17 de SHHS, sin filtro.

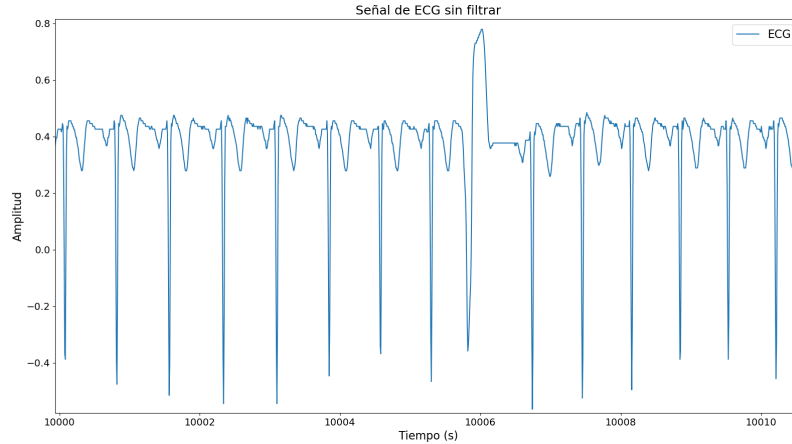


Figura 8.15: Gráfica de ECG del participante 17 de SHHS, sin filtro y con acercamiento.

■ **Preprocesamiento de la señal:**

1. *Filtro de paso alto:* El filtro de paso alto se utiliza para eliminar la “deriva” de la línea base, que corresponde a variaciones lentas o de baja frecuencia en la señal, provocadas por factores como la respiración o el movimiento corporal. Al eliminar estos componentes, el filtro permite enfocar las frecuencias relacionadas con los latidos del corazón. En la Figura 8.16 se observa con mayor detalle el funcionamiento del filtro de paso alto, estabilizando la señal alrededor de cero; si la señal presenta un desplazamiento por encima o por debajo de cero, este filtro la centra en torno a dicho valor.
2. *Filtro de paso bajo:* Posteriormente, se aplica un filtro de paso bajo para eliminar el ruido de alta frecuencia, que puede deberse a interferencias eléctricas o artefactos del entorno, manteniendo únicamente las frecuencias relevantes para la actividad cardíaca. En la Figura 8.17 se ilustra cómo este filtro atenúa las variaciones de alta frecuencia en la señal, logrando una línea más suave que acentúa solo las variaciones significativas.

Estos dos filtros aplicados conjuntamente resultan en una señal más limpia y estable, con menos variaciones indeseadas, lo cual facilita el análisis posterior y la detección precisa de picos.

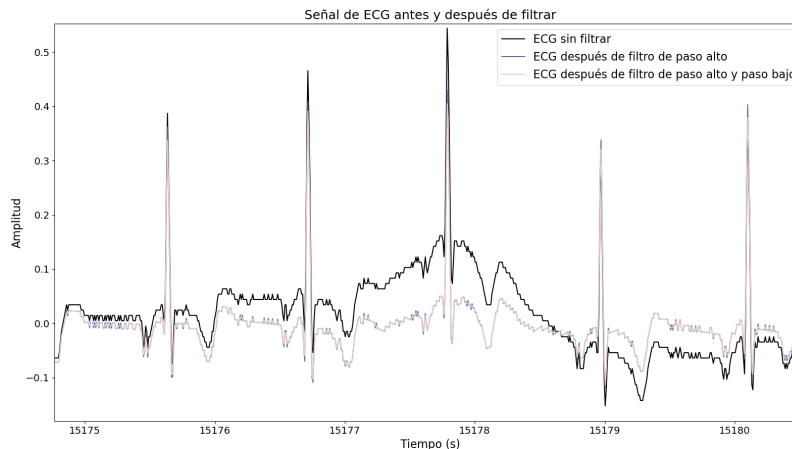


Figura 8.16: Gráfica de ECG del participante 10 de SHHS con filtro de paso alto y filtro de paso bajo.

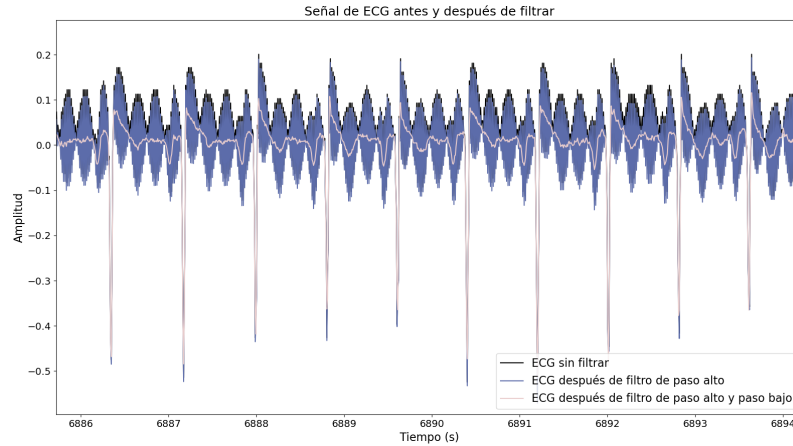


Figura 8.17: Gráfica de ECG del participante 13 de SHHS con filtro de paso alto y filtro de paso bajo.

■ **Transformación de la señal:**

1. *Derivación:* La señal filtrada se deriva para acentuar los cambios rápidos en la amplitud de la señal, lo que ayuda a resaltar los latidos del corazón.

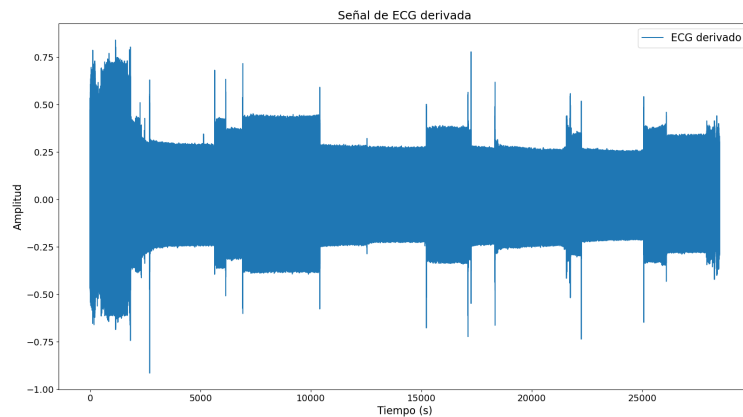


Figura 8.18: Gráfica de ECG del participante 17 de SHHS, derivada.

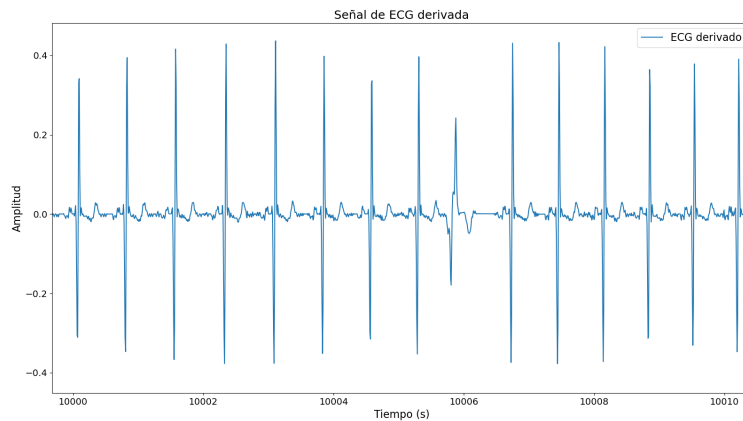


Figura 8.19: Gráfica de ECG del participante 17 de SHHS, derivada y con acercamiento.

2. *Cuadrado de la señal:* Al elevar la señal derivada al cuadrado, se amplifican las diferencias y se suprimen pequeñas fluctuaciones, lo que hace más notables los picos correspondientes a los latidos.

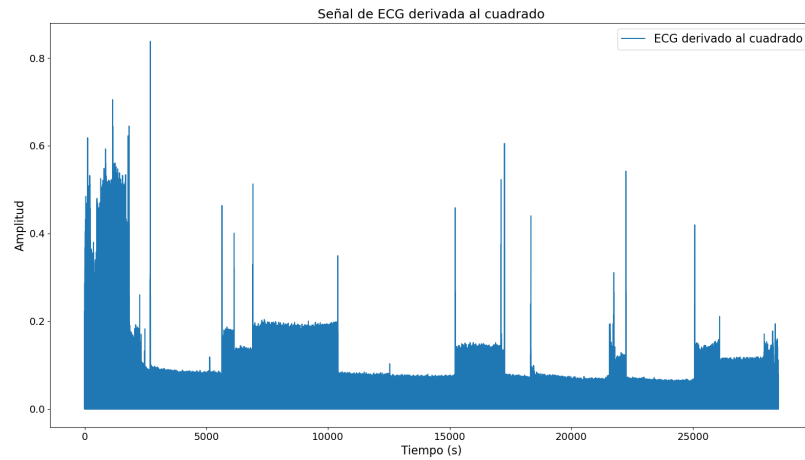


Figura 8.20: Gráfica de ECG del participante 17 de SHHS, elevada al cuadrado.

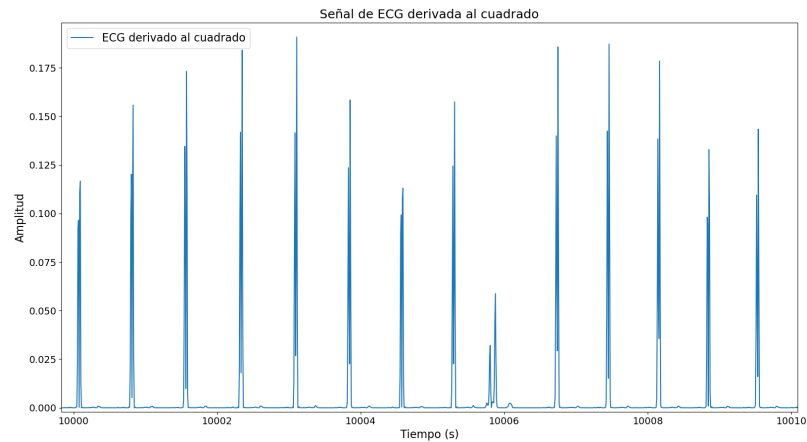


Figura 8.21: Gráfica de ECG del participante 17 de SHHS, elevada al cuadrado y con acercamiento.

3. *Integración:* Se aplica una integración con una ventana móvil, lo cual permite suavizar las variaciones rápidas y enfocar la información en los picos correspondientes a los latidos cardíacos. En la Figura 8.23 se observa cómo los picos aparecen más uniformes y definidos en comparación con los picos sin integrar en la Figura 8.21. Este proceso reduce la probabilidad de interpretar erróneamente pequeños picos dobles o artefactos de ruido como latidos adicionales, mejorando así la precisión en la detección de latidos.

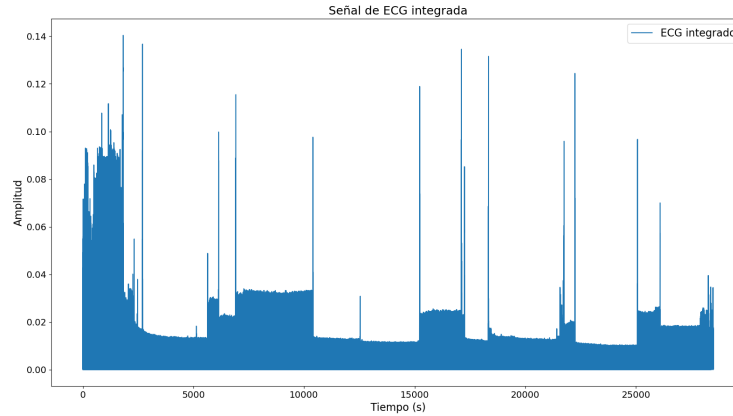


Figura 8.22: Gráfica de ECG del participante 17 de SHHS, integrada.

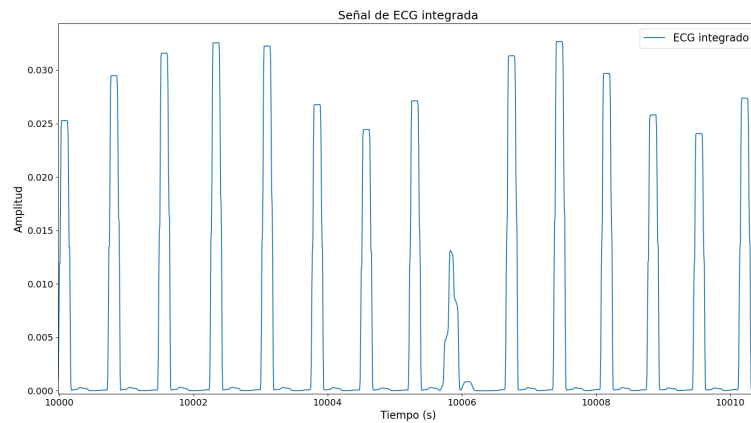


Figura 8.23: Gráfica de ECG del participante 17 de SHHS, integrada y con acercamiento.

- Detección de picos:** Usando la función `find_peaks` de la biblioteca `SciPy`, se detectan los picos en la señal integrada, que corresponden a los latidos del corazón. En este punto, es importante hacer referencia visual a los picos observados en la Figura [8.24](#) que representan los puntos de interés en la señal.

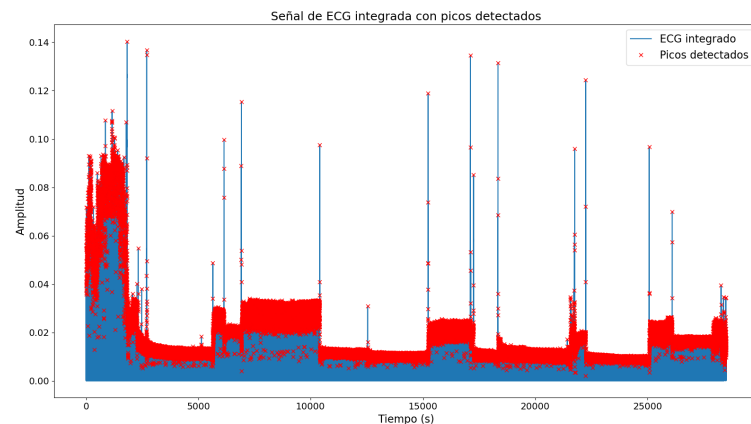


Figura 8.24: Gráfica de ECG del participante 17 de SHHS, detección de picos.

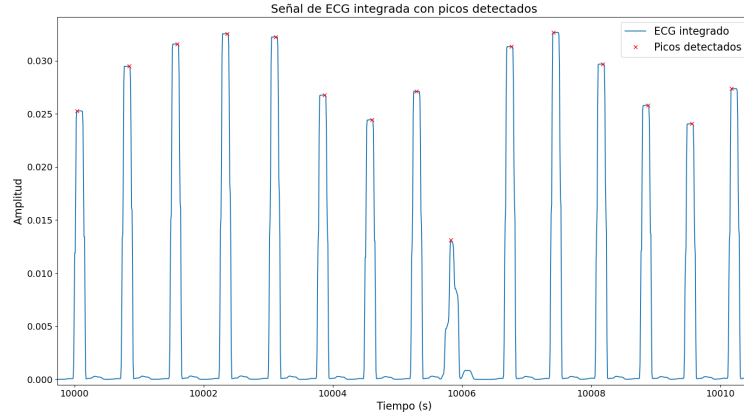


Figura 8.25: Gráfica de ECG del participante 17 de SHHS, detección de picos con acercamiento.

- **Cálculo del ritmo cardíaco (BPM):** A partir de los picos detectados, se calculan los intervalos RR (el tiempo entre latidos consecutivos). Con estos intervalos, se calcula la frecuencia cardíaca en latidos por minuto (Beats Per Minute (BPM)), dividiendo 60 por el valor de los intervalos RR.
- **Promedio de BPM por intervalos:** Los valores de Beats Per Minute (BPM) se agrupan en intervalos de 5 segundos para obtener un promedio de la frecuencia cardíaca a lo largo del tiempo, lo que permite observar la variación en el ritmo cardíaco de manera más uniforme. Cabe aclarar que en este punto se define que se está usando Heart Rate (HR) en lugar de Heart Rate Variability (HRV) ya que si bien el HRV está directamente relacionado con los intervalos RR, al obtener la frecuencia y promediarla en ventanas de 5 segundos, se consolida que la medida es HR (un promedio de latidos por minuto) y se pierde la variabilidad entre intervalos individuales, lo cual es clave para el HRV.

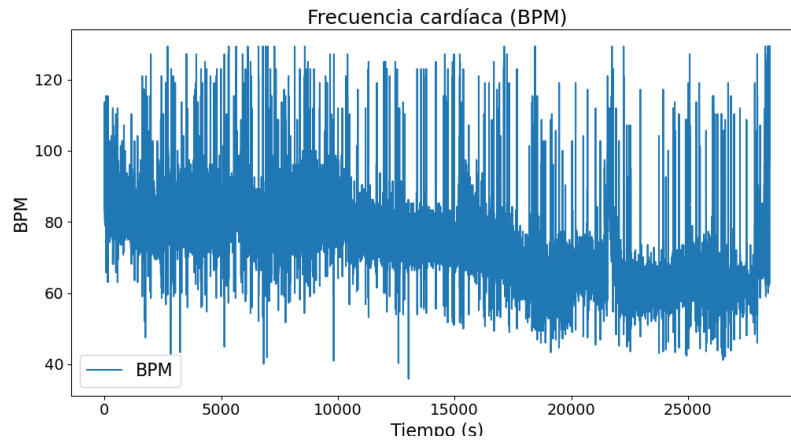


Figura 8.26: Gráfica de BPM del participante 17 de SHHS.

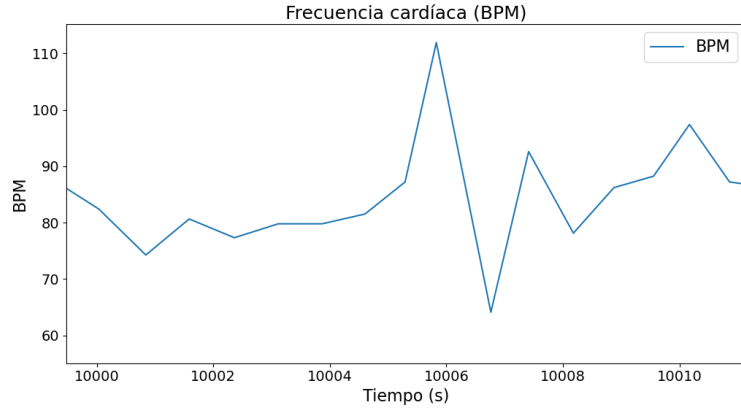


Figura 8.27: Gráfica de BPM del participante 17 de SHHS, con acercamiento.

- Asignación de fases de sueño:** Finalmente, los valores de *Beats Per Minute (BPM)* se vinculan con las fases de sueño (vigilia, sueño ligero, sueño profundo), asignando una fase a cada intervalo de *Beats Per Minute (BPM)*. Esto permite analizar la relación entre las fases del sueño y los cambios en el ritmo cardíaco de manera detallada.

Este proceso completo nos permite obtener un conjunto de datos que representa la evolución del ritmo cardíaco durante las distintas fases del sueño, lo que facilita el análisis de la actividad cardíaca en estas fases.

Sin embargo, después de aplicar el proceso de transformación de *Electrocardiografía (ECG)* a *Beats Per Minute (BPM)*, algunas secuencias presentan rangos y valores incoherentes. Como se observa en las Figuras 8.29 y 8.31, las gráficas resultantes de *Beats Per Minute (BPM)* no son apropiadas para su uso, ya que contienen muchos valores por debajo del rango común de *Beats Per Minute (BPM)*, incluso llegando a 0. Estas grandes diferencias de rango provienen de la señal original de *Electrocardiografía (ECG)*, como se puede ver en las Figuras 8.28 y 8.30. A diferencia de la señal mostrada en la Figura 8.14, estas señales están saturadas y presentan una pérdida de variabilidad en diferentes tramos. Este problema se propaga a lo largo del proceso de transformación. La mejor forma de filtrar estas secuencias es descartar las sesiones en las que un porcentaje significativo de los valores esté fuera de los rangos fisiológicos. En este caso, se aplica un filtro que elimina las secuencias en las que más de  $\frac{1}{3}$  de los valores se encuentran por debajo de 40 *Beats Per Minute (BPM)* o por encima de 200 *Beats Per Minute (BPM)*.

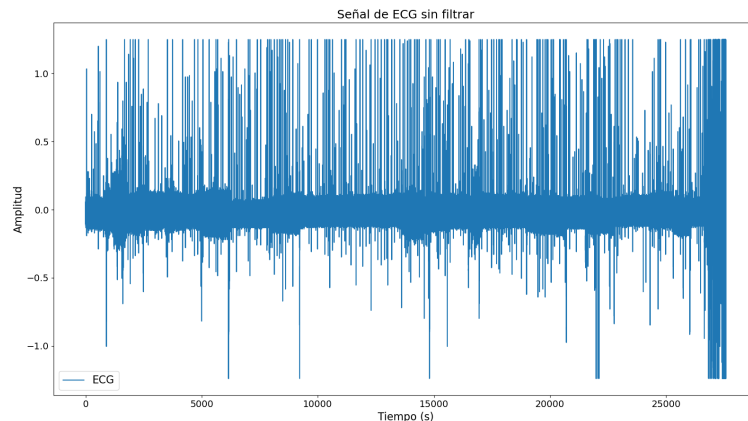


Figura 8.28: Gráfica de ECG del participante 7 de SHHS, sin filtro.

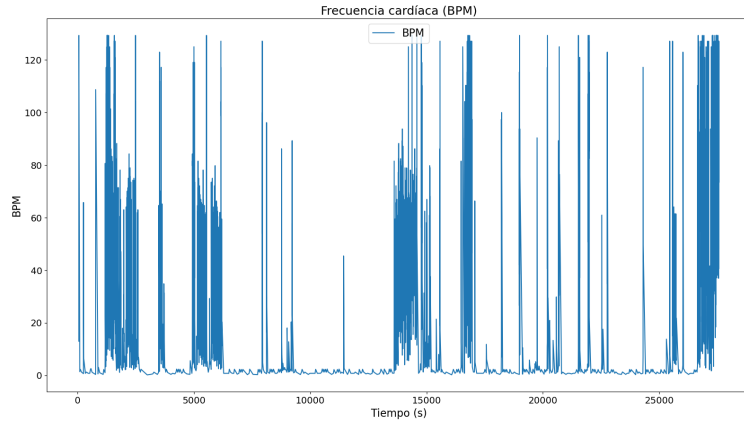


Figura 8.29: Gráfica de BPM del participante 7 de SHHS.

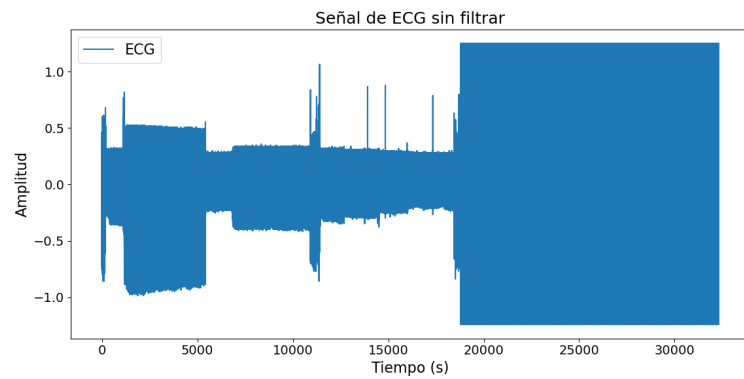


Figura 8.30: Gráfica de ECG del participante 20 de SHHS, sin filtro.

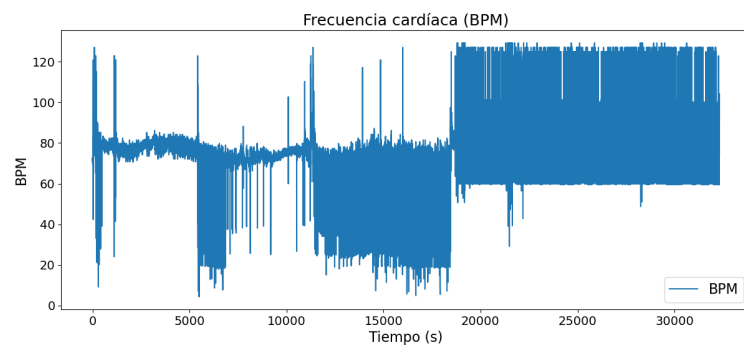


Figura 8.31: Gráfica de BPM del participante 20 de SHHS.

### Análisis de datos procesados SHHS

Una vez generadas las secuencias de ritmo cardíaco, se integran con las variables estáticas de cada participante, lo que permite generar un reporte completo del *dataset* y descubrir información que no es visible a simple vista.

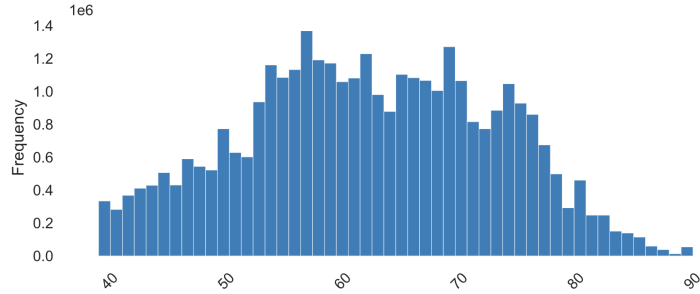


Figura 8.32: Histograma de la columna `age` del *dataset* SHHS.

En la Figura 8.32, se observa que este *dataset* está compuesto principalmente por personas mayores, lo cual es relevante ya que no hay restricciones de edad para conducir un vehículo en este contexto.

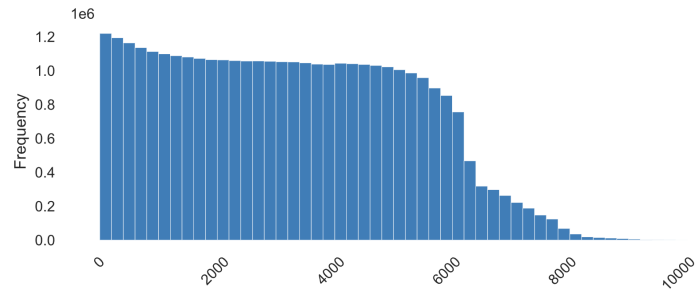


Figura 8.33: Histograma de la columna `BPM_count` del *dataset* SHHS.

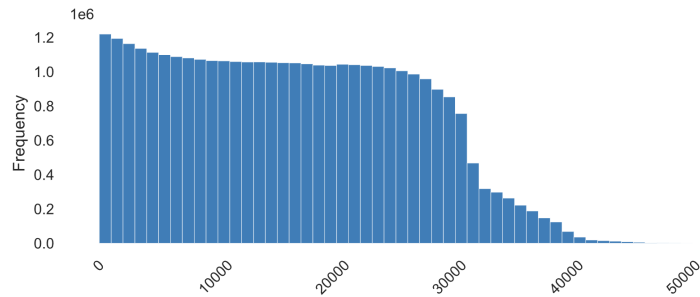


Figura 8.34: Histograma de la columna `time` del *dataset* SHHS.

En las Figuras 8.33 y 8.34, se observa que, al igual que en PMData (8.2), las sesiones presentan una distribución acorde a su duración. Aunque las sesiones en términos de tiempo son similares, en SHHS se registra casi el doble de datos en la misma cantidad de tiempo, a comparación de PMData (8.1), posiblemente debido a la mayor precisión del electrocardiograma, que permite una toma de mediciones más constante y con mayor frecuencia.

Value	Count	Frequency (%)
2	13501959	39.1%
0	11064182	32.0%
5	4626691	13.4%
3	3887036	11.2%
1	1254897	3.6%
4	234244	0.7%
9	858	< 0.1%
6	6	< 0.1%

Figura 8.35: Conteo de la columna `SleepStage` del *dataset* SHHS.

Posteriormente, se realizó un *mapeo* de esta columna: los valores 9 se convirtieron en 0, los valores 3, 4 y 5 se convirtieron en 2, y los valores 6 se eliminaron. A diferencia de `PMData` (8.4), en este *dataset* se registraron más datos de la fase de sueño profundo que de sueño ligero. De hecho, hay muy pocos registros de sueño ligero, lo que sugiere una menor cantidad de muestras para estudiar el cambio de fase de vigilia a sueño ligero.

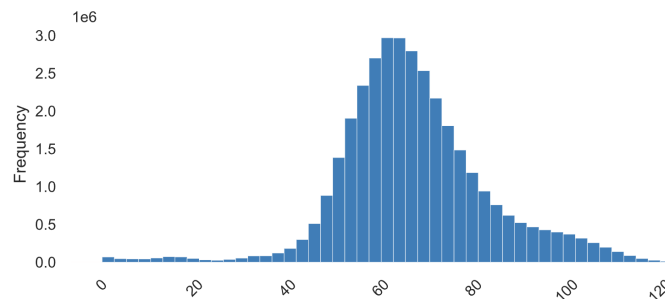


Figura 8.36: Conteo de la columna `BPM` del *dataset* SHHS.

En el histograma 8.36, se observa que la mayoría de los valores de BPM se encuentran dentro del rango fisiológico esperado (40-120 *Beats Per Minute (BPM)*), con una mayor concentración alrededor de los 70 *Beats Per Minute (BPM)*. No obstante, persisten algunos valores por debajo de los 40 *Beats Per Minute (BPM)*. A pesar de haber filtrado las sesiones con más de  $\frac{1}{3}$  de valores fuera de los límites aceptables, es normal encontrar algunos valores atípicos debido a variaciones individuales o eventos específicos durante el monitoreo del sueño. Estos valores pueden ser considerados *outliers*, pero eliminarlos no es tan sencillo, ya que esto afectaría la integridad de las secuencias temporales. Por ello, se optó por mantener estos valores. En trabajos futuros, se podría investigar la posibilidad de etiquetar dichos valores como errores de medición, retirada del dispositivo por parte del usuario o incluso indicadores de posibles problemas de salud.

Value	Count	Frequency (%)
3	16325384	47.2%
4	13088907	37.9%
5	2701724	7.8%
2	1764641	5.1%
1	689217	2.0%

Figura 8.37: Conteo de la columna `fatigue` del *dataset* SHHS.

Value	Count	Frequency (%)
3	17122222	49.5%
2	13696797	39.6%
1	3203720	9.3%
4	547134	1.6%

Figura 8.38: Conteo de la columna `sleep_quality` del *dataset* SHHS.

Value	Count	Frequency (%)
1	22226559	64.3%
2	7025601	20.3%
3	5317713	15.4%

Figura 8.39: Conteo de la columna `stress` del *dataset* SHHS.

Al igual que en PMData, para las variables `fatigue` (8.37) y `sleep_quality` (8.38), se observa una tendencia hacia una valoración neutral de su estado. Sin embargo, en la variable de estrés (8.39), la mayoría de los participantes evaluaron su nivel de estrés por debajo de la media, lo que podría explicarse debido a la edad avanzada de los participantes, lo que podría influir en la menor cantidad de situaciones en las que deban mantenerse despiertos, como en pruebas de vigilia simulando la conducción.

### Análisis de secuencias de ritmo cardíaco de SHHS

Al igual que se analizaron algunas sesiones de ritmo cardíaco del *dataset* PMData, lo hacemos con las del *dataset* SHHS para comprender cómo están estructuradas las secuencias.

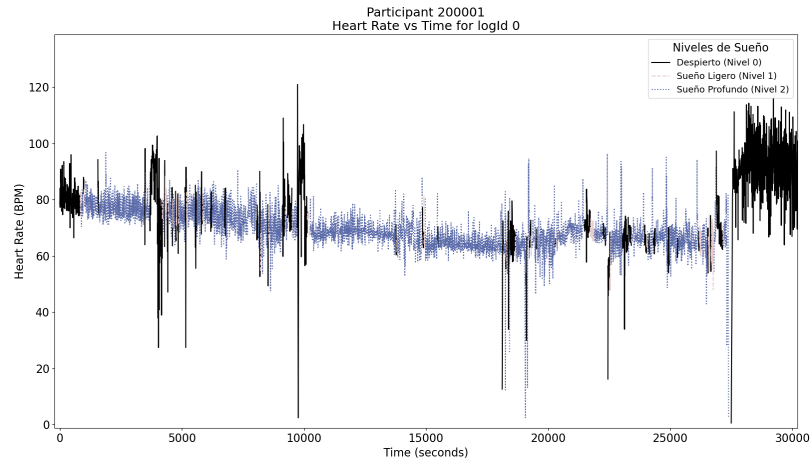


Figura 8.40: Gráfica de ritmo cardíaco con picos a 0.

En la Figura [8.40](#) vemos que, aunque la mayor parte del tiempo se mantiene en un rango válido de ritmo cardíaco, en ocasiones baja a 0. Como no ocurre en un gran porcentaje de veces, esta gráfica pasó el filtro aplicado, lo que explica por qué algunos valores muy bajos aparecen en el histograma [8.36](#).

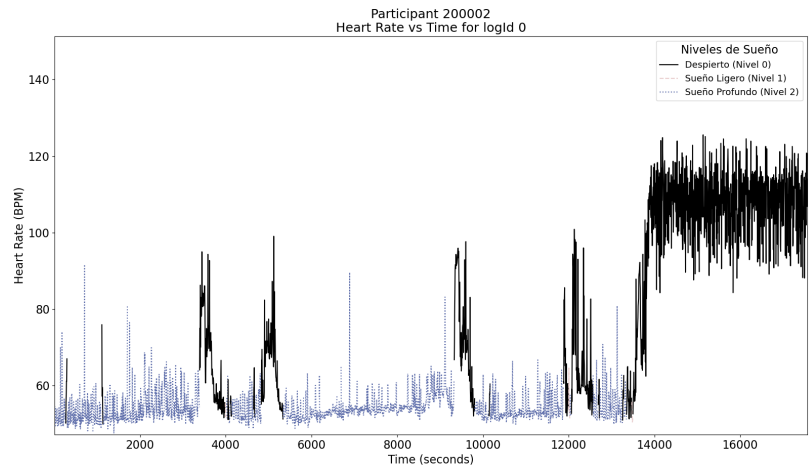


Figura 8.41: Gráfica de ritmo cardíaco en escenario no relacionado con somnolencia en conductores.

En la Figura [8.41](#) observamos un comportamiento inusual del ritmo cardíaco respecto a las otras sesiones y que no es representativo para nuestro contexto de conducción. Sin embargo, según nuestros filtros, esta sesión fue considerada válida. Casos como este pueden introducir ruido en el entrenamiento de nuestros modelos, ya que son atípicos. En este ejemplo, se observan muchas fases de sueño profundo al inicio, momentos en los que se despierta y sube su ritmo cardíaco para luego bajar de nuevo en fase de sueño ligero y profundo y, por último una gran cantidad de datos en fase despierto en una frecuencia alta respecto a lo anterior, lo cual es inusual.

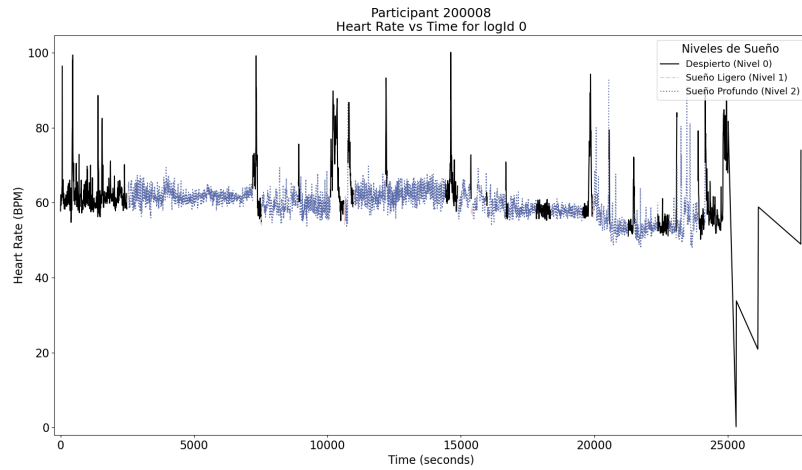


Figura 8.42: Gráfica de ritmo cardíaco con anomalías en los valores finales.

La Figura 8.42 es otro ejemplo de una sesión que puede introducir ruido, ya que hacia el final se observa un comportamiento anormal del ritmo cardíaco, con valores muy bajos y con muy pocas variaciones.

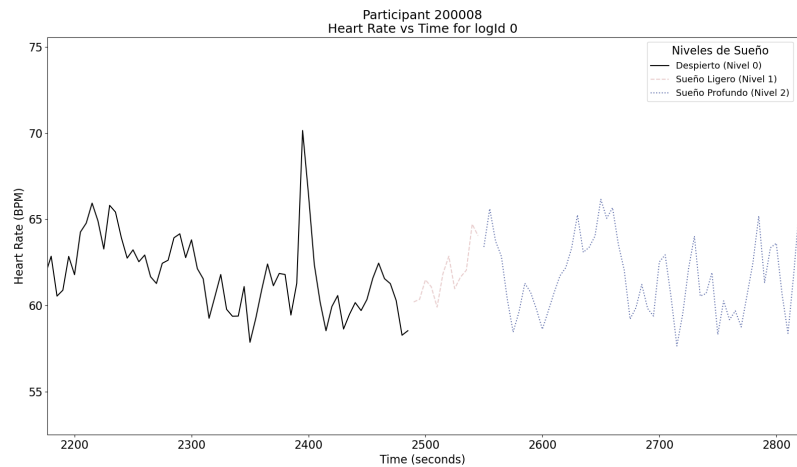


Figura 8.43: Gráfica de ritmo cardíaco de SHHS con acercamiento.

En la Figura 8.43 vemos un acercamiento a los cambios de fase en una sesión de SHHS. Aunque en este caso no se observa un patrón claro, sí notamos que los cambios de fase son continuos según la secuencia de *Beats Per Minute (BPM)*.

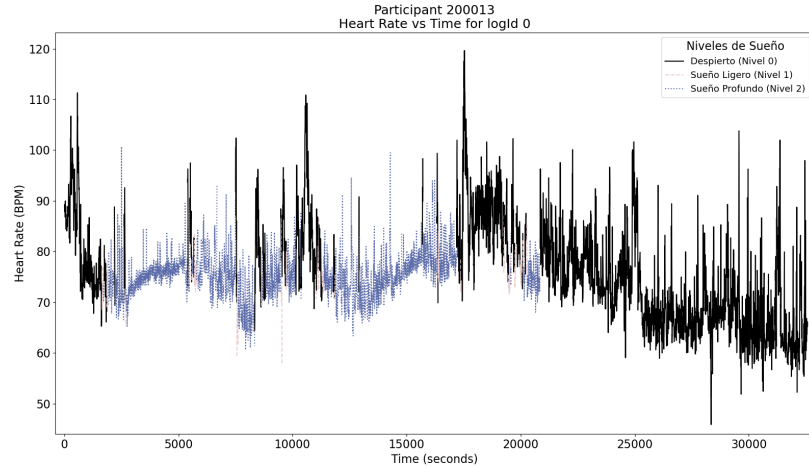


Figura 8.44: Gráfica de ritmo cardíaco de una sesión correcta de SHHS.

Uno de los objetivos durante la recolección de datos en las visitas del *dataset* SHHS es registrar las fases del sueño. Por ello, en algunas sesiones observamos varios cambios de fase, lo cual es beneficioso para el entrenamiento de nuestros modelos, ya que existen más casos en los que se capturan estos cambios de fase, desde despierto hasta dormido, que son clave para nuestros objetivos.

### 8.1.3. Procesamiento en conjunto de PMData y SHHS

Luego de unir los *datasets* PMData y SHHS de forma que tengan las mismas *features* con los mismos nombres y rangos de valores, se realizaron algunas operaciones para enriquecer el contexto del *dataset*, principalmente en torno al ritmo cardíaco.

El conteo final de las fases de sueño, como se observa en la Figura 8.45 muestra una cantidad reducida de la fase de sueño ligero (N1), lo cual es esperado ya que esta es una de las fases con menor duración y la que consideramos fase de sueño profundo termina siendo una combinación de las fases N2, N3 y la fase REM. El porcentaje total de la fase de sueño ligero es el 5.6% lo cual es bastante bajo para la fase más clave en nuestro proyecto. La fase de sueño profundo representa el 63.2% y la fase despierto el 31.2% sobre un total de 35,968,257 datos.

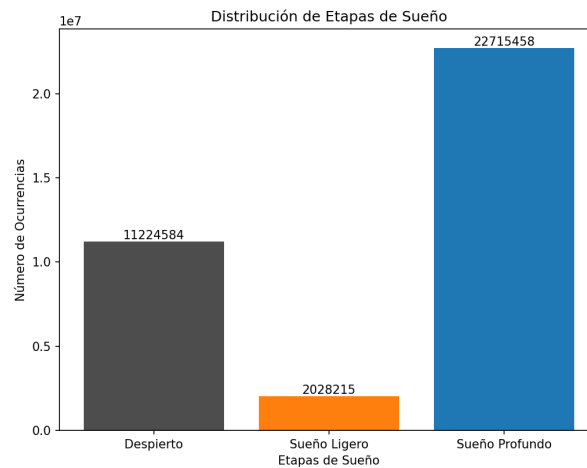


Figura 8.45: Distribución de las fases de sueño SleepStage en el *dataset* completo.

Primero, se realizó una unión entre el id del participante y el id de la sesión, con el objetivo de ir removiendo los IDs, ya que son variables que no aportan a los modelos o que pueden influir de manera negativa. Por el momento, solo se dejó un identificador de sesión para hacer operaciones de agrupación más adelante.

A la variable `BPM_count`, que representa un contador de tomas de ritmo cardíaco, se le realizó un pequeño ajuste: ahora se reinicia a 0 cada 15 minutos, con el objetivo de acortar ventanas o crear secuencias de menor tamaño, evitando que influyan registros de tiempo muy anteriores. Aunque posteriormente se crearán secuencias de manera manual para los modelos LSTM y *AutoGluon Forecast*, este cambio se realizó porque la variable `time` ya lleva un registro continuo desde el inicio de la sesión y no se reinicia.

Se agregó una variable llamada `BPM_Mean_Acc`, que calcula el promedio de los datos de BPM hasta el momento de esa toma durante la sesión. Lo mismo se hizo para la varianza, con la variable `BPM_Var_Acc`, y para la desviación estándar, con `BPM_Std_Acc`.

También se incluyó una variable derivada de la diferencia entre el valor de BPM actual y el anterior, llamada `BPM_Diff`, y su propia derivada, es decir, la diferencia entre el `BPM_Diff` actual y el `BPM_Diff` anterior, lo cual representa la aceleración del BPM y se denomina `BPM_Acceleration`. Por otro lado, la variable `BPM_Mean_Diff` es la diferencia entre el BPM actual y el promedio de BPM hasta ese momento.

En lugar de tener una variable con la edad como número, se utilizaron categorías de rangos de edad: menor a 25 años, entre 25 y 40 años, entre 40 y 60 años, y mayor a 60 años.

Se agregaron dos variables que cambian cada vez que cambia la fase de sueño. La primera es `SleepStage_Changes`, que es prácticamente un contador de las veces que cambia la variable `SleepStage`, y la segunda es `Time_SleepStage`, que cuenta el tiempo que se lleva en esa fase y se reinicia a 0 cuando cambia.

Por último, se aplicó una operación para calcular la tendencia del BPM en las últimas 5 filas. Esta variable se llama `BPM_Trend` y calcula la pendiente de la siguiente manera.

$$m = \frac{n \sum_{i=1}^n x_i y_i - (\sum_{i=1}^n x_i) (\sum_{i=1}^n y_i)}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}$$

En esta ecuación,  $n$  representa el tamaño de la ventana de cálculo, que en este caso es de 5 muestras. La variable  $x$  es un índice temporal de las observaciones en la ventana, comenzando desde 0 hasta  $n - 1$ , lo que permite calcular la posición de cada valor dentro de la ventana. La variable  $y$  corresponde a los valores de BPM dentro de la ventana.

El cálculo de la pendiente  $m$  se realiza mediante una regresión lineal sobre los puntos de la ventana, donde:

- $\sum_{i=1}^n x_i$  es la suma de los índices de tiempo dentro de la ventana.
- $\sum_{i=1}^n y_i$  es la suma de los valores de BPM dentro de la ventana.
- $\sum_{i=1}^n x_i y_i$  es el producto cruzado entre los índices de tiempo y los valores de BPM.
- $\sum_{i=1}^n x_i^2$  es la suma de los cuadrados de los índices de tiempo.

La pendiente  $m$  obtenida de esta manera indica la tendencia de los valores de BPM dentro de la ventana: valores positivos indican una tendencia al alza en el BPM, mientras que valores negativos indican una tendencia a la baja.

En la Figura 8.46 se muestra una gráfica de caja y bigotes que representa la relación entre los valores de BPM en cada una de las fases de sueño que vamos a predecir. Como era de esperarse, la clase 0 (Despierto) presenta valores de BPM más altos en comparación con las otras fases, con un valor central cercano a los 70 BPM, lo que sugiere que los ritmos cardíacos son más elevados cuando la persona está despierta. También se observa que en esta clase la variación es mayor, ya que los bigotes son más largos, lo que podría indicar momentos de actividad durante esta fase.

En cuanto a la clase 1 (Sueño ligero), es la fase que mantiene los valores más bajos de BPM y con menor variabilidad, según se la longitud de los bigotes. La clase 2 (Sueño profundo) es similar a la de sueño ligero, aunque con valores de BPM ligeramente más altos. En las tres clases observamos los valores atípicos, algunos que pueden ser válidos, principalmente los superiores, pero entre los inferiores vemos esos valores atípicos que suelen llegar incluso a 0, pero que se han decidido no quitar.

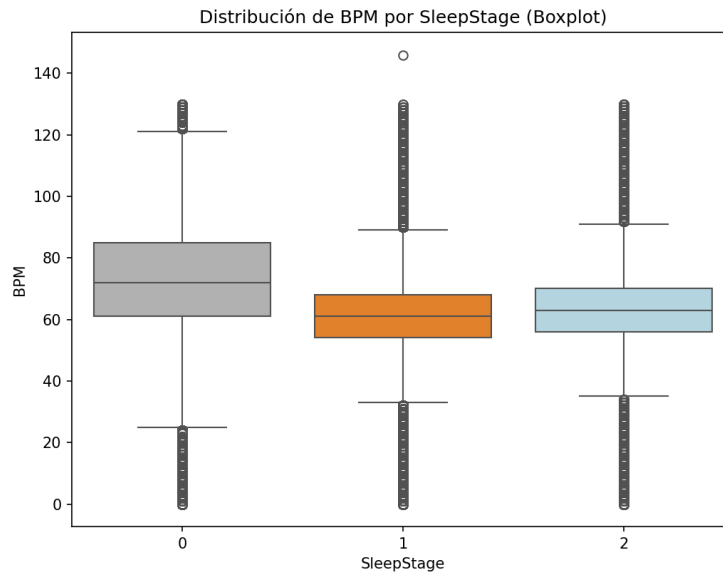


Figura 8.46: Gráfica de caja *box plot* de la distribución de BPM por fase de sueño (*SleepStage*).

En la Figura 8.47 se muestra la correlación entre las variables del dataset total, incluyendo las variables generadas a partir de BPM, como la media acumulada, la tendencia, entre otras. La mayoría de las variables presentan una correlación baja o muy baja entre sí, lo cual sugiere que representan aspectos independientes de los datos. Esto podría ser beneficioso para construir un modelo que combine diversas características sin redundancia excesiva. En este mapa de calor, se observa que *SleepStage* y BPM tienen una correlación negativa moderada, lo que indica que los valores de BPM tienden a disminuir en las etapas de sueño más profundas, algo ya evidenciado en la Figura 8.46. En general, *SleepStage* muestra una correlación moderada con varias variables. Otras variables generadas en este proceso parecen no estar relacionadas con el resto, más allá de aquellas también derivadas del BPM; por ejemplo, *BPM\_Diff* y *BPM\_Acceleration* muestran una correlación nula con *SleepStage* y otras variables.

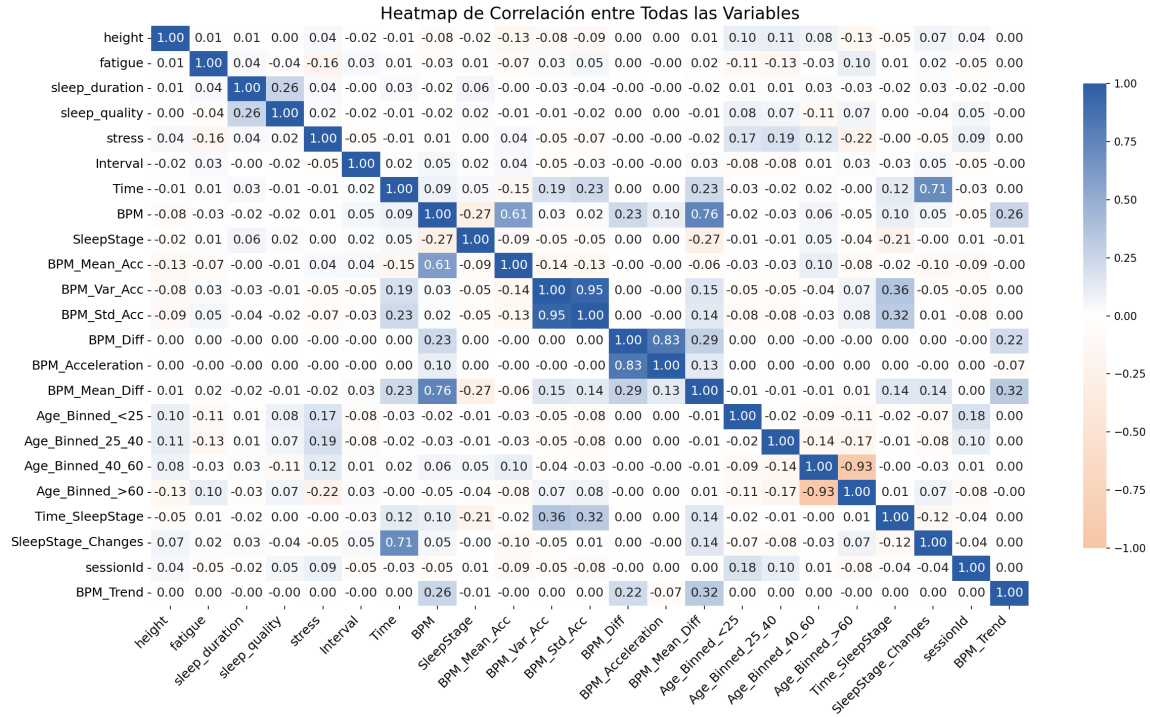


Figura 8.47: Mapa de calor (*heat map*) de las correlaciones de las variables del *dataset* total.

Durante el desarrollo y procesamiento de los datos, se almacena cada conjunto resultante después de cada operación de alto consumo de tiempo. Esto permite optimizar el flujo de trabajo y ahorrar tiempo, brindando la posibilidad de realizar ajustes sin tener que repetir los procesos previos, que suelen requerir recursos significativos debido al volumen de datos manejado. Además, se realizan operaciones en conjuntos reducidos a modo de prueba para validar que las transformaciones se aplican correctamente antes de proceder con el conjunto completo.

## 8.2. Modelo LSTM

### 8.2.1. Preparación de datos para LSTM

El modelo Long Short-Term Memory (LSTM) requiere recibir una secuencia de datos, ya que su principal característica es inferir patrones temporales o secuenciales dentro de los datos, pudiendo predecir el siguiente valor de una secuencia basándose en la información pasada.

Primero, se dividió el *dataset* en un 80% para entrenamiento, un 10% para validación durante el entrenamiento y un 10% para validación posterior, utilizando la función `train_test_split` de *Scikit-Learn*. Esta partición se realizó de forma aleatoria a nivel de grupos de sesiones para mantener la coherencia de los datos de cada sesión. Para lograrlo, primero se obtuvieron los `sessionId` únicos de cada sesión y se aplicó la separación a estos `sessionId`. Luego, en cada conjunto de datos (entrenamiento y validación), se incluyeron los registros asociados con los `sessionId` correspondientes a ese conjunto. Cada uno de estos conjuntos es almacenado en un archivo de formato `parquet` el cual es un formato de almacenamiento de datos eficiente y comprimido, optimizado especialmente para almacenar grandes conjuntos de datos en columnas, ya que permite almacenar y leer solo las columnas necesarias, lo cual reduce significativamente el tiempo de lectura y el espacio en disco.

Antes de crear las secuencias, las *features* son escaladas. Se aplica un *StandardScaler* a las variables continuas y un *MinMaxScaler* a las variables con rangos fijos; estos escaladores se ajustan utilizando únicamente el 80% de los datos de entrenamiento. El *StandardScaler* estandariza las variables, ajustando los datos para que tengan una media de 0 y una desviación estándar de 1, y se utiliza principalmente para datos que siguen una distribución normal. Por otro lado, el *MinMaxScaler* escala los datos para que queden dentro de un rango específico, típicamente entre 0 y 1, y se aplica a variables con límites definidos. Las variables binarias, como los grupos de edad, no son escaladas y se incluyen como variables categóricas en el *dataframe*.

- **Variables continuas:** `height`, `sleep_duration`, `BPM_count`, `time`, `BPM`, `BPM_Mean_Acc`, `BPM_Var_Acc`, `BPM_Std_Acc`, `BPM_Diff`, `BPM_Acceleration`, `BPM_Mean_Diff`, `Time_SleepStage`, `SleepStage_Changes`, `BPM_Trend`.
- **Variables de rango fijo:** `fatigue`, `stress`, `sleep_quality`.

Las secuencias generadas contienen 24 filas, lo que representa los últimos 2 minutos, dado que las lecturas de ritmo cardíaco se realizan en intervalos de aproximadamente 5 segundos. El modelo `Long Short-Term Memory (LSTM)` recibe dos tipos de entradas: variables secuenciales, que cambian con el tiempo, y variables no secuenciales, que permanecen constantes durante toda la sesión. Al procesar y almacenar las secuencias en *TFRecord* —un formato de *TensorFlow* diseñado para manejar grandes volúmenes de datos de manera eficiente—, los datos se serializan diferenciando entre las variables y la etiqueta de clasificación (`SleepStage`). Es importante señalar que la variable `sessionId` se utiliza únicamente para agrupar las secuencias, pero no forma parte de las características del modelo.

La generación de secuencias se realizó en paralelo utilizando `Parallel` de `joblib`, donde cada subproceso almacena *chunks* de 500,000 filas. Este enfoque permite agilizar significativamente el proceso de generación de secuencias y evita sobrecargar las etapas posteriores, como el entrenamiento y la validación, al permitir que estos procesos se ejecuten con conjuntos de datos más manejables en lugar de cargar todos los datos simultáneamente.

### 8.2.2. Estructura del modelo LSTM

El modelo está diseñado para manejar secuencias de datos, como las generadas a partir de la lectura del ritmo cardíaco, y realizar predicciones a partir del historial de la secuencia. Este modelo se compone de varias capas que procesan tanto las variables secuenciales como las no secuenciales, donde estas últimas, que no cambian a través del tiempo, no interfieren en la capa `Long Short-Term Memory (LSTM)`. Las capas incluyen entradas de secuencia, operaciones matriciales, redes bidireccionales, normalización y atención. En la Figura 8.48 se muestra la estructura del modelo LSTM, donde en la salida de cada capa, el valor “None” significa una dimensión flexible que es la dimensión de lote o la cantidad de ejemplos procesados simultáneamente.

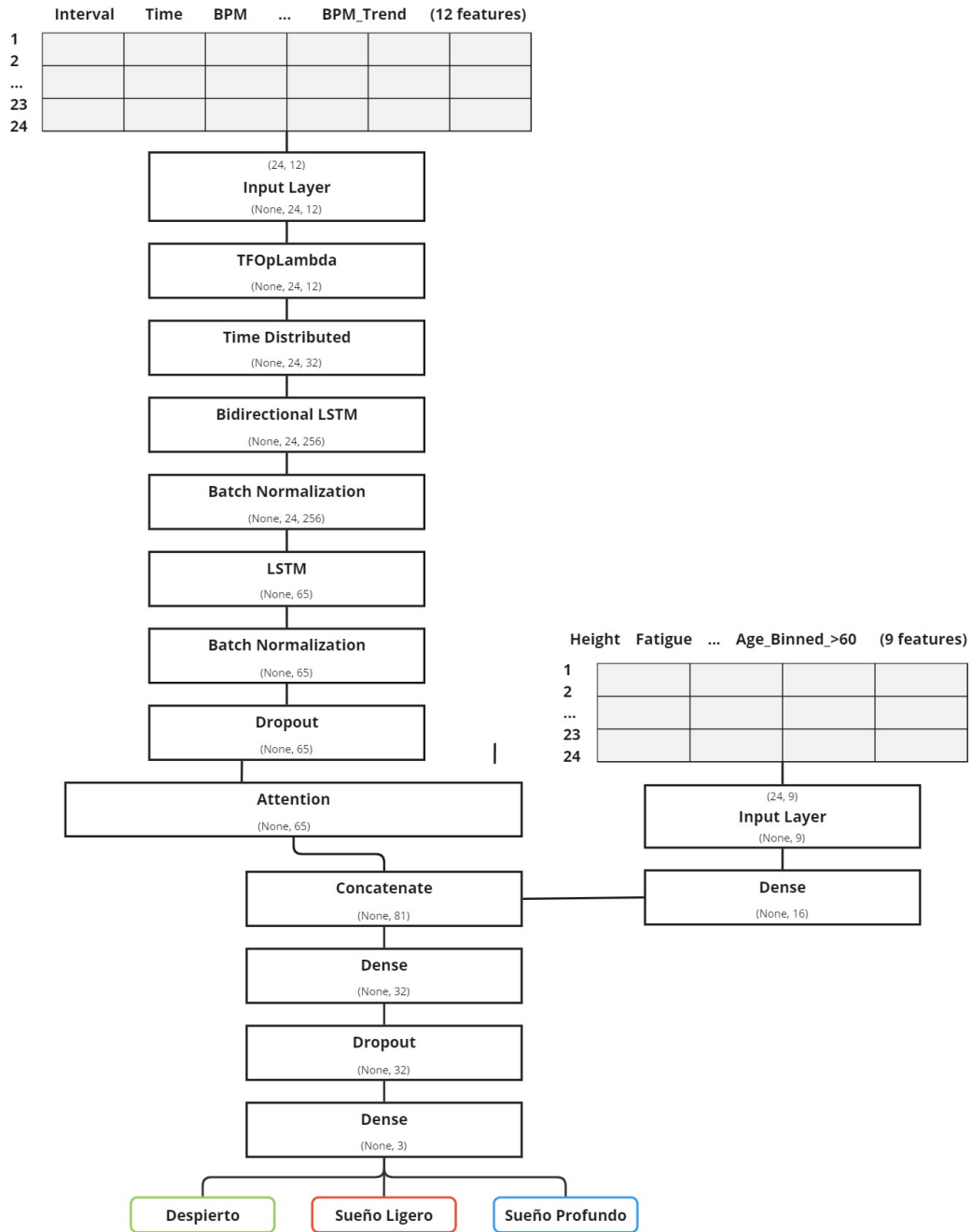


Figura 8.48: Arquitectura del modelo LSTM.

- **Entrada de las variables secuenciales (InputLayer):** Recibe las características secuenciales (21 en total), como el ritmo cardíaco (BPM), a través de la capa Input.
- **Asignación de mayor peso a BPM (TFOpLambda):** Se da mayor importancia a la característica BPM mediante la creación de una matriz de pesos que pondera esta característica antes de procesarla. Se le asignó el doble del peso que a las otras variables.
- **Procesamiento de secuencias con TimeDistributedDense:** Aplica una capa densa con activación ReLU a cada paso de la secuencia temporalmente distribuida, proporcionando procesamiento inicial de las variables secuenciales.

- **LSTM bidireccional:** Procesa la secuencia en ambas direcciones para captar mejor las dependencias temporales en ambos sentidos. Esta capa utiliza `tanh` como activación y tiene el retorno de secuencias activado, lo cual sirve para pasar las secuencias a la próxima capa LSTM.
- **Normalización por lotes:** Aplica normalización por lotes (`BatchNormalization`) para estabilizar y acelerar el entrenamiento, estas se aplican después de las capas LSTM para evitar cualquier sesgo que pueda surgir luego de analizar las secuencias completas.
- **Segunda capa LSTM:** Procesa la secuencia una vez más para obtener la salida final de la secuencia. Esta capa utiliza `tanh` como activación y tiene el retorno de secuencias desactivado, ya que es la última capa LSTM y solo se quiere la salida final de la secuencia, la cual es un resumen de toda la secuencia procesada.
- **Capa de Dropout:** Ayuda a prevenir el sobreajuste al descartar aleatoriamente una fracción de neuronas durante el entrenamiento.
- **Entrada de variables no secuenciales (`InputLayer`):** Capa de entrada que recibe las variables estáticas, como la edad y otros factores que no cambian durante toda la sesión, es decir, que son estáticos.
- **Capa densa para no secuenciales:** Aplica una capa densa con `ReLU` y regularización `L2` a las variables no secuenciales.
- **Capa de atención:** Aplica una capa de atención para darle peso a las partes más relevantes de las secuencias procesadas por el LSTM. Aquí todavía no se agregan las características no secuenciales.
- **Concatenación:** Combina la salida de la capa de atención y las variables no secuenciales.
- **Capa densa final:** Realiza el procesamiento final con una capa densa y `Dropout`, seguida de una capa densa de salida con activación `softmax` para la predicción de tres clases.

El modelo fue entrenado en una máquina con Ubuntu, utilizando Python 3.11 y *TensorFlow* 2.14.0, ejecutado en una *NVIDIA GeForce RTX 3090* de 24GB de VRAM y 125 de RAM, 16 vCPU y 75GB de disco. El proceso de entrenamiento tomó entre 6 y 9 horas, configurado para 12 épocas, con aproximadamente 990 pasos por época. Se empleó *early stopping*, deteniendo el entrenamiento si no se observaba una mejora de 0.0001 en la métrica `val_loss` durante 5 épocas consecutivas, restaurando los mejores resultados. La evaluación se realizaba cada 3 épocas para reducir el tiempo de entrenamiento, en lugar de hacerlo tras cada una. Durante el proceso, se guardaban tanto la mejor versión del modelo como la última, permitiendo reanudar el entrenamiento si era necesario. Este ciclo se repitió para cada uno de los 57 *chunks* de datos, que contienen 500,000 registros por *chunk*, donde algunos completaron las 12 épocas y otros se detuvieron antes gracias al *early stopping*.

### 8.2.3. Evaluación del modelo LSTM

Para la evaluación, se procesaron 8 *chunks* del mismo tamaño que los utilizados en el entrenamiento. El proceso de predicción tomó aproximadamente 30 minutos. Los resultados del reporte de clasificación, mostrados en la Tabla 8.1, indican un *accuracy* aceptable de 0.77. Sin embargo, las predicciones para la clase 1 (sueño ligero) son deficientes, como se observa en la matriz de confusión (Figura 8.49), donde prácticamente no hay distinción en la clasificación de esta clase, que es la más relevante para nuestro análisis.

En la Tabla 8.1 se presenta un reporte de clasificación generado por `classification_report` de *Scikit-Learn*, que incluye tanto el promedio macro como el promedio ponderado. Tanto para este reporte como para el de los siguientes dos modelos, el promedio macro calcula el promedio de cada

métrica de manera uniforme sobre todas las clases, sin considerar el tamaño de cada una, lo que implica que cada clase tiene el mismo peso en el cálculo. En cambio, el promedio ponderado ajusta el cálculo según la cantidad de ejemplos en cada clase, de modo que las clases con más ejemplos tienen mayor influencia en el resultado final.

Dado el desbalance entre las tres clases en nuestro conjunto de datos, el promedio macro podría no reflejar el rendimiento general del modelo de forma precisa. No obstante, considerando que la clase 1 es clave para la detección de somnolencia y, al mismo tiempo, la que presenta menos soporte y métricas más bajas, los resultados obtenidos sugieren que el rendimiento del modelo es insuficiente para nuestros objetivos.

Clase	Precisión	Recall	F1-score	Soporte
0	0.66	0.74	0.70	1,089,685
1	0.37	0.04	0.07	190,237
2	0.82	0.84	0.83	2,293,599
<b>Exactitud</b>	0.77			
<b>Promedio macro</b>	0.62	0.54	0.53	3,573,521
<b>Promedio ponderado</b>	0.75	0.77	0.75	3,573,521

Tabla 8.1: Métricas de clasificación para las distintas clases del modelo LSTM.

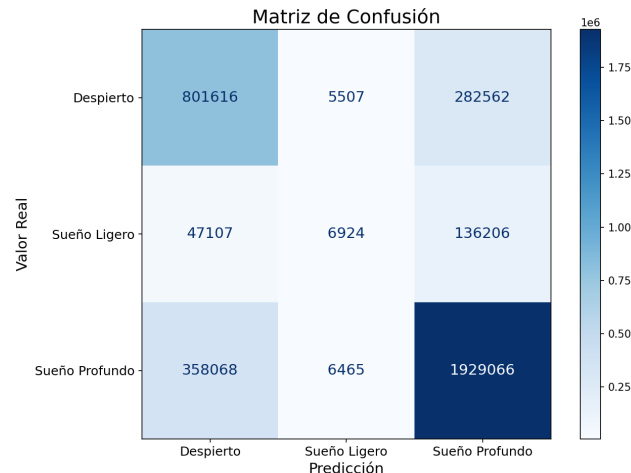


Figura 8.49: Matriz de confusión de las predicciones del modelo LSTM.

## 8.3. Modelo tabular de AutoGluon

### 8.3.1. Preparación de datos para el modelo tabular de AutoGluon y análisis PCA

El modelo tabular de AutoGluon, basado en el *TabularPredictor*, a diferencia del *Long Short-Term Memory (LSTM)* o de *Forecast* de AutoGluon, no tiene memoria. Es decir, no realiza predicciones basadas en una secuencia de datos, sino únicamente a partir de una fila. Entonces, ¿cómo podemos adaptar nuestros datos para un modelo de este tipo? Para esto, agregamos las llamadas variables *Lag*, sobre las siguientes variables: *BPM*, *BPM\_Diff*, *BPM\_Acceleration*, *BPM\_Mean\_Diff*, *BPM\_Mean\_Acc*, *BPM\_Trend* y *Time\_SleepStage*. Estas variables contienen información de las filas anteriores. Por

ejemplo, la variable `BPM_Lag_1` contiene el valor de `BPM` de la fila anterior, `BPM_Lag_2` el de dos filas antes, y así sucesivamente.

Los modelos con memoria, como el `Long Short-Term Memory (LSTM)`, infieren patrones al procesar secuencias completas de datos. Sin embargo, el modelo tabular de AutoGluon no tiene acceso a los datos secuenciales más allá de las variables acumulativas que le proporcionamos. Por ello, hemos añadido promedios móviles de 5 filas, que calculan el promedio, máximo, mínimo, varianza y desviación estándar. Esto da lugar a variables como `BPM_rolling_mean_5`, `BPM_rolling_max_5`, `BPM_rolling_min_5`, `BPM_rolling_std_5` y `BPM_rolling_range_5` (esta última calculada a partir del valor mínimo y máximo). Estas variables ayudan a proporcionar el contexto de las observaciones anteriores dentro de una sola fila.

A estas variables también se les aplica un escalador, el `MinMaxScaler` a las variables de rango fijo y el `StandardScaler` a las variables continuas, que ahora incluyen a todas las variables `Lag` y `Rolling`. Las variables de grupos de edad se mantienen como categóricas.

La adición de las variables `Lag` y `Rolling` aumentó considerablemente el número de variables en el `dataset`, lo que implica un mayor tamaño y complejidad al dárselas al modelo. Para evitar sobrecargar el modelo con un exceso de variables cuya importancia no conocemos completamente, aplicamos un análisis `PCA`, que sirve para reducir la dimensionalidad de los datos y encontrar los componentes relevantes que explican mejor la variabilidad. Para ello, utilizamos un análisis que busca explicar el 90% de los datos. Las variables a las que se aplicó son las secuenciales, y los componentes resultantes reemplazan a estas variables, resultando en 10 componentes que explican un 92.4% de la varianza, como se detalla en la Tabla 8.2 y en la Figura 8.50. También es posible obtener la carga que cada variable tiene en cada componente. Algunas variables contribuyen poco a un componente y mucho a otro; en la Figura 8.51 y en la Tabla 8.3 se muestra la carga máxima (**absoluta**) de cada variable en alguno de los componentes principales. La carga varía entre -1 y 1; cuanto más cercana esté a -1 o 1, mayor es la contribución de la variable a ese componente. En este caso, la tabla muestra los valores absolutos, ya que el signo solo representa si la relación es en la misma dirección o inversa. Esta tabla nos indica que todas las variables tienen importancia en al menos uno de los componentes. El `PCA` se guarda de la misma manera que los escaladores, ya que también fue ajustado con las variables de entrenamiento y debe usarse para transformar las variables de validación.

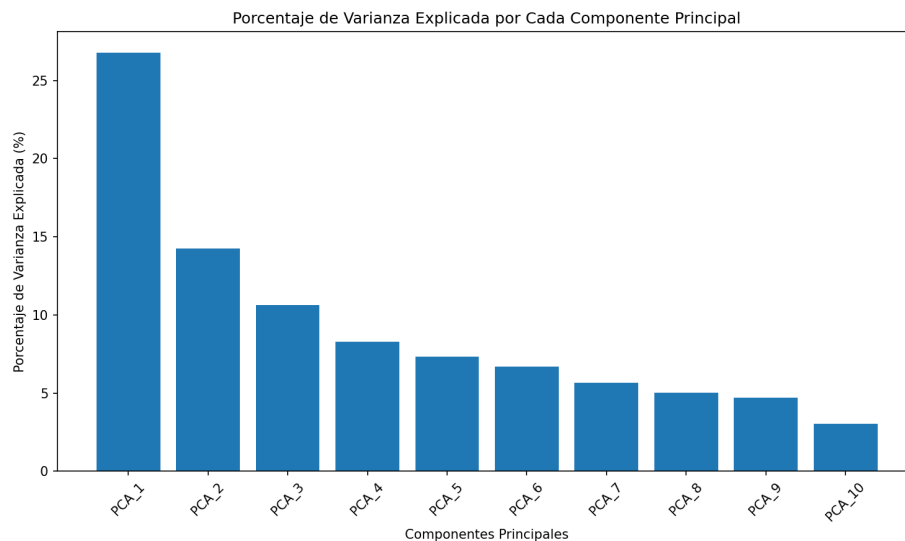


Figura 8.50: Gráfica de barras de los porcentajes de varianza explicada por cada componente principal (PCA).

Componente	Porcentaje de varianza explicada
PCA_1	26.79 %
PCA_2	14.25 %
PCA_3	10.65 %
PCA_4	8.29 %
PCA_5	7.33 %
PCA_6	6.69 %
PCA_7	5.65 %
PCA_8	5.02 %
PCA_9	4.72 %
PCA_10	3.02 %

Tabla 8.2: Porcentaje de varianza explicada por cada componente principal (PCA).

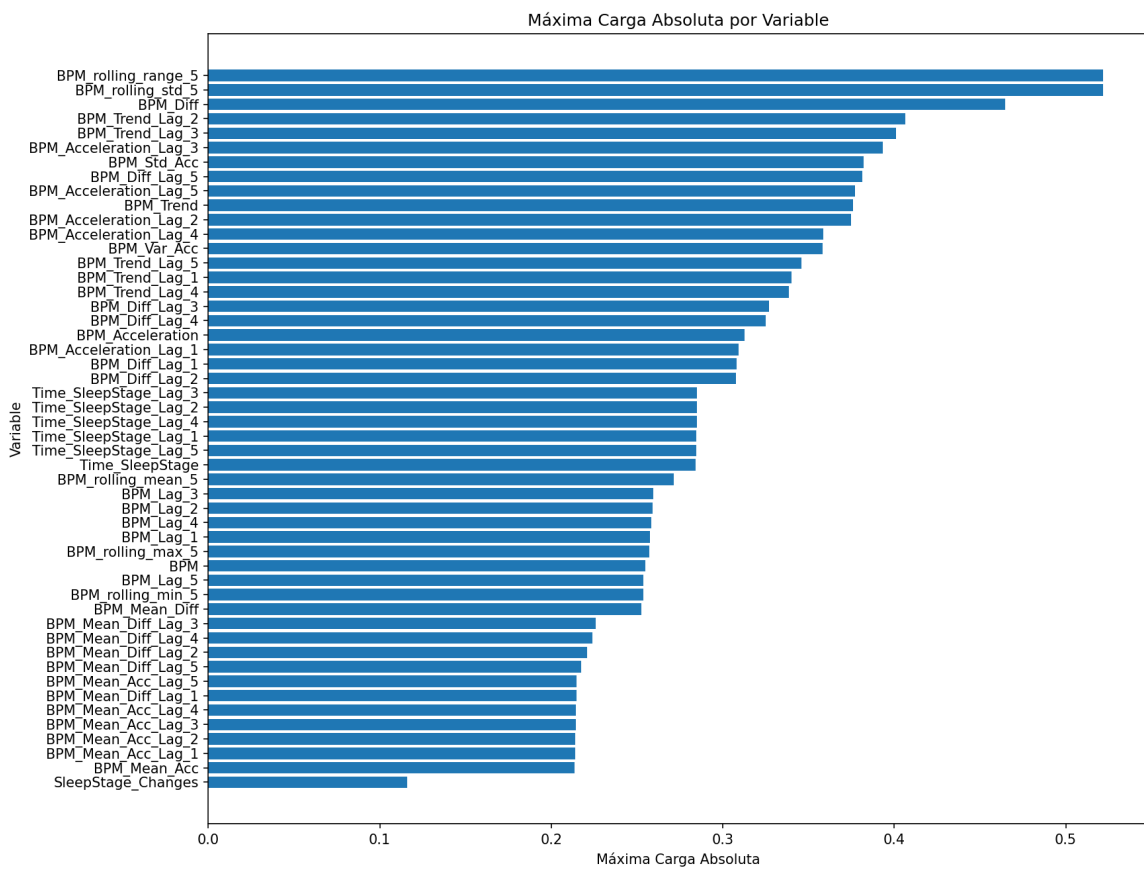


Figura 8.51: Gráfica de barras de las cargas máximas absolutas por variable en al menos un componente principal (ordenada).

Variable	Máxima carga absoluta
BPM	0.254866
BPM_Mean_Acc	0.213597
BPM_Var_Acc	0.358108
BPM_Std_Acc	0.381977
BPM_Diff	0.464838
BPM_Acceleration	0.312580
BPM_Mean_Diff	0.252680
Time_SleepStage	0.284211
SleepStage_Changes	0.115935
BPM_Trend	0.375837
BPM_Lag_1	0.257682
BPM_Diff_Lag_1	0.308074
BPM_Acceleration_Lag_1	0.309057
BPM_Mean_Diff_Lag_1	0.214699
BPM_Mean_Acc_Lag_1	0.213834
BPM_Trend_Lag_1	0.340106
Time_SleepStage_Lag_1	0.284676
BPM_Lag_2	0.259065
BPM_Diff_Lag_2	0.307580
BPM_Acceleration_Lag_2	0.374766
BPM_Mean_Diff_Lag_2	0.220933
BPM_Mean_Acc_Lag_2	0.214060
BPM_Trend_Lag_2	0.406367
Time_SleepStage_Lag_2	0.284921
BPM_Lag_3	0.259377
BPM_Diff_Lag_3	0.326980
BPM_Acceleration_Lag_3	0.393504
BPM_Mean_Diff_Lag_3	0.225872
BPM_Mean_Acc_Lag_3	0.214278
BPM_Trend_Lag_3	0.401142
Time_SleepStage_Lag_3	0.284953
BPM_Lag_4	0.258220
BPM_Diff_Lag_4	0.325136
BPM_Acceleration_Lag_4	0.358687
BPM_Mean_Diff_Lag_4	0.224052
BPM_Mean_Acc_Lag_4	0.214500
BPM_Trend_Lag_4	0.338564
Time_SleepStage_Lag_4	0.284767
BPM_Lag_5	0.253546
BPM_Diff_Lag_5	0.381465
BPM_Acceleration_Lag_5	0.377174
BPM_Mean_Diff_Lag_5	0.217291
BPM_Mean_Acc_Lag_5	0.214708
BPM_Trend_Lag_5	0.345801
Time_SleepStage_Lag_5	0.284365
BPM_rolling_mean_5	0.271580
BPM_rolling_max_5	0.257071
BPM_rolling_min_5	0.253540
BPM_rolling_std_5	0.521580
BPM_rolling_range_5	0.521909

Tabla 8.3: Cargas máximas absolutas por variable en al menos un componente principal.

### 8.3.2. Entrenamiento del modelo tabular de AutoGluon

El entrenamiento del modelo tabular con AutoGluon es mucho más simple, ya que, después de preparar los datos, solo se le pasan como parámetro, se le indica la columna a predecir y otras configuraciones como parámetros del método *fit*. En este método, se puede manipular con mayor profundidad la forma en la que AutoGluon realiza el entrenamiento, y es necesario modificarlo si enfrentamos limitaciones en nuestro sistema, de tiempo o si nuestro *dataset* es demasiado grande.

El parámetro `presets`, como su nombre indica, son preajustes ya establecidos. Por ejemplo, uno llamado `best_quality` busca el mejor rendimiento, a costa de una alta demanda durante el entrenamiento/inferencia. También existen modelos como `high_quality`, que ofrece un mejor balance entre calidad de predicción y velocidad de entrenamiento, y `medium_quality`, que prioriza la velocidad de entrenamiento sobre la calidad de predicción. Además, existen otros presets para prototipado rápido. Alternativamente, podemos ajustar los `hyperparameters`, que permiten modificar parámetros exactos de modelos específicos, seleccionar y excluir modelos, o bien seleccionar un conjunto de modelos más ligeros. Si no se especifican, se utiliza el `default` del *preset* seleccionado.

Otro parámetro modificado fue el `time_limit`, que indica, en segundos, el tiempo máximo permitido para el proceso de entrenamiento. De lo contrario, los modelos se entrenan hasta completarse, lo que en *datasets* grandes como el nuestro puede tardar muchas horas. Este ajuste es crucial dado que el entrenamiento se realizó en máquinas rentadas en la nube; aunque es común que el tiempo de entrenamiento varíe debido a cambios en los recursos disponibles y los procesos intermedios.

Para el entrenamiento de nuestro modelo, se utilizó el preset `best_quality` con los `hyperparameters` por defecto. Se estableció un tiempo límite de 8 horas y se agregó el siguiente parámetro:

```
ag_args_fit={
    'ag.max_memory_usage_ratio': 1.5,
}
```

Este parámetro indica que AutoGluon puede usar hasta 1.5 veces la cantidad de memoria asignada de manera predeterminada. Se observó que la memoria era un requisito elevado en el entrenamiento de nuestro modelo, debido al volumen del *dataset*, por lo que se optó por utilizar una máquina con las especificaciones de la Tabla 8.4. El número de filas de entrenamiento fue 25,563,233, y AutoGluon determinó que el problema era de clasificación multiclase. Durante el entrenamiento, no se experimentaron problemas con modelos detenidos o saltados por falta de memoria, aunque sí por falta de tiempo, ya que algunos modelos requerían más de 20 días para completarse.

El entrenamiento del modelo se completó en 6.73 horas y se determinó que el mejor modelo fue `WeightedEnsemble_L3`, que no difiere mucho de los otros modelos en el top (Tabla 8.5), lo que indica que los modelos fueron bien ajustados y no sufren de sobreajuste (*overfitting*). Sin embargo, el *Weighted Ensemble* destaca por su eficiencia en tiempo, principalmente en `pred_time_val` (Tiempo total de predicción en el conjunto de validación) y muy poca variación en el `pred_time_test` (Tiempo total de predicción en el conjunto de prueba).

Especificación	Detalle
Memoria RAM	128GB DDR5
Procesador	16 vCPUs a 5.7GHz
Disco	50GB
Sistema operativo	Ubuntu
Versión de Python	3.10.15
Versión de AutoGluon	1.1.1

Tabla 8.4: Especificaciones del sistema utilizado para el entrenamiento de AutoGluon.

Modelo	score_test	score_val	pred_time_test (s)	pred_time_val (s)
LightGBMXT_BAG_L2	0.8422	0.8395	3420.85	366.93
WeightedEnsemble_L3	0.8422	0.8395	3421.57	1.72
NeuralNetFastAI_BAG_L1	0.8395	0.8381	1166.63	147.98
WeightedEnsemble_L2	0.8395	0.8381	1167.36	1.61

Tabla 8.5: Resultados del entrenamiento del modelo tabular de AutoGluon

### 8.3.3. Evaluación del modelo tabular de AutoGluon

Para la evaluación del mejor modelo generado por AutoGluon se utilizó el 20% de los datos que se habían reservado para validación, a los cuales se les aplicaron las transformaciones del `StandardScaler`, el `MinMaxScaler` y el PCA. Las predicciones se realizaron por bloques para finalmente realizar la evaluación entre los valores predichos y los reales. La evaluación del rendimiento de AutoGluon se muestra en la Tabla 8.6, donde, a pesar de tener una *accuracy* alta de 0.82, el *balanced accuracy* (que toma en cuenta la tasa de acierto por clase) y el MCC (que considera los verdaderos positivos y negativos, así como los falsos positivos y negativos) muestran valores más bajos. Esto indica que el modelo tiene problemas con las clases minoritarias. Esto se confirma con el reporte de clasificación en la Tabla 8.7, donde se observa que las clases 0 (despierto) y 2 (sueño profundo) tienen una buena tasa de predicción, pero la clase 1 (sueño ligero) muestra un desempeño inferior, probablemente debido a la menor cantidad de soporte.

Métrica	Valor
Accuracy	0.8243
Balanced Accuracy	0.6524
MCC (Matthews Corr Coef)	0.6346

Tabla 8.6: Resultados de las métricas de evaluación del modelo tabular de AutoGluon.

Clase	Precision	Recall	F1-score	Soporte
0	0.86	0.68	0.76	2251090
1	0.59	0.33	0.43	409840
2	0.82	0.94	0.88	4548689
<b>Accuracy</b>	0.82			7209619
<b>Macro Avg</b>	0.76	0.65	0.69	7209619
<b>Weighted Avg</b>	0.82	0.82	0.82	7209619

Tabla 8.7: Reporte de clasificación del modelo tabular de AutoGluon.

En la Figura 8.52 se muestra la matriz de confusión para la evaluación de este modelo, lo que

refuerza lo mencionado anteriormente: la clase 1 (sueño ligero) es poco precisa en sus predicciones, aunque mejora en comparación con los resultados del modelo LSTM (8.49). En esta matriz se observa una mayor diferencia en los falsos positivos de la clase 0 (despierto), lo cual es importante, ya que estos son los casos en los que el usuario se está quedando dormido, pero el modelo sigue prediciendo que está despierto.

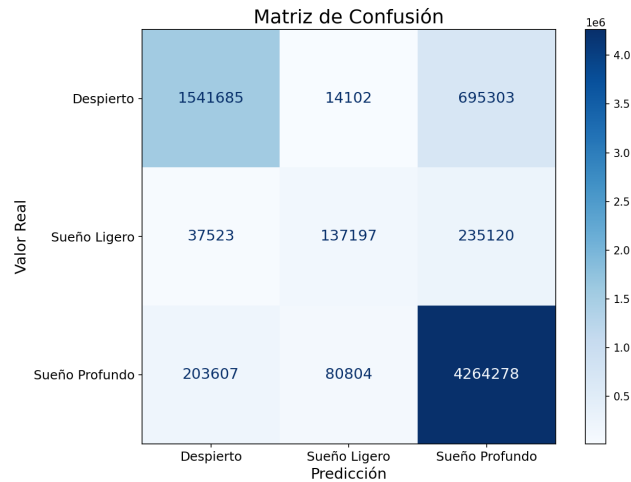


Figura 8.52: Matriz de confusión de las predicciones del modelo tabular de AutoGluon.

El modelo tabular de AutoGluon también genera una tabla de importancia de las características (Tabla 8.8 y Figura 8.53), donde los valores más altos indican una mayor influencia en el modelo. Las desviaciones estándar pequeñas reflejan una estabilidad en la importancia de cada característica, mientras que los valores bajos de *p-value* (aproximadamente cuando son menores a 0.05 o 0.01) sugieren que las variables tienen un efecto significativo en el modelo. En este caso, todas las características, excepto `BPM_count`, se consideran relevantes; esta última muestra baja importancia y un *p-value* alto, indicando que su influencia es limitada.

Por otro lado, características como `PCA_4` y `PCA_5`, aunque presentan una baja importancia, tienen un *p-value* bajo, lo cual implica una relación estadísticamente significativa con el resultado. Sin embargo, dada su baja importancia, es razonable considerar su eliminación, o en su defecto, excluir las variables que contribuyen más a estos componentes. Para `PCA_4` y `PCA_5`, las variables con mayor peso son `BPM_Diff`, `BPM_Acceleration` y sus respectivas variables de retraso (*Lag*), además de algunas versiones de `BPM_Trend` en forma de *Lag*, como se detalla en la Tabla 8.9 o en la Figura 8.54 donde se muestran las cargas absolutas y ordenadas.

En este proyecto, no se procedió con la eliminación de las variables mencionadas debido a que ello implicaría volver a entrenar el modelo, un proceso que demandaría tiempo y recursos adicionales. Además, es importante comprender el papel de estas variables en el funcionamiento del modelo, ya que fueron seleccionadas inicialmente por su relevancia lógica en el contexto del proyecto. Aunque la eliminación de variables con baja importancia podría mejorar ligeramente el rendimiento, no es una necesidad estricta para alcanzar los objetivos del modelo en este caso. Por ello, se decidió mantener dichas variables en el conjunto de datos para evaluar su impacto en las predicciones y conservar el propósito original de probar los modelos con todas las características propuestas.

Característica	Importancia	Desv. Est.	p-value
PCA_3	0.14068	0.002270	8.1293e-09
Time	0.10600	0.006280	1.4716e-06
PCA_2	0.08404	0.004832	1.3056e-06
PCA_1	0.07484	0.006563	7.0229e-06
PCA_8	0.06920	0.003873	1.1725e-06
Age_Binned_<25	0.04624	0.003480	3.8193e-06
Age_Binned_>60	0.02580	0.002561	1.1503e-05
Age_Binned_40_60	0.02152	0.001404	2.1635e-06
height	0.02128	0.003534	8.8054e-05
stress	0.01660	0.002126	3.1593e-05
sleep_quality	0.00924	0.002488	5.7378e-04
fatigue	0.00876	0.003235	1.8783e-03
Age_Binned_25_40	0.00768	0.000642	5.8009e-06
PCA_9	0.00600	0.001517	4.5073e-04
sleep_duration	0.00520	0.001530	8.0367e-04
PCA_10	0.00472	0.001635	1.4813e-03
PCA_6	0.00432	0.002460	8.5758e-03
PCA_7	0.00228	0.001254	7.6350e-03
PCA_5	0.00136	0.001228	3.4237e-02
PCA_4	0.00116	0.001004	3.0550e-02
BPM_count	0.00104	0.001729	1.2486e-01

Tabla 8.8: Importancia de características (*leaderboard*) para el modelo tabular de AutoGluon.

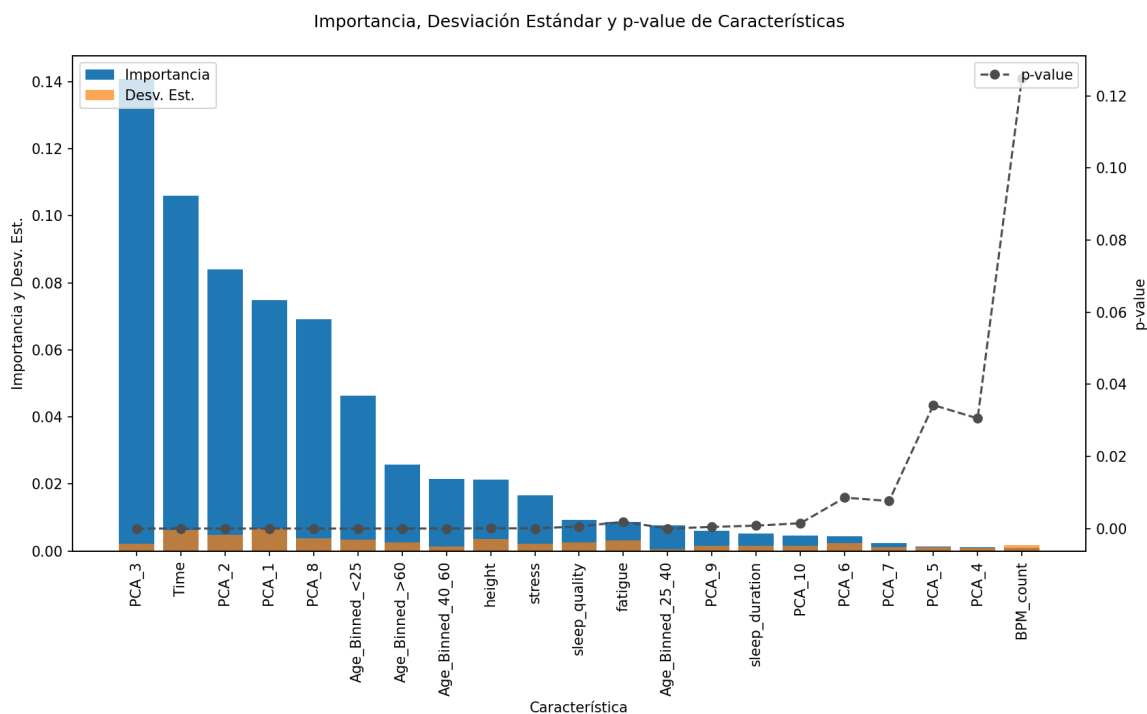


Figura 8.53: Gráfica de la importancia, desviación estándar y p-value de las características para el modelo tabular de AutoGluon.

<b>Variable</b>	<b>PCA_4</b>	<b>PCA_5</b>
BPM	-0.001394	0.013141
BPM_Mean_Acc	0.005062	-0.013884
BPM_Var_Acc	-0.001528	0.002727
BPM_Std_Acc	-0.001664	0.002726
BPM_Diff	-0.074448	-0.213412
BPM_Acceleration	-0.176521	-0.312580
BPM_Mean_Diff	-0.005965	0.028118
Time_SleepStage	0.003252	-0.007559
SleepStage_Changes	-0.002250	0.009818
BPM_Trend	0.094358	0.088550
BPM_Lag_1	0.032325	0.109827
BPM_Diff_Lag_1	0.218103	0.304638
BPM_Acceleration_Lag_1	0.309057	0.265767
BPM_Mean_Diff_Lag_1	0.036569	0.150098
BPM_Mean_Acc_Lag_1	0.005116	-0.013765
BPM_Trend_Lag_1	-0.039277	0.339343
Time_SleepStage_Lag_1	0.003289	-0.007559
BPM_Lag_2	-0.066483	-0.028182
BPM_Diff_Lag_2	-0.294095	-0.135780
BPM_Acceleration_Lag_2	-0.374766	-0.136314
BPM_Mean_Diff_Lag_2	-0.087940	-0.023551
BPM_Mean_Acc_Lag_2	0.004810	-0.014497
BPM_Trend_Lag_2	-0.079676	0.041087
Time_SleepStage_Lag_2	0.003274	-0.007504
BPM_Lag_3	0.066744	0.033330
BPM_Diff_Lag_3	0.326980	0.090105
BPM_Acceleration_Lag_3	0.393504	-0.103787
BPM_Mean_Diff_Lag_3	0.079861	0.053965
BPM_Mean_Acc_Lag_3	0.005318	-0.014318
BPM_Trend_Lag_3	0.042979	0.023705
Time_SleepStage_Lag_3	0.003276	-0.007503
BPM_Lag_4	-0.081384	-0.007486
BPM_Diff_Lag_4	-0.325136	0.262150
BPM_Acceleration_Lag_4	-0.301422	0.358687
BPM_Mean_Diff_Lag_4	-0.106732	0.002675
BPM_Mean_Acc_Lag_4	0.004786	-0.014612
BPM_Trend_Lag_4	-0.079050	-0.140123
Time_SleepStage_Lag_4	0.003282	-0.007432
BPM_Lag_5	0.065913	-0.126242
BPM_Diff_Lag_5	0.174379	-0.332269
BPM_Acceleration_Lag_5	0.128013	-0.230612
BPM_Mean_Diff_Lag_5	0.078738	-0.147096
BPM_Mean_Acc_Lag_5	0.005403	-0.014828
BPM_Trend_Lag_5	0.092966	-0.222887
Time_SleepStage_Lag_5	0.003251	-0.007379

Tabla 8.9: Carga de cada una de las variables en los componentes principales PCA\_4 y PCA\_5.

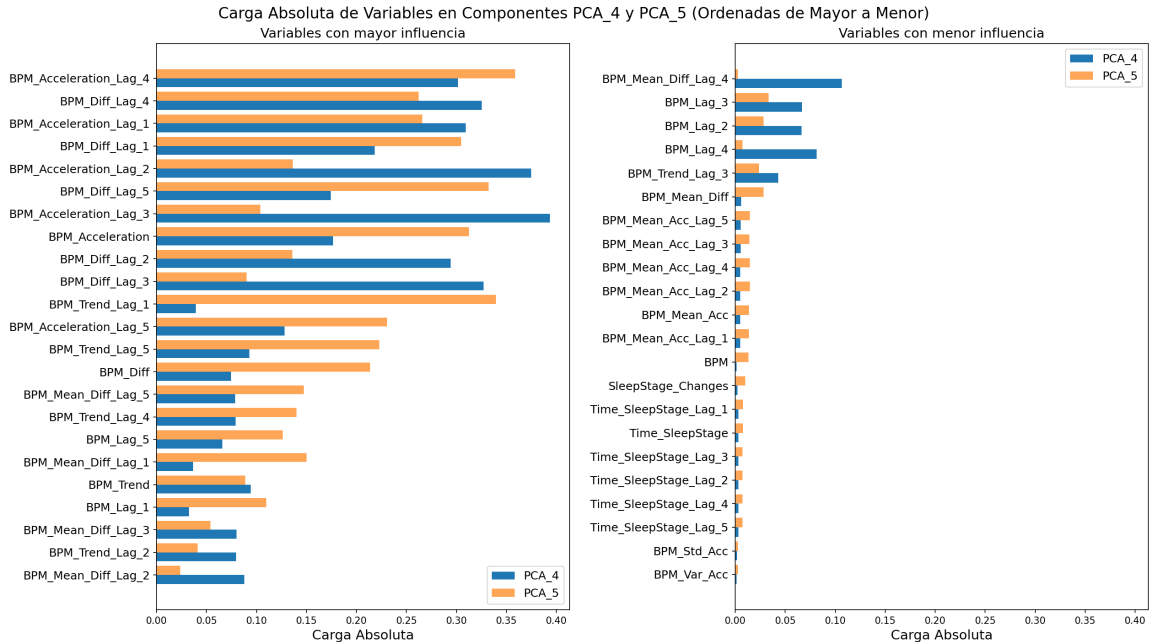


Figura 8.54: Gráfica de la carga absoluta de cada una de las variables en los componentes principales PCA\_4 y PCA\_5 (ordenada).

## 8.4. Modelo forecast de AutoGluon

### 8.4.1. Preparación de datos para el modelo forecast de AutoGluon

El modelo forecast de AutoGluon funciona de manera similar al LSTM en el sentido de que puede observar una secuencia durante el entrenamiento, por lo tanto, no requiere de las variables *Lag* ni *Rolling* creadas para el modelo de tabular de AutoGluon. Sin embargo, es necesario generar secuencias de un tamaño específico. Para ello, se agrupan las sesiones en secuencias de 50 filas, esto debido a que en un principio se habían hecho de 24, pero AutoGluon no tomaba en cuenta aquellas menores a 49. A diferencia del modelo LSTM, estas secuencias no se dividen entre *features* secuenciales y no secuenciales. En su lugar, se agrega una variable llamada `sequence_id` y se almacenan todas las secuencias en un único archivo de entrenamiento, que representa el 80% de las sesiones. El 20% restante se reserva para la validación, ambos en formato `parquet`.

### 8.4.2. Entrenamiento del modelo forecast de AutoGluon

#### Interpolación de las secuencias a 5 segundos

Antes de pasar los datos al modelo forecast de AutoGluon, se realizaron algunas optimizaciones y correcciones. En primer lugar, se ajustaron los tipos de datos, cambiando la columna `time` a `integer` y `SleepStage` a `int8`. Luego, se renombró la columna `time` a `timestamp`, convirtiéndola al tipo `datetime`, y se eliminaron duplicados en la combinación (`sequenceId`, `timestamp`). Los registros del *dataset* `PMDData` no estaban espaciados uniformemente en intervalos de 5 segundos, debido a irregularidades en las sesiones de ritmo cardíaco.

El proceso de interpolación asegura que todas las secuencias tengan intervalos consistentes de

5 segundos, un requisito fundamental para el correcto funcionamiento de los modelos de series temporales de AutoGluon. Esta interpolación se realiza sobre la columna `timestamp` como índice, aplicando la función `resample` con el parámetro “5S” para crear nuevos índices cada 5 segundos si no existen, y luego se utiliza un relleno hacia adelante (`ffill`) para completar esos nuevos índices con el valor más reciente disponible.

Los grupos interpolados se guardan en archivos de formato `parquet` en `chunks` de 500,000 filas, debido al alto consumo de memoria. Aunque, posteriormente, es necesario cargar todo el conjunto de datos, ya que AutoGluon no permite realizar entrenamientos por partes.

## Creación de `TimeSeriesDataFrame` y `TimeSeriesPredictor`

Para entrenar el modelo `forecast` de AutoGluon, los datos deben estar en un formato específico, lo que se logra al pasarlos por un `TimeSeriesDataFrame`. Este formato requiere que se especifique una columna de identificación (ID), razón por la cual se agregó `sequence_id` durante la creación de las secuencias, además de indicar la columna `timestamp`. Esto permite que AutoGluon gestione las secuencias temporales de manera eficiente, estructurándolas mediante un índice múltiple basado en el ID de secuencia y el `timestamp`.

El nuevo `DataFrame` se pasa como parámetro al `TimeSeriesPredictor`, lo que crea el predictor. En este paso, se especifica la columna objetivo a predecir, así como el parámetro `prediction_length`, que define cuántas filas a futuro se van a predecir. En este caso, se configuró en 12, lo que significa predecir el próximo minuto.

El predictor de series temporales, al igual que el tabular, permite el uso de `presets`. En este caso, se utilizó `best_quality` para obtener la mejor predicción posible, con un límite de tiempo de 8 horas de entrenamiento, especificado por el parámetro `time_limit`. El entrenamiento se realizó en una máquina con las mismas especificaciones que el modelo tabular, las cuales se detallan en la Tabla [8.4](#).

## Inferencias para el entrenamiento del modelo `forecast` de AutoGluon

AutoGluon infirió que la frecuencia de las series de tiempo es “5S”, es decir, 5 segundos, razón por la cual se realizó la interpolación. Después del proceso de filtrado que realiza AutoGluon, el conjunto de datos de entrenamiento contiene 28,635,756 filas. El conjunto incluye 467,692 series temporales individuales. La longitud mediana de estas series es de 50 filas, con un mínimo de 37 y un máximo de 8,008. Cabe resaltar que el conjunto que se le dió a AutoGluon contenía 28,665,510 filas y 468,621 secuencias, de las cuales solo se eliminaron 929 que tenían un tamaño menor a 37 filas. Anteriormente cuando las secuencias generadas eran de tamaño 24, AutoGluon eliminaba el 83% de los datos que se le daban.

## Resultados del entrenamiento del modelo `forecast` de AutoGluon

El entrenamiento tuvo una duración de 7.87 horas. En la Tabla [8.10](#) se observan los resultados de diversos modelos utilizados en el entrenamiento. El modelo `WeightedEnsemble` presenta el mejor desempeño, con un `score_val` de -0.0728. Sin embargo, esto no significa que sea la única opción; el modelo `TemporalFusionTransformer` muestra un puntaje muy cercano y una mejora significativa en el tiempo de predicción. Además, el modelo `RecursiveTabular` destaca por tener el tiempo de predicción más rápido y ser el tercero con mejor puntaje, aunque su precisión se aleja un poco de la de `WeightedEnsemble`. La elección entre estos modelos depende de la relación que busquemos entre precisión y tiempo de predicción.

Modelo	Score val	Pred time val	Fit time marginal
WeightedEnsemble	-0.072830	1239.04	2683.95
TemporalFusionTransformer	-0.072952	521.69	2086.91
RecursiveTabular	-0.085197	64.04	900.34
CrostonSBA	-0.087095	653.30	752.65
SeasonalNaive	-0.087111	542.25	598.54
AutoETS	-0.088220	1179.38	1165.28
DynamicOptimizedTheta	-0.090008	1173.58	1294.46
DirectTabular	-0.093212	939.76	2034.85
PatchTST	-0.121754	193.65	488.29

Tabla 8.10: Resultados del entrenamiento del modelo forecast de AutoGluon.

### 8.4.3. Evaluación del modelo forecast de AutoGluon

Para la evaluación de este modelo, se realiza el mismo proceso de interpolación que se utilizó antes de entrenar el modelo, solo que en este caso para el 20 % de los datos que son para validación. La puntuación de evaluación, usando la métrica WQL o Pérdida de Cuantiles Ponderada, es de -0.0758; esta métrica se utiliza en modelos de pronóstico probabilístico y evalúa qué tan bien coinciden los cuantiles predichos por el modelo con los valores reales, asignando mayor peso a ciertos cuantiles. La métrica en sí es negativa, ya que mide la pérdida, y los valores más cercanos a 0 son mejores. El resultado obtenido indica que el modelo tiene una precisión moderada, aunque tampoco es un mal valor.

Para el reporte de clasificación de este modelo se utilizó solamente el 10 % de ese 20 % que era el conjunto de validación, esto ya que si se utilizaba todo el conjunto, la evaluación tomaría hasta 11 horas según estimaciones. Esta evaluación muestra mejoras significativas respecto a los anteriores, con una precisión general de 0.92. Se puede decir que tiene una buena predicción, pero el modelo tabular de AutoGluon también presentaba una precisión alta y, sin embargo, mostraba lo contrario para las predicciones de la clase 1 (sueño ligero). Una diferencia interesante y notoria es que, en esta evaluación, la clase 1 ya no tiene una gran diferencia en la cantidad de soporte respecto a las otras dos clases. Aunque en conjunto las tres clases tienen una menor cantidad de muestras respecto a las evaluaciones de los otros modelos, se obtuvieron mejores resultados. Se podría intuir que existe sobreajuste (*overfitting*), pero por los valores consistentes de *F1-score* y *recall*, además de la poca diferencia en el *score val* de la Tabla 8.10, no parece que el modelo esté sobreajustado.

Clase	Precisión	Recall	F1-score	Soporte
0.0	0.95	0.89	0.92	42396
1.0	0.50	0.70	0.58	7115
2.0	0.96	0.96	0.96	88153
<b>Exactitud</b>	0.92 (137664 muestras)			
<b>Promedio macro</b>	0.60	0.64	0.61	137664
<b>Promedio ponderado</b>	0.93	0.92	0.93	137664

Tabla 8.11: Reporte de clasificación del modelo forecast de AutoGluon.

En la matriz de confusión de la Figura 8.55 se observa que la clase 1 (sueño ligero) tiene un mayor número de predicciones correctas; sin embargo, este número no es suficiente para superar la suma de las predicciones incorrectas, lo que explica el valor elevado de *recall* en esta evaluación. Es importante destacar que anteriormente se realizó una evaluación con el modelo forecast de AutoGluon utilizando secuencias de 24 filas, en la cual se eliminó el 83 % de los datos iniciales. Esto hizo que el entrenamiento del modelo fuera más ligero, aunque la evaluación se realizó con un conjunto de

datos de mayor soporte. En esa evaluación, la clase 1 contaba con 64,344 ejemplos, obteniendo una precisión de 0.8, un *recall* de 0.91 y un *F1-score* de 0.85. Esto podría indicar que al proporcionar una mayor cantidad de datos de entrenamiento, los modelos tienden a generalizar más, lo que puede dificultar la identificación precisa de la clase 1. Además, un modelo más ligero permite realizar evaluaciones de mayor tamaño en menor tiempo.

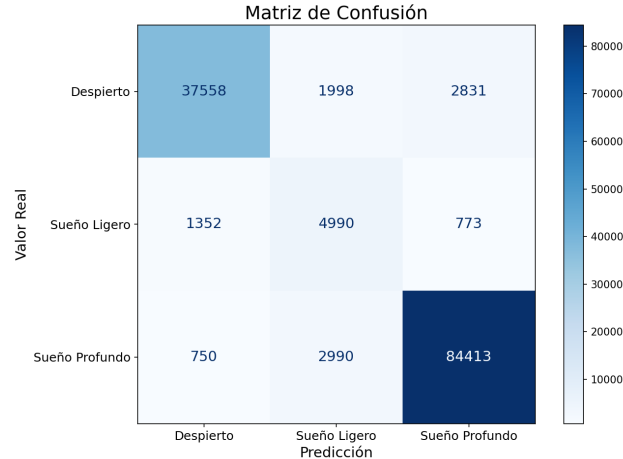


Figura 8.55: Matriz de confusión de las predicciones del modelo de forecast de AutoGluon.

## 8.5. Discusión sobre los resultados de los modelos

Es importante señalar que, aunque los modelos utilizaron el mismo conjunto de datos para su entrenamiento y validación, cada uno requirió un preprocesamiento particular para adaptar los datos a su estructura específica. Este preprocesamiento generó variaciones entre los modelos en aspectos como la cantidad de ejemplos por clase, el tamaño de los conjuntos de entrenamiento y validación, y la selección de sesiones de sueño. Por ejemplo, en el modelo LSTM, los datos se dividieron en un 80 % para entrenamiento, un 10 % para validación durante el entrenamiento y un 10 % para validación post-entrenamiento. En contraste, los modelos de *AutoGluon* (tanto *Tabular* como *Forecast*) se dividieron en un 80 % para entrenamiento, dejando un 20 % para validación post-entrenamiento, permitiendo que *AutoGluon* gestione internamente los datos en la fase de entrenamiento.

Otro aspecto diferenciador es el uso de características adicionales en el modelo tabular de *AutoGluon*. Debido a que este modelo no conserva memoria de las filas previas en una secuencia, fue necesario introducir variables *Lag* y *Rolling* para proporcionar información contextual sobre las filas anteriores en cada secuencia, replicando así una forma básica de memoria.

La generación de secuencias para los modelos LSTM y *Forecast* de *AutoGluon* también difiere. En el modelo LSTM, las secuencias se crearon con un tamaño de ventana de 24 filas y un paso de 1 fila, es decir, cada fila del conjunto de datos formaba parte de una secuencia de 24 filas consecutivas. En cambio, en el modelo forecast de *AutoGluon*, se crearon secuencias de 50 filas, con paso de 50 filas, es decir las secuencias no llegan a repetir valores, no se solapan. Además, se aplicó interpolación a una frecuencia de 5 segundos para mantener la consistencia temporal que requiere el `TimeSeriesDataFrame` de *AutoGluon*. Esto resultó en una menor cantidad de soporte para cada clase en el modelo forecast, tanto en el entrenamiento como en la evaluación.

En la Tabla [8.12](#) se observa que el modelo forecast de *AutoGluon* obtuvo mejores resultados en *recall* y *F1-score*, aunque no en precisión, destacando especialmente en la clase 1 (sueño ligero), que es fundamental para el propósito del proyecto. Esta consistencia entre clases se aprecia con mayor

claridad en la Figura 8.56, donde las barras del modelo forecast de AutoGluon presentan tamaños similares. La menor cantidad de soporte en la evaluación (137,664 ejemplos frente a millones en los otros modelos) podría no reflejar completamente el desempeño del modelo, pero se considera válida en relación al rendimiento esperado en pruebas reales.

A pesar de estas diferencias en el preprocesamiento y la estructura de los modelos, las pocas mejoras obtenidas en el modelo forecast de *AutoGluon* en la clase 1 son una ventaja significativa para el proyecto, ya que refuerza la hipótesis de que el ritmo cardíaco es un indicador útil para la detección de somnolencia. Sin embargo, debido a las limitaciones de los conjuntos de datos utilizados (provenientes de SHHS y PMData), es posible que aún exista sesgo en los resultados. Se recomienda realizar pruebas adicionales con datos más variados y extensos para confirmar que el modelo pueda mantener su rendimiento en escenarios reales y con un conjunto de datos más amplio y representativo.

Modelo	Clase	Precisión	Recall	F1-score	Soporte
LSTM	0	0.66	0.74	0.70	1,089,685
	1	0.37	0.04	0.07	190,237
	2	0.82	0.84	0.83	2,293,599
	<b>Exactitud general</b>	0.77			<b>3,573,521</b>
AutoGluon Tabular (WeightedEnsemble)	0	0.86	0.68	0.76	2,251,090
	1	0.59	0.33	0.43	409,840
	2	0.82	0.94	0.88	4,548,689
	<b>Exactitud general</b>	0.82			<b>7,209,619</b>
AutoGluon Forecast (WeightedEnsemble)	0	0.95	0.89	0.92	42,396
	1	0.50	0.70	0.58	7,115
	2	0.96	0.96	0.96	88,153
	<b>Exactitud general</b>	0.92			<b>137,664</b>

Tabla 8.12: Comparación de métricas de clasificación y soporte para los tres modelos utilizados.

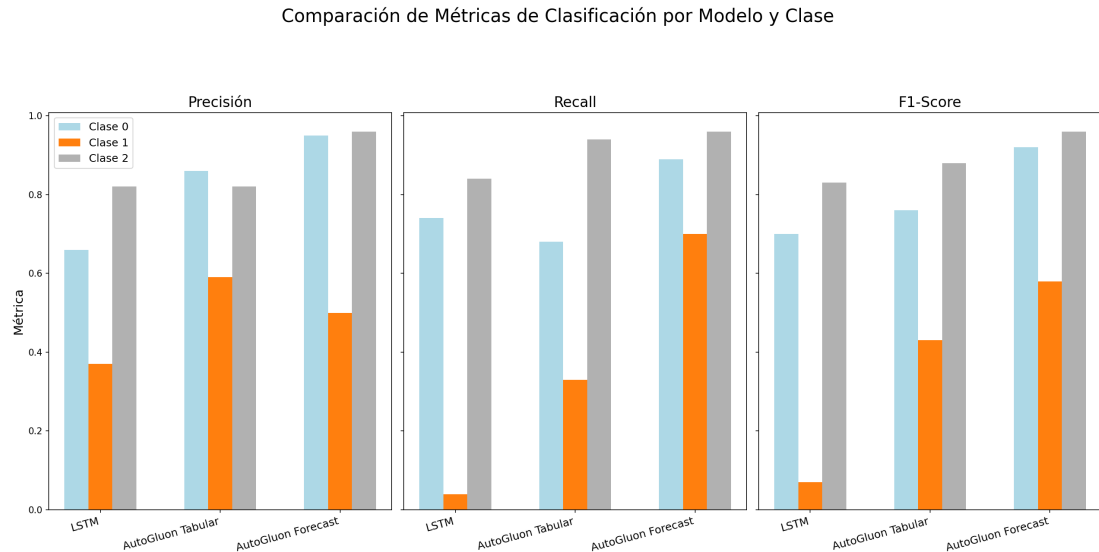


Figura 8.56: Gráficas de comparación para precisión, recall y F1-score entre los modelos utilizados.

## 8.6. Sistema en tiempo real

Siguiendo los objetivos planteados en este proyecto, se desarrolló un sistema de predicción en tiempo real utilizando un *smartwatch* para la captura continua del ritmo cardíaco. Este sistema integra los modelos de predicción entrenados previamente con el objetivo de detectar somnolencia de forma temprana, brindando alertas en tiempo real que pueden ser útiles en escenarios de conducción.

### 8.6.1. Hardware

En el desarrollo del sistema de predicción de somnolencia en tiempo real, se optó por utilizar un *smartwatch* Xiaomi Mi Band 5 para la toma de datos de ritmo cardíaco. No obstante, el sistema no está limitado a este dispositivo en particular, ya que la arquitectura está diseñada para ser independiente del hardware utilizado. Esto significa que cualquier dispositivo capaz de medir y transmitir datos de frecuencia cardíaca en tiempo real podría integrarse al sistema, siempre que cumpla con los requerimientos de precisión y frecuencia de muestreo necesarios para los modelos de predicción.

La Xiaomi Mi Band 5, utilizada en este caso, es capaz de recolectar constantemente los valores de ritmo cardíaco con una frecuencia aproximada de 25.52 registros por minuto, según las pruebas empíricas mostradas en la Tabla 8.13, usando el sensor de Fotopletismografía (PPG), lo cual es adecuado para este proyecto. Su conectividad Bluetooth y capacidad de sincronizar datos en tiempo real la convierten en una opción accesible y eficiente para la recolección de datos.

Nº	Tiempo en minutos	Cantidad de registros de BPM	Frecuencia (tomas por minuto)
1	1	28	28
2	14	361	25.79
3	2	51	25.50
4	6	151	25.17
5	10	225	22.50
6	7	183	26.14
<b>Total</b>			<b>25.52</b>

Tabla 8.13: Frecuencia de toma de datos de la Mi Band 5.

### 8.6.2. Software para la obtención del ritmo cardíaco

Para conectar el *smartwatch* Xiaomi Mi Band 5 al sistema de predicción, se utilizó el repositorio `miband-5-heart-rate-monitor` [14]. Este repositorio proporciona una forma de conectar el *smartwatch* mediante tecnologías web, ya que la aplicación se ejecuta en el navegador y hace uso del API de *Bluetooth* del navegador para conectar con el *smartwatch*. Además, presenta una interfaz muy simple y eficiente.

Al ejecutar el programa, se solicita una *Auth key*, como se muestra en la Figura 8.57. Esta llave se puede obtener utilizando diferentes métodos. Una vez obtenida la llave, se permite la conexión con el *smartwatch* y se puede observar la frecuencia cardíaca enviada en pantalla. Sin embargo, es necesario enviar este dato a nuestro servidor, que ejecuta los modelos de predicción, por lo que se realizaron algunos cambios en el código.

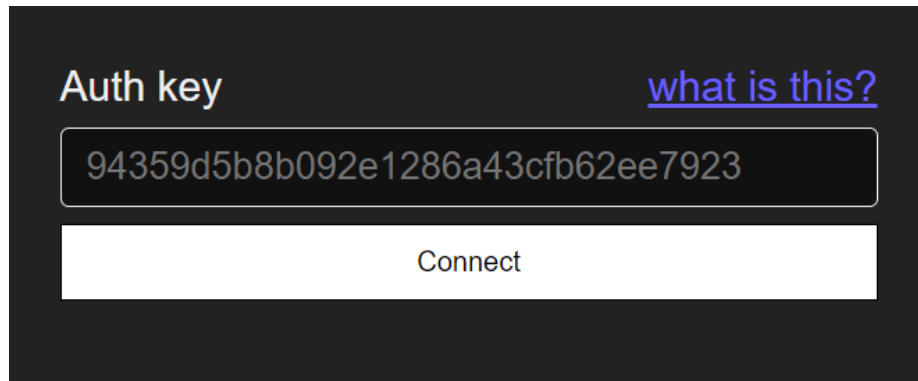


Figura 8.57: Pantalla de inicio del cliente para conectar con el *smartwatch* Xiaomi Mi Band 5.

### Modificaciones para conectar con el servidor

La conexión entre este programa y el servidor que recibe los datos del ritmo cardíaco se estableció mediante *WebSockets*, donde el programa actúa como cliente. Cada vez que se recibe un dato del *smartwatch*, este se envía al servidor, que ejecuta el modelo de predicción. Si se detecta una alerta, esta se envía de vuelta al cliente.

Durante el desarrollo del programa, se identificó que en ocasiones este perdía la conexión con el *smartwatch*. Para solventarlo, se implementó un intervalo que intenta reconectar y solicitar los valores si no se recibe un dato en 30 segundos. Esta reconexión también afectaba la comunicación con el servidor, por lo que se añadió un mecanismo de ‘ping-pong’ para mantener la conexión activa. Además, se incluyó la funcionalidad de recibir y mostrar una alerta cuando se predice un estado de sueño ligero o profundo desde el servidor, mostrando en pantalla el texto y, en la consola, la información de la fecha y hora de la alerta, como se ve en la Figura [8.59](#).

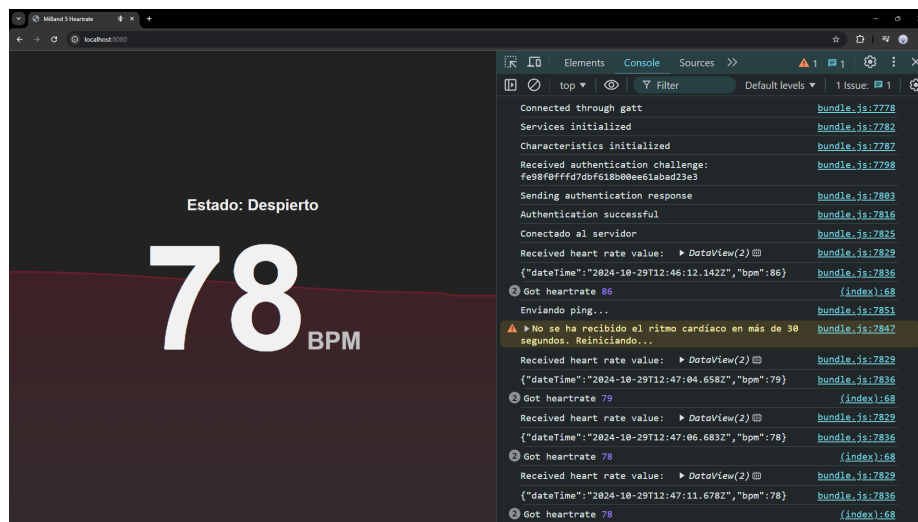


Figura 8.58: Pantalla que muestra el ritmo cardíaco y sin alerta.

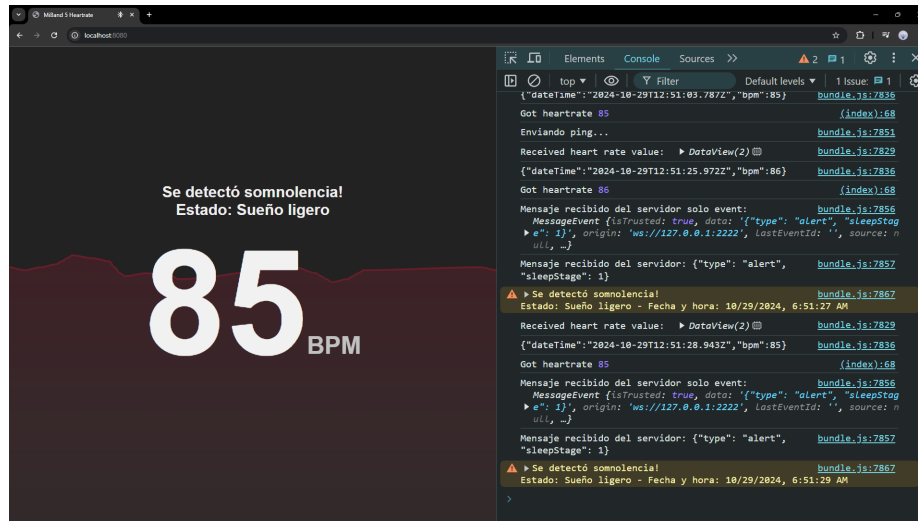


Figura 8.59: Pantalla que muestra el ritmo cardíaco y alerta de somnolencia

### 8.6.3. Servidor que ejecuta el modelo

Dependiendo del modelo que se ejecute, el servidor ajusta ligeramente su funcionamiento. Por ejemplo, entrega los datos como secuencia para los modelos Long Short-Term Memory (LSTM) y AutoGluon Forecast, o aplica Principal Component Analysis (PCA) para el modelo tabular de AutoGluon. Sin embargo, la estructura general se mantiene consistente para los tres modelos.

El servidor comienza solicitando las variables no secuenciales del usuario, como la edad, altura en centímetros, duración del último sueño en horas, nivel actual de fatiga, calidad del último sueño y nivel de estrés, los tres últimos en una escala del 1 al 5. También se cargan los escaladores específicos y el modelo correspondiente.

Este servidor se ejecuta de manera indefinida, permaneciendo en un bucle infinito que escucha continuamente las conexiones del cliente. Implementa un mecanismo de “ping-pong” para mantener la conexión activa, de modo que, si recibe un mensaje de “ping” del cliente, responde con un “pong”. Si el mensaje es un objeto *JSON*, extrae el dato de la frecuencia cardíaca (BPM) y la fecha y hora (*dateTime*). Antes de realizar la predicción, actualiza la variable *time*. Si han pasado más de 15 minutos, elimina las filas más antiguas y reinicia la variable *BPM\_count*. También se calculan todas las variables derivadas del BPM. La predicción no se realiza hasta que exista al menos una secuencia de 24 filas, asegurando que las variables derivadas sean coherentes.

El servidor realiza la predicción devolviendo el *SleepStage* predicho. Si este es diferente al último registrado, se reinicia la variable *Time\_SleepStage*, se incrementa *SleepStage\_Changes*, y se envía una alerta al cliente.

Finalmente, la fila actualizada con la predicción se agrega al *DataFrame* que almacena las secuencias, se guarda el archivo, y se incrementa el contador para la variable *BPM\_count*.

#### Predicción para el modelo LSTM

Para este modelo, es necesario crear secuencias de 24 filas. Se realiza una copia del *DataFrame* original para almacenar los registros sin alterar sus valores. Posteriormente, las características (*features*) se transforman utilizando los escaladores *StandardScaler* y *MinMaxScaler*. El *DataFrame*

puede contener más de 24 filas, ya que almacena datos de hasta 15 minutos. A partir de estos, se generan subsecuencias que cumplen con el tamaño requerido por el modelo. Las características secuenciales y no secuenciales se separan para ser usadas como entradas, según lo requiere nuestro modelo *Long Short-Term Memory (LSTM)*, y luego se procede a realizar la predicción.

Dado que se generan varias predicciones debido a las múltiples subsecuencias, se utiliza una función que asigna pesos a cada predicción, donde las más recientes tienen un mayor peso que las más antiguas. A partir de estos valores ponderados, se calcula un promedio. La función aplicada es [8.1](#), donde  $\hat{y}$  es la predicción final,  $\hat{y}_i$  son las predicciones individuales,  $n$  es el total de predicciones, y  $\lambda$  es el factor de decaimiento. Con un factor de decaimiento de 0.75, los pesos asignados a cada predicción se muestran en la Figura [8.60](#). Si el valor promedio supera un 0.7 de probabilidad para la clase 1 (sueño ligero) o la clase 2 (sueño profundo), se devuelve la predicción de la nueva fase.

$$\hat{y} = \frac{\sum_{i=0}^{n-1} (\lambda^{n-i-1} \cdot \hat{y}_i)}{\sum_{i=0}^{n-1} \lambda^i} \quad (8.1)$$

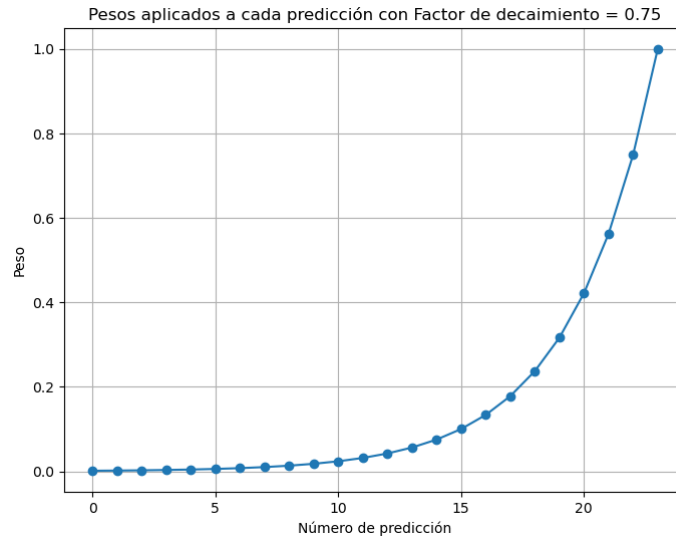


Figura 8.60: Gráfica de los pesos que se aplican en la función de predicción.

### Predicción para el modelo tabular de *AutoGluon*

En la función de predicción del modelo tabular de *AutoGluon*, se utiliza únicamente la última fila del `DataFrame`. A esta fila se le aplican los escaladores correspondientes y la transformación *Principal Component Analysis (PCA)*, reemplazando las características originales por los componentes principales obtenidos. A continuación, se procede a realizar la predicción. Este modelo permite obtener directamente la probabilidad de predicción para cada clase, lo que facilita la comparación de valores. Si la probabilidad supera el 0.7 para la clase 1 (sueño ligero) o la clase 2 (sueño profundo), se devuelve la predicción de la nueva fase.

### Predicción para el modelo forecast de *AutoGluon*

El modelo forecast de *AutoGluon* puede recibir secuencias de mayor tamaño, no solo de 24 filas. Por ello, se enviará una copia de todas las filas disponibles hasta ese momento. De igual forma, se aplican los escaladores correspondientes, se agrega la columna `sequence_id`, se renombra la variable

`time` a `timestamp`, y se realiza la interpolación a intervalos de 5 segundos (`'5S'`). Luego, se realiza una limpieza de valores nulos y se verifica que las columnas tengan los tipos de datos correctos para crear el `TimeSeriesDataFrame`. Posteriormente, se revisa nuevamente que los índices sean los correctos, y se elimina la variable `item_id` de las características, dejándola únicamente como índice. A continuación, se procede a hacer la predicción, que devuelve el valor más cercano a la clase predicha para cada una de las 24 predicciones futuras.

Para esta predicción, se utiliza la misma función de pesos [8.1](#) otorgando mayor importancia a las predicciones más distantes en el tiempo que a las más cercanas. Finalmente, se calcula el promedio ponderado, y dado que las predicciones no son enteros, se redondea el valor resultante para obtener la clase más cercana de la predicción ponderada.



Detectar la somnolencia sigue siendo una tarea compleja. Si bien existen rasgos fisiológicos en los que podemos apoyarnos, alcanzar una precisión extremadamente alta es difícil. Esto se debe a la naturaleza del problema, en la cual no existe una diferencia claramente definida entre estar despierto, somnoliento, en fase de sueño ligero o sueño profundo. Por lo tanto, la capacidad para hacer estas detecciones siempre será algo difusa. Además, cuando el problema exige una predicción con minutos o incluso segundos de anticipación, como es el caso al estar conduciendo, una predicción inexacta puede conllevar consecuencias graves.

El ritmo cardíaco, por sí solo, no es un indicador sólido de somnolencia. Esto lo podemos observar en las gráficas de las sesiones, como las Figuras 8.9 y 8.11, en las que se aprecian diferentes patrones: tendencias a subir, bajar, o incluso casos en los que el ritmo cardíaco se mantiene en un rango constante antes de entrar en una fase de sueño ligero, sin una señal clara de cambio. Además se descubrió un patrón común en las gráficas de ritmo cardíaco, donde se observa un pico de subida del ritmo cardíaco antes del cambio de fase a sueño ligero.

Al transformar la señal **Electrocardiografía (ECG)** a valores de frecuencia cardíaca expresados en **Beats Per Minute (BPM)**, se pierden estas características, las cuales podrían haber ayudado a detectar mejor las fases del sueño.

El desafío principal de este proyecto radica en que el sujeto se encuentra en una situación muy específica: intentar mantenerse despierto a pesar de tener somnolencia, lo que se conoce como **Maintenance of Wakefulness Test (MWT)**. Esta es una prueba en la que el sujeto se coloca en una condición que puede considerarse “extrema”, por lo que se requiere su consentimiento informado. Esta particularidad complicó la obtención de datos para el entrenamiento, ya que no fue posible encontrar un conjunto que cumpliera con todas las características requeridas. Aunque los *datasets* *PMData* y *SHHS* contienen algunos casos que sugieren que se dio esta condición, no siempre es así, lo que puede introducir ruido en nuestros modelos y afectar las predicciones.

Existen modelos especializados en tareas específicas, como el LSTM y el modelo forecast de *AutoGluon*, pero también es posible adaptar datos para entrenar modelos que, en principio, no están diseñados para dicha tarea y aun así obtener predicciones satisfactorias. Un ejemplo de esto es la adición de variables *lag* y *rolling* en las entradas del modelo tabular de *AutoGluon*, lo cual resultó en mejores resultados que con el modelo LSTM.

En cuanto al modelo LSTM, aunque la lógica de su estructura parece adecuada —con la sepa-

ración de características secuenciales y no secuenciales, una mayor ponderación de la variable *Beats Per Minute* (BPM), y la inclusión de una capa de atención para las características secuenciales, como se muestra en la Figura 8.48—, los resultados no fueron los esperados. Esto se atribuye en gran medida al desbalance en la cantidad de datos de soporte para cada clase, siendo la clase 1 (sueño ligero) la que presenta menor cantidad de ejemplos. Una situación similar ocurrió en el modelo tabular de *AutoGluon*, aunque cambió en el modelo forecast de *AutoGluon*. Aunque la clase 1 sigue teniendo menos soporte, la diferencia es menos pronunciada, lo que pudo haber contribuido a obtener mejores resultados y mayor consistencia en la precisión de las predicciones entre las distintas clases. El primer objetivo específico se cumplió, ya que proponía entrenar los modelos con al menos 100,000 ejemplos por clase; según nuestro *dataset* total, la clase con menor cantidad contaba con 2,028,215 ejemplos, como se observa en la Figura 8.45.

El modelo forecast de *AutoGluon* obtuvo los mejores resultados, con un *F1-score* de 0.92 para la clase 0, 0.58 para la clase 1 y 0.96 para la clase 2. Esta métrica es especialmente relevante considerando el desbalance de clases en el dataset. Aunque el *F1-score* de la clase 1, que representa la detección de somnolencia, alcanzó un valor de 0.58, los resultados muestran que el modelo es capaz de generar predicciones útiles. Por lo tanto, se considera cumplido el segundo objetivo específico, ya que se lograron predicciones que pueden aportar valor dentro de un sistema complementario de detección de fatiga.

Durante el desarrollo del sistema, se evaluaron los modelos entrenados para verificar su capacidad de realizar predicciones en intervalos de tiempo cortos entre tomas de ritmo cardíaco. Los tres modelos demostraron que es viable generar predicciones con diferencias entre tomas de tan solo 3 a 5 segundos. En los modelos con memoria, como el LSTM y el *Forecast* de *AutoGluon*, las secuencias procesadas no son excesivamente largas: 24 filas para el LSTM y entre 24 y 200 para el *Forecast* de *AutoGluon*. Esto permite concluir que se cumple el tercer objetivo específico, el cual proponía que las predicciones se realizaran cada 5 segundos.

Durante el desarrollo del sistema, no se logró predecir la somnolencia con anticipación o en el momento exacto. Tampoco se observó que los modelos hicieran predicciones de somnolencia a los pocos minutos después de haber comenzado la toma de datos. Generalmente, al analizar las probabilidades, se nota cómo estas cambian con el tiempo, acercándose a la predicción de otra clase (de clase 0 a clase 1). Sin embargo, en ocasiones, después de observar una tendencia hacia la clase 1, la probabilidad vuelve a reforzar la clase 0. Es importante determinar un punto de corte en el que las probabilidades de cada clase sean suficientes para tomar una decisión. Por ejemplo, que el modelo indique una probabilidad de 0.6 para la clase 1 y de 0.2 para la clase 0 no implica necesariamente que la predicción sea lo suficientemente sólida como para considerarse una detección de somnolencia. Esta configuración depende de qué tan estrictos queramos ser en la detección, aunque si se establece un valor elevado, corremos el riesgo de no detectar la somnolencia a tiempo.

Los resultados de este proyecto confirman que la frecuencia cardíaca sí guarda relación con los cambios en las fases de sueño y el estado de somnolencia, pero por sí sola no es un indicativo fuerte y preciso para detectar un cambio. Los modelos generados en este proyecto no son aptos para ser utilizados en un contexto de conducción real y se considera que incluso aplicando mejoras, el ritmo cardíaco no debe utilizarse como única fuente de predicción de somnolencia. Este sistema tiene la ventaja de ser adaptable a aplicaciones que obtienen la frecuencia de ritmo cardíaco por medio de sensores no invasivos, lo que favorece la aceptación de los usuarios. Esto satisface el objetivo general del proyecto, donde a partir del sistema de predicción en tiempo real y de la evaluación de los modelos, se analizó la influencia y viabilidad del ritmo cardíaco para la detección de somnolencia.

Dado que la frecuencia cardíaca no es un indicador preciso de las fases del sueño, se recomienda siempre combinarla con otras características fisiológicas, sensores, estrategias o métodos que se complementen entre sí. Incluso los *smartwatches* no proporcionan un análisis completo de las fases del sueño basándose únicamente en la frecuencia cardíaca; a menudo combinan un acelerómetro para detectar los movimientos y, a partir de ahí, se realiza un análisis posterior, no en tiempo real ni predictivo.

El ritmo cardíaco es un buen complemento para otros indicadores más sólidos, como los visuales (parpadeos, bostezos o cabeceos) o los comportamentales (en el caso de la conducción, el mantenimiento en el carril, la distancia entre vehículos o la aceleración y frenado del coche). Además, su uso es sencillo y no invasivo, lo que facilita su incorporación a sistemas más complejos de detección de somnolencia.

En este proyecto se justificaron las variables utilizadas; sin embargo, para futuros estudios, podría explorarse la incorporación de variables adicionales, considerando siempre la confidencialidad de los datos en función de la sensibilidad de la información solicitada al usuario, así como la facilidad para obtenerla. No se recomienda pedir al usuario que proporcione variables difíciles de obtener en el momento o que estén estrechamente relacionadas con estudios médicos, ya que es posible que no tenga conocimiento de ellas. También podría considerarse la eliminación de aquellas variables evaluadas como de “poca importancia” según la propia evaluación de los modelos.

Además, el uso de datos más detallados sobre el ritmo cardíaco, como la variabilidad del ritmo cardíaco Heart Rate Variability (HRV), en lugar de solo la frecuencia, podría aportar información adicional relevante al modelo, así como los datos de la señal de Electrocardiografía (ECG) de forma directa. Aunque estos últimos dependen del dispositivo que se utilice para la toma de muestras y, actualmente, no son del todo accesibles ni poco invasivos, en el futuro, con el avance de la tecnología, podría considerarse su incorporación si se vuelven más accesibles y menos invasivos.

Dado que es complicado recrear condiciones reales de fatiga al conducir, incluso en entornos controlados como las pruebas de mantenimiento de la vigilia (Maintenance of Wakefulness Test (MWT)), se sugiere investigar más sobre este tipo de pruebas o filtrar los datos en función de otras características cuando se detecte fatiga. También es recomendable aumentar la diversidad en los datos; para este proyecto se utilizaron dos *datasets* que, si bien se complementaban, los datos de *SHHS* terminaron predominando, lo cual puede no ser ideal para el caso específico, ya que incluían muestras de personas de entre 70 y 90 años, lo que introduce sesgos no deseados.

El *Fine-Tuning* es una técnica útil, en la que se toma un modelo ya entrenado con una gran cantidad de datos y luego se ajusta con un conjunto más pequeño y específico. Esto podría ser una excelente adición a un sistema ya establecido, permitiendo que el modelo se adapte a las características de un usuario específico con cada sesión, mejorando la precisión de las predicciones.

En este proyecto se desarrolló un modelo LSTM, y también se empleó *AutoGluon*, que ofreció los mejores resultados. Sin embargo, aunque *AutoGluon* brinda buenos resultados y permite ajustar los parámetros de los modelos, no debe descartarse la posibilidad de replicar manualmente el mejor modelo o conjunto de modelos. Esto no solo facilita un mejor entendimiento del funcionamiento del modelo, sino que también permite optimizarlo para que pueda ejecutarse en dispositivos con recursos limitados, como teléfonos inteligentes, microcomputadoras como *Raspberry Pi* o *Arduino*, o incluso en el propio *smartwatch*. Además, hasta la fecha, *AutoGluon* no admite *fine-tuning*, lo que limita su capacidad de adaptación en tiempo real.

Asimismo, sería valioso implementar técnicas de interpretación de modelos como *SHAP* o *LIME*, que permiten visualizar qué variables están influyendo más en las predicciones. En este proyecto se hizo algo similar utilizando el *leaderboard* generado por *AutoGluon*, lo cual ayudó a descartar variables como el tipo de persona (A o B) o el género del sujeto desde las primeras etapas. También es recomendable explorar otras variables que puedan aportar valor, además de las métricas comunes derivadas del Beats Per Minute (BPM), como el promedio, la varianza o la desviación estándar.

- 
- 
- [1] Abd El-Nabi, Samy *et al.*: *Machine learning and deep learning techniques for driver fatigue and drowsiness detection: a review*. Multimedia Tools and Applications, 83:1–37, June 2023.
  - [2] Arakawa, Toshiya: *A review of heartbeat detection systems for automotive applications*. Sensors, 21(18), 2021, ISSN 1424-8220. <https://www.mdpi.com/1424-8220/21/18/6112>.
  - [3] AutoGluon Team: *Autogluon: Automl for text, tabular, image, and time series data*, 2023. <https://auto.gluon.ai>, Accedido el 15 de octubre de 2024.
  - [4] Boneva, Roumiana S. *et al.*: *Higher heart rate and reduced heart rate variability persist during sleep in chronic fatigue syndrome: A population-based study*. Autonomic Neuroscience, 137(1):94–101, 2007, ISSN 1566-0702. <https://www.sciencedirect.com/science/article/pii/S1566070207004468>.
  - [5] Castaneda, Denisse *et al.*: *A review on wearable photoplethysmography sensors and their potential future applications in health care*. International Journal of Biosensors & Bioelectronics, 4(4):195–202, 2018. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6426305>, Epub 2018 Aug 6.
  - [6] Dorfman Furman, Gabriela *et al.*: *Early detection of falling asleep at the wheel: A heart rate variability approach*. In *Computers in Cardiology*, volume 35, pages 1109 – 1112, October 2008.
  - [7] Federal Motor Carrier Safety Administration: *Large truck and bus crash facts 2021*. FMCSA Data and Statistics, 2023. <https://www.fmcsa.dot.gov/safety/data-and-statistics/large-truck-and-bus-crash-facts-2021>.
  - [8] FitBit: *Get sleep log by date range*, n.d. <https://dev.fitbit.com/build/reference>, FitBit Developer Build Reference.
  - [9] Fujiwara, Koichi *et al.*: *Heart rate variability-based driver drowsiness detection and its validation with eeg*. IEEE Transactions on Biomedical Engineering, 66(6):1769–1778, 2019.
  - [10] Google Brain: *Tensorflow: An open-source library for machine learning and ai*, 2023. <https://www.tensorflow.org>, Accedido el 15 de octubre de 2024.

- [11] Grace, R. *et al.*: *A drowsy driver detection system for heavy vehicles*. In *17th DASC. AIAA/IEEE/SAE. Digital Avionics Systems Conference. Proceedings (Cat. No.98CH36267)*, volume 2, pages I36/1–I36/8 vol.2, 1998.
- [12] Hossain, Md. Yousuf and George, Fabian Parsia: *Iot based real-time drowsy driving detection system for the prevention of road accidents*. In *2018 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*, volume 3, pages 190–195, 2018.
- [13] Hyundai Mobis: *Hyundai mobis m.vics and smart cabin secures safe and sound autonomous driving*. *hyundai motor group*. Technical report, Hyundai, 2022. <https://www.hyundaimotorgroup.com/story/CONT000000000043965>.
- [14] Jaapp: *miband-5-heart-rate-monitor*. <https://github.com/Jaapp-/miband-5-heart-rate-monitor>, 2023. Último acceso: 16 de octubre de 2024.
- [15] Jo, Sang Ho, Kim, Jin Myung, and Kim, Dong Kyoo: *Heart rate change while drowsy driving*. *J Korean Med Sci*, 34(8), Mar 2019. <https://doi.org/10.3346/jkms.2019.34.e56>.
- [16] Johns, Murray W: *Assessing the drowsiness of drivers*. [https://www.mwjohns.com/wp-content/uploads/2017/05/assessing\\_drowsiness\\_of\\_drivers.pdf](https://www.mwjohns.com/wp-content/uploads/2017/05/assessing_drowsiness_of_drivers.pdf), Unpublished report commissioned by VicRoads, Melbourne, 2001.
- [17] Kryger, Meir, Roth, Thomas, and Dement, William C. (editors): *Foreword*. Elsevier, sixth edition edition, 2017, ISBN 978-0-323-24288-2. <https://www.sciencedirect.com/science/article/pii/B978032324288200177X>.
- [18] Lee, Hyeonjeong, Lee, Jaewon, and Shin, Miyoung: *Using wearable ecg/ppg sensors for driver drowsiness detection based on distinguishable pattern of recurrence plots*. *Electronics*, 8(2), 2019, ISSN 2079-9292. <https://www.mdpi.com/2079-9292/8/2/192>.
- [19] Li, Gang and Chung, Wan Young: *Detection of driver drowsiness using wavelet analysis of heart rate variability and a support vector machine classifier*. *Sensors*, 13(12):16494–16511, 2013, ISSN 1424-8220. <https://www.mdpi.com/1424-8220/13/12/16494>.
- [20] Mabry, Jessica Erin *et al.*: *Unravelling the complexity of irregular shiftwork, fatigue and sleep health for commercial drivers and the associated implications for roadway safety*. *International Journal of Environmental Research and Public Health*, 19(22), 2022, ISSN 1660-4601. <https://www.mdpi.com/1660-4601/19/22/14780>.
- [21] MathWorks Community: *Obtaining heart rate from ECG signal*, 2023. <https://la.mathworks.com/matlabcentral/answers/2117156-obtaining-heart-rate-from-ecg-signal>, [Online forum post], accessed on 29/10/2024.
- [22] National Sleep Research Resource: *Sleep heart health study (shhs)*, 2023. <https://sleepdata.org/datasets/shhs>, Accedido el 15 de octubre de 2024.
- [23] Pacheco, Danielle and Rehman, Anis: *Drowsy driving vs. drunk driving: How similar are they?* Sleep Foundation, November 3 2023. <https://www.sleepfoundation.org/sleep-deprivation/drowsy-driving>, Accessed: 2024-10-10.
- [24] Pacheco, Danielle and Rehman, Anis: *Sleep latency*. <https://www.sleepfoundation.org/how-sleep-works/sleep-latency>, 2023. <https://www.sleepfoundation.org/how-sleep-works/sleep-latency>, Accessed: 2023-11-6.
- [25] Parák, Jakub and Havlík, Jan: *Ecg signal processing and heart rate frequency detection methods*. Department of Circuit Theory, Faculty of Electrical Engineering Czech Technical University in Prague, November 2011.

- [26] Philip, Pierre *et al.*: *Maintenance of wakefulness test scores and driving performance in sleep disorder patients and controls*. International Journal of Psychophysiology, 89(2):195–202, 2013, ISSN 0167-8760. <https://www.sciencedirect.com/science/article/pii/S0167876013001505>, Psychophysiology and Cognitive Neuroscience of Sleep and Sleep Disorders.
- [27] Programa de las Naciones Unidas para el Desarrollo (PNUD): *Análisis sobre hechos de tránsito en Guatemala, durante 2023*. PNUD Publicaciones, Octubre 2 2024. <https://www.undp.org/es/guatemala/publicaciones/analisis-sobre-hechos-de-transito-en-guatemala-durante-2023>.
- [28] Radha, Mustafa *et al.*: *Sleep stage classification from heart-rate variability using long short-term memory neural networks*. Scientific Reports, 9(1):14149, 2019. <https://doi.org/10.1038/s41598-019-49703-y>.
- [29] Rahman, Ashiqur, Hriday, Mamun Bin Harun, and Khan, Riasat: *Computer vision-based approach to detect fatigue driving and face mask for edge computing device*. Heliyon, 8(10):e11204, 2022. <https://doi.org/10.1016/j.heliyon.2022.e11204>.
- [30] Russell, Allison, Heneghan, Conor, and Venkatraman, Subbu: *Investigation of an estimate of daily resting heart rate using a consumer wearable device*. medRxiv, 2019. <https://www.medrxiv.org/content/early/2019/10/18/19008771>.
- [31] Saini, Vandna and Saini, Rekha: *Driver drowsiness detection system and techniques: A review*. In (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (3), 2014. <https://api.semanticscholar.org/CorpusID:9022866>.
- [32] Segerstrom, Suzanne C. and Nes, Lise Solberg: *Heart rate variability reflects self-regulatory strength, effort, and fatigue*. Psychological Science, 18(3):275–281, 2007. <https://doi.org/10.1111/j.1467-9280.2007.01888.x>, PMID: 17444926.
- [33] Shigetomo, Mitani: *Development of the in-vehicle pulse sensor. about the pulse rate estimate technique from a microwave sensor*. Technical report, OMRON, 2019. <https://www.omron.com/global/en/technology/omrontechnics/vol50/006.html>.
- [34] Sinnapolu, GiriBabu and Alawneh, Shadi: *Intelligent wearable heart rate sensor implementation for in-vehicle infotainment and assistance*. Internet of Things, 12:100277, 2020, ISSN 2542-6605. <https://www.sciencedirect.com/science/article/pii/S2542660520301104>.
- [35] Sissons, Beth: *What is the difference between type a and type b personalities?* Medical News Today, 2024. <https://www.medicalnewstoday.com/articles/type-a-vs-type-b>.
- [36] Strickland, Eliza: *3 ways ford cars could monitor your health > ford is experimenting with car features that could help drivers with diabetes, heart problems, and more*. Technical report, IEEE Spectrum, 2017. <https://spectrum.ieee.org/3-ways-ford-cars-could-monitor-your-health>.
- [37] Summer, Jay and Singh, Abhinav: *Maintenance of wakefulness test (mwt)*. <https://www.sleepfoundation.org/sleep-studies/mwt>, 2022. <https://www.sleepfoundation.org/sleep-studies/mwt>, Accessed: 2023-11-6.
- [38] Sun, Yu and Thakor, Nitish: *Photoplethysmography revisited: From contact to noncontact, from point to imaging*. IEEE Transactions on Biomedical Engineering, 63(3):463–477, 2016. <https://pubmed.ncbi.nlm.nih.gov/26390439>, Epub 2015 Sep 15.
- [39] Thambawita, Vajira *et al.*: *Pmdata: a sports logging dataset*. In *Proceedings of the 11th ACM Multimedia Systems Conference, MMSys '20*, page 231–236, New York, NY, USA, 2020. Association for Computing Machinery, ISBN 9781450368452. <https://doi.org/10.1145/3339825.3394926>.

- [40] Vural, Esra *et al.*: *Drowsy driver detection through facial movement analysis*. In Lew, Michael *et al.* (editors): *Human-Computer Interaction*, pages 6–18, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg, ISBN 978-3-540-75773-3.
- [41] Wadhwa, Anmol and Roy, Sanjiban Sekhar: *10 - driver drowsiness detection using heart rate and behavior methods: A study*. In Lee, Kun Chang *et al.* (editors): *Data Analytics in Biomedical Engineering and Healthcare*, pages 163–177. Academic Press, 2021, ISBN 978-0-12-819314-3. <https://www.sciencedirect.com/science/article/pii/B9780128193143000112>.
- [42] Waxenbaum, Jonathan A, Reddy, Vinayak, and Varacallo, Matthew: *Anatomy, autonomic nervous system*. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing, 2023. Updated 2023 Jul 24. Available from: <https://www.ncbi.nlm.nih.gov/books/NBK539845/>.
- [43] Yu, Xun: *Real-time nonintrusive detection of driver drowsiness*. Center for Transportation Studies, University of Minnesota, 2009. <https://hdl.handle.net/11299/97650>, Retrieved from the University Digital Conservancy.

**Adaboost**

*Adaboost* es un algoritmo de aprendizaje automático que combina varios clasificadores débiles para formar un clasificador fuerte. Funciona ajustando el peso de cada clasificador basado en su precisión, enfocándose en los errores que comete. Se utiliza comúnmente en tareas de clasificación. [10](#)

**BPM**

*Beats Per Minute* o Latidos por Minuto es una medida de la frecuencia cardíaca, que indica el número de latidos del corazón en un minuto. Es un parámetro comúnmente utilizado para monitorear el estado físico y emocional de una persona. Se considera que un BPM dentro del rango normal indica un ritmo cardíaco saludable, mientras que variaciones significativas pueden ser indicativas de fatiga o estrés. [7](#) [14](#) [31](#) [36](#) [37](#) [40](#) [43](#) [71](#) [72](#) [74](#)

**CNN**

Las redes neuronales convolucionales (*Convolutional Neural Networks* en inglés) son un tipo de red neuronal profunda especialmente efectiva para el procesamiento y análisis de imágenes y datos secuenciales. Utilizan convoluciones para extraer características espaciales y temporales de los datos de entrada. [8](#) [13](#)

**DL**

*Deep Learning* o Aprendizaje Profundo, es un subcampo del [Machine Learning \(ML\)](#) que utiliza redes neuronales artificiales con múltiples capas para modelar patrones complejos en grandes volúmenes de datos. [14](#)

**ECG**

La *Electrocardiografía* es una técnica que mide la actividad eléctrica del corazón. Se utiliza en estudios de sueño y monitoreo de la salud para registrar la frecuencia y el ritmo cardíaco, y es útil para detectar irregularidades que podrían estar relacionadas con la fatiga o trastornos del sueño. [11](#) [14](#) [18](#) [30](#) [31](#) [37](#) [71](#) [73](#)

**EDF**

Formato de archivo estándar para almacenar datos biológicos y de señales médicas, conocido como *European Data Format*. Este tipo de archivo es ampliamente utilizado en investigaciones y

aplicaciones clínicas para almacenar señales fisiológicas como ECG, EEG, y otras, permitiendo una estructura consistente y fácil de analizar para datos continuos y anotaciones. [31](#)

## EEG

El *Electroencefalograma* es una técnica no invasiva utilizada para registrar la actividad eléctrica del cerebro. Se utiliza en estudios de sueño para detectar las diferentes fases del sueño y la somnolencia al analizar las ondas cerebrales. [11](#) [14](#) [15](#)

## EMG

La *Electromiografía* es una técnica que registra la actividad eléctrica de los músculos. En estudios del sueño, se utiliza para monitorear la relajación muscular y detectar el tono muscular, lo que ayuda a distinguir las diferentes fases del sueño. [11](#)

## ENS

El sistema nervioso entérico es una división del sistema nervioso autónomo que controla el funcionamiento del tracto gastrointestinal. A menudo es referido como el “segundo cerebro” debido a su capacidad para operar de manera autónoma y regular funciones digestivas como el movimiento intestinal y la secreción de enzimas. [11](#)

## EOG

La *Electrooculografía* es una técnica utilizada para medir los movimientos oculares mediante electrodos colocados alrededor de los ojos. Esta técnica es particularmente útil en estudios del sueño para identificar las fases de sueño REM, que se caracterizan por movimientos oculares rápidos. [11](#)

## Forecast

Modelo de predicción utilizado para series temporales. En *AutoGluon*, se refiere a un predictor que utiliza datos de series temporales para hacer predicciones futuras a partir de secuencias de datos previas. [51](#)

## GDPR

*General Data Protection Regulation*, una normativa de la Unión Europea que establece pautas para la recopilación y procesamiento de datos personales de individuos dentro del Espacio Económico Europeo, con el objetivo de proteger la privacidad y los derechos de los usuarios. [21](#)

## HMM

Los modelos ocultos de Markov (*Hidden Markov Models* en inglés) son modelos estadísticos utilizados para representar sistemas que evolucionan con el tiempo de manera probabilística. Son útiles en el reconocimiento de patrones, particularmente en secuencias temporales, como en la detección de fases del sueño o somnolencia. [8](#)

## HR

El ritmo cardíaco (*Heart Rate* en inglés) es el número de veces que el corazón late por minuto. Es una medida importante en la monitorización de la salud y el bienestar, ya que varía según la actividad física, las emociones y las condiciones de salud de la persona. [7](#) [9](#) [36](#)

## HRV

La variabilidad de la frecuencia cardíaca (*Heart Rate Variability*) es una medida de las fluctuaciones en los intervalos de tiempo entre latidos consecutivos del corazón. Refleja la actividad del sistema nervioso autónomo, específicamente el equilibrio entre el Sistema Nervioso Simpático (SNS) y el Sistema Nervioso Parasimpático (PNS). Un HRV alto indica una mayor capacidad del cuerpo para adaptarse a cambios y estrés, mientras que un HRV bajo puede ser señal de fatiga, estrés o problemas de salud. Se utiliza en estudios de somnolencia porque los cambios en HRV pueden anticipar episodios de sueño o fatiga. [7](#) [8](#) [11](#) [13](#) [14](#) [36](#) [73](#)

## IA

La inteligencia artificial es un área de la informática que se centra en la creación de sistemas capaces de realizar tareas que requieren inteligencia humana, como el reconocimiento de voz, la toma de decisiones y el aprendizaje automático. [1](#) [6](#)

## IR-LED

El LED infrarrojo (*Infrared Light Emitting Diode*) es una fuente de luz que emite luz en el espectro infrarrojo. Se utiliza en sensores PPG para medir el flujo sanguíneo en tejidos más profundos, como los músculos, debido a su capacidad de penetrar la piel. [9](#)

## LSTM

*Long Short-Term Memory* (Memoria a Corto y Largo Plazo). Es un tipo de red neuronal recurrente (RNN) que es particularmente eficaz para aprender dependencias a largo plazo en secuencias de datos. [13](#) [19](#) [47](#) [48](#) [51](#) [52](#) [67](#) [68](#)

## ML

*Machine Learning* o Aprendizaje Automático, es una rama de la inteligencia artificial que permite a los sistemas aprender y mejorar automáticamente a partir de la experiencia sin ser explícitamente programados. [8](#) [14](#) [79](#)

## MSLT

La prueba de latencia múltiple de sueño (*Multiple Sleep Latency Test*) es un examen que mide el tiempo que tarda una persona en quedarse dormida durante varias siestas a lo largo del día. Se utiliza para evaluar la somnolencia diurna y diagnosticar trastornos del sueño como la narcolepsia. [10](#)

## MWT

La prueba de mantenimiento de la vigilia (*Maintenance of Wakefulness Test*) evalúa la capacidad de una persona para permanecer despierta en condiciones relajadas. Se utiliza en estudios de somnolencia para determinar si alguien es capaz de mantenerse alerta en situaciones donde es importante evitar quedarse dormido. [10](#) [15](#) [17](#) [21](#) [25](#) [71](#) [73](#)

## N1

La fase N1 es la primera etapa del sueño no REM (NREM). Es una etapa de transición entre la vigilia y el sueño, donde el cuerpo comienza a relajarse y la actividad cerebral disminuye. Durante esta fase, los músculos se relajan y las ondas cerebrales se ralentizan, con la aparición de ondas theta. [11](#) [14](#) [26](#)

## N2

La fase N2 es la segunda etapa del sueño no REM (NREM). En esta fase, el cuerpo entra en un estado de sueño más profundo. La frecuencia cardíaca y la temperatura corporal disminuyen. Las ondas cerebrales muestran patrones característicos llamados “husos del sueño” y “complejos K”, que indican la estabilización del sueño. [11](#)

## N3

La fase N3, también conocida como sueño de ondas lentas o sueño profundo, es la tercera etapa del sueño no REM (NREM). Es la fase más profunda y restauradora del sueño, donde el cuerpo repara tejidos y fortalece el sistema inmunológico. Durante esta fase, predominan las ondas delta en el cerebro. [11](#)

## NREM

La fase de No movimiento ocular rápido (*Non-Rapid Eye Movement*) es la etapa del sueño en la que el cerebro reduce su actividad, permitiendo el descanso y la recuperación física. Se divide en tres subfases: N1 (sueño ligero), N2 (sueño medio), y N3 (sueño profundo o de ondas lentas). [10](#)

## PCA

*Principal Component Analysis*. Un método de reducción de dimensionalidad que transforma un conjunto de observaciones de variables posiblemente correlacionadas en un conjunto de valores de variables no correlacionadas llamadas componentes principales, que explican la mayor parte de la varianza de los datos. [67](#), [68](#)

## PERCLOS

*PERcentage of eye CLOSure*, una medida ampliamente utilizada para evaluar la somnolencia basada en el porcentaje de tiempo que los ojos de una persona permanecen cerrados más del 80% durante un período determinado. Es un indicador no invasivo y confiable del nivel de fatiga, especialmente útil en aplicaciones como la vigilancia de conductores, ya que se ha demostrado que está altamente correlacionado con la disminución del estado de alerta y el aumento de la somnolencia. [6](#), [7](#), [13](#)

## PMData

*PMData* es un conjunto de datos que contiene información sobre la actividad física y patrones de movimiento humano. Los datos fueron recolectados utilizando dispositivos *Fitbit* para evaluar la actividad física y la calidad del sueño. Este conjunto es útil para estudios de salud, rendimiento físico y evaluación de la fatiga en diversas actividades. [17](#), [23](#)

## PNS

El sistema nervioso parasimpático es responsable de las funciones de “reposo y digestión” del cuerpo. Actúa como un contrapeso para el sistema nervioso simpático y se activa en estados de calma y relajación. Este sistema ayuda a conservar la energía, disminuyendo la frecuencia cardíaca, aumentando la actividad digestiva, y promoviendo procesos de recuperación y regeneración en el cuerpo. [11](#)

## PPG

La *Fotoplethismografía* es una técnica óptica no invasiva utilizada para medir cambios en el volumen de sangre en los vasos sanguíneos de la piel, lo que permite calcular la frecuencia cardíaca. Los sensores PPG suelen estar presentes en dispositivos *wearables*, como los smartwatches, y utilizan luces LED y fotodetectores. [9](#), [12](#), [13](#), [31](#), [65](#)

## PSG

La *Polisomnografía* es un estudio exhaustivo del sueño que utiliza múltiples mediciones fisiológicas, incluyendo el EEG (actividad cerebral), EOG (movimientos oculares), EMG (actividad muscular) y ECG (ritmo cardíaco), para evaluar las fases del sueño y detectar trastornos del sueño como la apnea. [11](#)

## REM

La fase de movimiento ocular rápido (*Rapid Eye Movement*) es una etapa del ciclo de sueño caracterizada por movimientos rápidos de los ojos, sueños vívidos, y actividad cerebral elevada. Esta fase es esencial para la recuperación mental y cognitiva. [10](#)

## RNN

*Recurrent Neural Network*. Una clase de redes neuronales que es capaz de procesar datos secuenciales y de utilizar dependencias temporales entre las entradas, lo que permite recordar información a lo largo de secuencias de tiempo. [19](#)

## SHHS

El *Sleep Heart Health Study* es un estudio diseñado para investigar cómo los trastornos del sueño, como la apnea, afectan la salud cardiovascular y otras condiciones relacionadas. Los datos se recopilan como parte de la iniciativa *National Sleep Research Resource* (NSRR) e incluyen mediciones polisomnográficas y de salud de miles de participantes para estudiar la relación entre los patrones de sueño y las enfermedades cardiovasculares. [17](#) [23](#) [26](#)

## Simula

Simula es una organización de investigación que lleva a cabo estudios de alta calidad en el ámbito de las tecnologías de la información y la comunicación. [18](#)

## SNS

El sistema nervioso simpático es una parte del sistema nervioso autónomo que se activa principalmente en situaciones de estrés o emergencia, desencadenando la respuesta de “lucha o huida”. Esto incluye el aumento de la frecuencia cardíaca, la dilatación de los bronquios, la inhibición de la digestión, entre otros efectos. [11](#) [12](#)

## SVM

Las máquinas de vectores de soporte (*Support Vector Machines* en inglés) son un tipo de algoritmo de *machine learning* supervisado que se utiliza principalmente para la clasificación y regresión de datos. Funcionan encontrando un hiperplano que mejor separe las clases en un espacio multidimensional. [8](#) [14](#)

## Vanishing gradient

Fenómeno que ocurre durante el entrenamiento de redes neuronales profundas, en el cual los gradientes se vuelven extremadamente pequeños en las capas iniciales de la red, impidiendo que los pesos se actualicen de manera efectiva. Esto causa que las capas cercanas a la entrada no aprendan, afectando la capacidad de la red para captar patrones a largo plazo. Es común en redes recurrentes (RNN) y redes con muchas capas, y puede mitigarse con técnicas como LSTM, funciones de activación ReLU y normalización por lotes (Batch Normalization). [19](#)

## VLF

La banda de frecuencia muy baja (*Very Low Frequency* en inglés) se refiere a un rango de frecuencia de la variabilidad de la frecuencia cardíaca (HRV), que suele estar entre 0.0033 y 0.04 Hz. Las variaciones en esta banda pueden estar relacionadas con procesos fisiológicos como la regulación de la presión arterial y otras funciones autónomas. [8](#)