



Instituto Tecnológico de Buenos Aires

Bioingeniería

Detección automática de fibrilación auricular en registros de electrocardiógrafo portátil

Autores:

Pedro Lucas Barrera (58746)

Lorenza Guadalupe Vecino Schandy (58652)

Tutora:

Dra. Bioing. María Paula Bonomini

Agosto 2023

Agradecimientos

Queremos agradecer a nuestra tutora María Paula Bonomini por ofrecernos la oportunidad de realizar este proyecto, que nos dio nuestro primer puntapié al mundo laboral, por su orientación y gran acompañamiento durante todo este trabajo, mostrándose siempre disponible y bien dispuesta para cualquier cosa que necesitáramos. Además, aunque las reuniones fueran muy temprano o hubiera que resolver un incendio para Virtual Sense, siempre se mostraba con una sonrisa y buena voluntad, transmitiéndonosla a nosotros (aún con lo duras que son las mañanas para mí, Pedro). A Sergio, Diego y todo el equipo de Virtual Sense por confiar en nosotros durante todo el trayecto y por las oportunidades que nos abrió este proyecto. También queremos agradecer a Juliana Gambini, nuestra profesora de la materia de Machine Learning, por ofrecer su tiempo y mostrar una gran predisposición para asesorarnos sobre todas las dudas acerca de los métodos empleados de inteligencia artificial que tanto nos atormentaron (y eran muchas).

Yo, Pedro, quiero por mi parte agradecer a toda la gente que me acompañó durante este largo año. A mis papás y a mi hermana María, que siempre me apoyaron durante toda la carrera y me tuvieron mucha fé. A mi hermano Nico, que me salvó las papas incontables veces enseñándome y ayudándome a resolver ciertos problemas relacionados a programación. A mis amigos, tanto del ITBA como de afuera, por estar siempre y porque, aunque me tomaron tanto el pelo porque parecía que no entregaba nunca, pusieron a disposición sus computadoras para aumentar mi capacidad de cómputo y no tardar cien años en correr todos los modelos. Finalmente a mi novia Mora, que como todos (pero un poquito más) aguantó mis infinitas quejas y me apoyó en todo, aún cuando estaba frustrado o triste, logrando siempre hacerme sonreír y motivarme para seguir.

Por mi lado, yo Lorenza, quiero agradecer a todos mis amigos y familia. En particular, a mis padres y mis hermanos que me acompañaron durante el proceso y vivieron en primera fila mi recurrente mal humor. También a mis amigas, por aguantar mis desahogos, y estar pendientes y preguntar incansables veces como viene la tesis y cuando la vamos a entregar. Además, a mi mejor amiga Paty, que está lejos pero siempre presente. Creo que es una etapa de la vida curiosa, emocionante pero llena de cambios y momentos de incertidumbre. Así que gracias a todos por

acompañar y apoyarme en mi camino.

Por último, queremos agradecer a nuestros perros Bruno, Tita y Bandido por ser siempre un incondicional apoyo moral peludo, aunque a veces se pongan a ladrar a las 2 de la mañana o rompan cosas, los queremos igual.

Resumen

La fibrilación auricular (FA) es el tipo de arritmia cardíaca más común [1]. Dado que generalmente es asintomática, a menudo pasa desapercibida hasta que surgen complicaciones graves, como un accidente cerebrovascular [2]. Por lo tanto, es crucial el desarrollo de herramientas de diagnóstico rápidas, económicas y accesibles para detectar la FA en una etapa temprana. En este sentido, los dispositivos de telemonitorización asistidos por aprendizaje automático muestran gran potencial.

Este trabajo presenta un algoritmo que detecta automáticamente la FA en señales obtenidas por electrocardiógrafos portátiles conectados a una plataforma de telemonitorización a través de smartphones. El algoritmo consta de tres etapas: detección de ruido, detección de latidos ectópicos y detección de FA. La etapa de detección de ruido implica analizar las señales de ECG utilizando ventanas de 5 segundos con un desplazamiento de 1 segundo. Un clasificador KNN predice la presencia o ausencia de ruido en cada ventana, lo que permite detectar segmentos ruidosos. Los segmentos no ruidosos se procesan utilizando un algoritmo de Pan-Tompkins para encontrar los tiempos de ocurrencia de los complejos QRS de la señal y la serie de intervalos RR correspondiente. Luego, se eliminan los latidos ectópicos usando un clasificador XGBoost, generando la serie NN. Esta última se utiliza para calcular los índices de entrada para un clasificador de predice la presencia o ausencia de FA en la señal de ECG.

El algoritmo se entrenó y validó utilizando la base de datos de FA de una derivación del desafío de Physionet de 2017, obteniendo dos modelos: Uno entrenado con datos balanceados que utiliza un clasificador de regresión logística y uno con desbalanceados que utiliza un clasificador AdaBoost, cada uno con distintas ventajas dependiendo el ambiente de uso. El primero logró una exactitud, precisión, sensibilidad, especificidad y F1-score del 92.35%, 52.73%, 90.62%, 92.51% y 66.67% respectivamente mientras que el segundo logró una exactitud, precisión, sensibilidad, especificidad y F1-score del 95.58%, 76.07%, 69.53%, 98.65% y 72.65% respectivamente. Además, se realizó una validación adicional utilizando otras dos bases de datos, alcanzando resultados similares, lo que demostró una buena adaptabilidad del algoritmo a diversas bases de datos.

Lista de contracciones

- **FA:** fibrilación auricular
- **SSLAFDB:** “Short Single Lead AF DataBase”
- **ACV:** accidente cerebro-vascular
- **ECG:** electrocardiograma
- **s:** segundos
- **ISISTAN:** Instituto Superior de Ingeniería de Software de Tandil
- **EMG:** electromiograma
- **PVC:** “premature ventricular contraction” - contracción ventricular prematura
- **PAC:** “premature auricular contraction” - contracción auricular prematura
- **KNN:** “K-nearest neighbors” - K-vecinos más cercanos
- **SVM:** “support-vector machines” - Máquinas de vectores de soporte
- **RL:** Regresión logística
- **SR:** “sinus rhythm” - ritmo sinusal
- **RMS:** “root mean square” - raíz cuadrática media
- **VA:** “various arrhythmias” - arritmias varias
- **BIH:** “Boston Israel Hospital” - Hospital Israelí de Boston
- **ms:** milisegundos
- **SQI:** “signal quality index” - índice de calidad de la señal
- **VP:** verdadero positivo
- **VN:** verdadero negativo

- **FP:** falso positivo
- **FN:** falso negativo
- **ICBA:** Instituto Cardiológico de Buenos Aires
- **PPV:** “positive predictive value” - valor predictivo positivo
- **PCA:** “principal component analysis” - análisis de componentes principales
- **BPM:** “beats per minute” - latidos por minuto
- **HR:** “heart rate” - ritmo cardíaco
- **iHR:** “instantaneous heart rate” - ritmo cardíaco instantáneo
- **HRV:** “heart rate variability” - variabilidad del ritmo cardíaco
- **SDNN:** “standard deviation of NN intervals” - desviación estándar de los intervalos NN
- **MiB:** mebibyte
- **MB:** megabyte
- **AW4:** Apple Watch 4
- **AHA:** “American Heart Association” - Asociación Estadounidense del Corazón

Índice

| | | |
|----------|--|-----------|
| 1 | Introducción | 10 |
| 1.1 | Motivación | 10 |
| 1.2 | Objetivos | 11 |
| 1.3 | Virtual Sense | 11 |
| 2 | Marco Teórico | 13 |
| 2.1 | Fibrilación auricular | 13 |
| 2.2 | ECGs | 14 |
| 2.2.1 | Tipos de ruido del ECG | 18 |
| 2.2.2 | Latidos ectópicos | 20 |
| 2.3 | Aprendizaje automático | 22 |
| 3 | Materiales | 23 |
| 3.1 | Bases de datos | 23 |
| 3.1.1 | SSLAFDB | 24 |
| 3.1.2 | PTB-XL Kaggle | 25 |
| 3.1.3 | Base de datos sintética | 25 |
| 3.1.4 | MIT-BIH Arrhythmia Database | 25 |
| 4 | Métodos | 27 |
| 4.1 | Preprocesamiento | 28 |
| 4.1.1 | Disminución de ruido | 28 |
| 4.1.2 | Estandarización de las señales | 31 |
| 4.2 | Detección de ruido | 33 |
| 4.2.1 | Selección de ventanas | 33 |
| 4.2.2 | Índices de ruido | 35 |
| 4.2.3 | Clasificación de ruido | 37 |
| 4.3 | Adquisición de la serie RR | 45 |
| 4.3.1 | Detección de complejos QRS | 45 |
| 4.3.2 | Procesamiento de las series RR | 50 |

| | | |
|----------|--|------------|
| 4.3.3 | Armado y viabilidad de la serie RR | 52 |
| 4.4 | Detección de latidos ectópicos | 53 |
| 4.4.1 | Clasificación de latidos | 53 |
| 4.4.2 | Armado y viabilidad de la serie NN | 58 |
| 4.5 | Detección de FA | 59 |
| 4.5.1 | Índices de detección de FA | 59 |
| 4.5.2 | Clasificación de FA | 68 |
| 4.5.3 | Tiempo de cómputo y uso de memoria | 69 |
| 4.6 | Funcionalidades adicionales | 70 |
| 5 | Resultados | 73 |
| 5.1 | Detección de ruido | 73 |
| 5.1.1 | Índices de ruido | 73 |
| 5.1.2 | Optimización de índices, hiperparámetros y clasificador | 76 |
| 5.1.3 | Clasificación de ruido de una señal completa | 79 |
| 5.2 | Detección de complejos QRS | 81 |
| 5.3 | Detección de latidos ectópicos | 83 |
| 5.3.1 | Índices de ectópicos | 83 |
| 5.3.2 | Optimización de índices, hiperparámetros y clasificador de latidos | 86 |
| 5.3.3 | Clasificación de latidos de señales ejemplo | 90 |
| 5.4 | Detección de FA | 91 |
| 5.4.1 | Índices de FA | 91 |
| 5.4.2 | Optimización de índices, hiperparámetros y clasificador de FA | 94 |
| 5.4.3 | Tiempo de cómputo y uso de memoria | 100 |
| 5.4.4 | Evaluación de la utilidad de la detección de ruido y ectópicos | 100 |
| 5.4.5 | Clasificación de FA de señales ejemplo | 101 |
| 6 | Discusión | 103 |
| 7 | Conclusiones | 108 |
| | Bibliografía | 115 |

| | | |
|----------|---|------------|
| A | Apéndice A | 116 |
| A.1 | Rendimiento de los 2 clasificadores de FA, con y sin detección de ruido y ectópicos | 116 |
| B | Apéndice B | 117 |
| B.1 | Lista de librerías y sus respectivas versiones | 117 |

Introducción

1.1 Motivación

Las enfermedades cardiovasculares constituyen la principal causa de muerte a nivel mundial, con un estimado de 17.9 millones de vidas perdidas al año [3]. Dentro de este tipo de enfermedades se encuentran las arritmias cardíacas. Si bien existen diversos tipos de arritmias, la fibrilación auricular (FA) es la de mayor prevalencia [1]. En América Latina, la FA se encuentra entre las cuatro enfermedades cardiovasculares más comunes, con cerca de 50,000 casos nuevos al año. En Argentina, en el año 2017, afectó a aproximadamente 312,000 personas, lo que representa el 0.71% de la población [4]. Además, la prevalencia global de la FA se ha triplicado en las últimas cinco décadas, y se anticipa que esta tendencia continuará en ascenso. En 2018, se estimó que el riesgo de padecer FA a lo largo de la vida era de 1 en 3 para individuos blancos y de 1 en 5 para individuos negros [5].

Un desafío significativo con la FA radica en su naturaleza típicamente asintomática, lo cual a menudo resulta en diagnósticos tardíos, cuando ya han surgido complicaciones mayores, como un accidente cerebrovascular (ACV) [2]. Por tanto, es crucial desarrollar herramientas de diagnóstico rápidas, económicas y ampliamente accesibles que permitan detectar la FA en etapas tempranas. En este contexto, el telemonitoreo a través de dispositivos asistidos por Aprendizaje Automático (ML, por sus siglas en inglés) emerge como una estrategia prometedora para alcanzar dicho objetivo.

La utilización de dispositivos portátiles de electrocardiograma (ECG) vinculados a plataformas de telemonitoreo posibilita un seguimiento remoto y constante del ritmo cardíaco de los pacientes. La aplicación de técnicas de ML para el análisis automático de estos registros ofrece la ventaja de un pre-diagnóstico sin requerir la intervención de un operador humano, lo que facilita la evaluación de un mayor número de registros en menos tiempo. Este enfoque permite una detección temprana de episodios de FA que podrán ser confirmadas por un médico

y una monitorización precisa de la frecuencia cardíaca y la respuesta al tratamiento, lo que puede resultar en una atención más oportuna y personalizada.

1.2 Objetivos

El objetivo de este trabajo fue desarrollar una solución de software para la detección de fibrilación auricular y ritmo cardíaco a partir de registros de ECG monoderivación provenientes de dispositivos portátiles. Este algoritmo fue diseñado específicamente para ser implementado por la empresa Virtual Sense en su plataforma de telemonitoreo.

Los objetivos específicos que contribuyen al logro de este objetivo general son los siguientes:

- Implementar una detección robusta de ruido en la señal que permita discriminar entre segmentos de alta y baja calidad, con el objetivo de minimizar la pérdida de información.
- Desarrollar un método confiable para la detección de fibrilación auricular a partir de señales cortas (30 segundos (s)), de una sola derivación (derivación I) y baja calidad.
- Estructurar el algoritmo en clases para modularizarlo, facilitando su integración en la plataforma de telemonitoreo de Virtual Sense.
- Crear un modelo predictivo que pueda ser ejecutado en un smartphone de forma veloz, permitiendo la detección en tiempo real.

1.3 Virtual Sense

Virtual Sense es una empresa argentina dedicada al desarrollo de plataformas digitales para el monitoreo remoto de la salud. Su objetivo es facilitar el acceso a la telemedicina a través de la creación de soluciones tecnológicas que mejoren la calidad y las expectativas de vida de los pacientes.

La plataforma de telemonitoreo fue desarrollada y sigue siendo ampliada por un equipo multidisciplinario que incluye ingenieros de software del Instituto Superior de Ingeniería de Software de Tandil (ISISTAN). Esta plataforma permite la conexión de varios dispositivos

portátiles, tales como relojes inteligentes y ECGs para smartphone del modelo WeCardio UN (descrito en la sección 2.2), facilitando la adquisición de sus señales y/o alertas a distancia en tiempo real.

En el contexto de este proyecto, se colaboró con Virtual Sense en el diseño y desarrollo de algoritmos para la detección de patologías cardíacas como la fibrilación auricular, utilizando señales obtenidas a partir de los ECGs para smartphone. Esta iniciativa se llevó a cabo como una práctica profesional supervisada de un año de duración, bajo la guía de María Paula Bonimini. Durante este periodo, se trabajó en estrecha colaboración con la empresa para asegurar que los algoritmos desarrollados estuvieran en consonancia con los objetivos y expectativas de la misma.

Marco Teórico

2.1 Fibrilación auricular

La FA es la arritmia cardíaca más común, caracterizada por una actividad eléctrica irregular en las aurículas del corazón. En lugar de los ritmos regulares y coordinados de contracción, las aurículas experimentan contracciones rápidas e irregulares, lo que puede llevar a una disfunción cardíaca y aumentar el riesgo de complicaciones graves, como el accidente cerebrovascular.

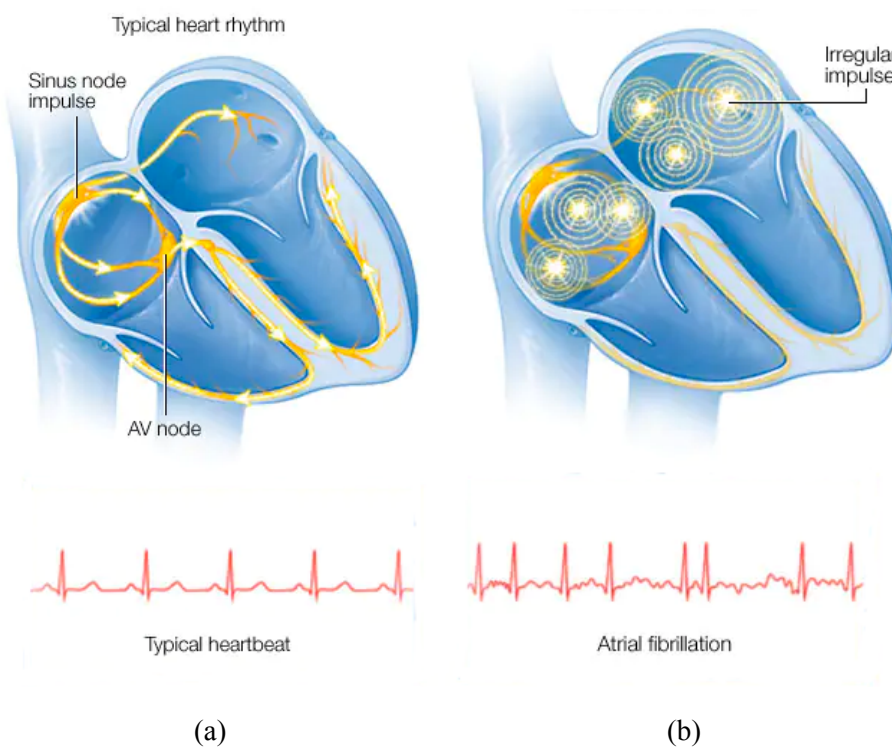


Figura 2.1. Conducción eléctrica en un corazón con: a) Ritmo cardíaco típico. b) FA. [6]

La FA puede clasificarse en diferentes subtipos, como la FA paroxística (episodios que duran menos de 7 días y se detienen de forma espontánea), la FA persistente (episodios que duran más de 7 días y requieren intervención médica para restaurar el ritmo sinusal normal) y la FA permanente (ritmo auricular irregular de forma continua y no susceptible de ser restaurado).

Los factores de riesgo para el desarrollo de FA incluyen la edad avanzada, enfermedades

cardíacas estructurales, hipertensión arterial, diabetes, enfermedad pulmonar crónica, obesidad y antecedentes familiares de FA. Además, ciertos factores desencadenantes, como el consumo excesivo de alcohol, el estrés, la cafeína y algunas drogas, pueden desencadenar episodios de FA en personas susceptibles.

El diagnóstico de la FA generalmente se realiza mediante la evaluación clínica y el registro de un ECG. El ECG muestra características específicas de la FA, como la ausencia de ondas P discernibles, ritmo irregular y presencia de ondas f. Sin embargo, debido a que la FA puede ser asintomática o presentar síntomas leves y transitorios, muchos casos de FA pasan desapercibidos y no se diagnostican hasta que ocurren complicaciones graves.

El tratamiento de la FA se enfoca en restaurar y mantener un ritmo cardíaco normal, así como en prevenir complicaciones. Las opciones de tratamiento pueden incluir medicamentos para controlar la frecuencia y el ritmo cardíaco, procedimientos de cardioversión eléctrica para restaurar el ritmo sinusal, y ablación por catéter para eliminar los focos anormales de actividad eléctrica.

En resumen, la fibrilación auricular es una arritmia cardíaca común que puede llevar a complicaciones graves. El diagnóstico temprano y el tratamiento adecuado son fundamentales para controlar la afección y prevenir complicaciones asociadas. Comprender los factores de riesgo, los síntomas y los métodos de diagnóstico de la FA es esencial para mejorar la atención y el manejo de esta arritmia cardíaca.

2.2 ECGs

Los ECGs son registros de la actividad eléctrica del corazón. Se utilizan para diagnosticar distintos trastornos cardíacos, incluyendo la FA, como se mencionó anteriormente. Como se muestra en la Figura 2.2, un ECG típico presenta las siguientes ondas características:

- **Onda P:** Representa la despolarización de las aurículas del corazón, que es la contracción de las cavidades superiores. Es una onda pequeña y redondeada.
- **Complejo QRS:** Es la suma de varias ondas y representa la despolarización de los

ventrículos del corazón, es decir, la contracción de las cavidades inferiores. Incluye la onda Q, la onda R y la onda S. En ciertas derivaciones del ECG, como la I y la II, la onda R es la más prominente y representa la despolarización ventricular principal.

- **Onda T:** Representa la repolarización de los ventrículos, que corresponde al proceso de recuperación eléctrica de las células cardíacas después de la despolarización. Es una onda redondeada y suele ser más ancha y menos prominente que el complejo QRS.

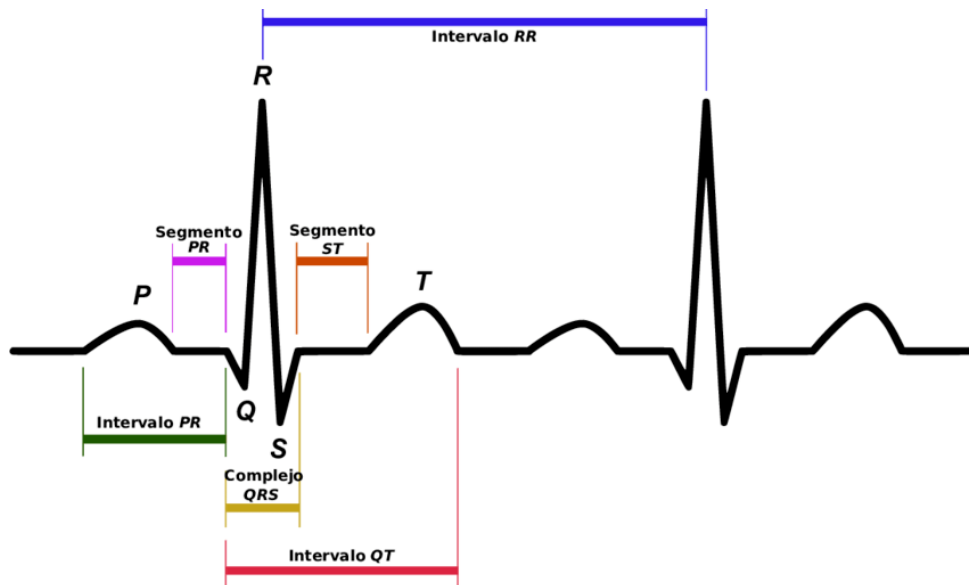


Figura 2.2. Elementos característicos de un ECG típico [7]

Además de las ondas mencionadas anteriormente existe la llamada onda F, también conocida como onda de fibrilación o flutter auricular. En la FA la actividad eléctrica auricular es caótica y rápida. Por lo tanto, en lugar de la onda P normal, en el ECG se pueden ver ondas F que suelen ser pequeñas y de forma irregular.

Otros elementos de interés en los ECGs son los intervalos RR y la serie RR. Los intervalos RR corresponden a los lapsos de tiempo que transcurren entre dos picos R consecutivos, como se puede observar en la Figura 2.2, y proporcionan una medida del ritmo cardíaco. Por su parte, la serie RR es la secuencia de intervalos RR de una señal, y es fundamental para el análisis de la variabilidad del ritmo cardíaco y para el diagnóstico de la FA.

Existen diversos tipos de electrocardiógrafos, dispositivos que registran las señales de ECG.

Los electrocardiógrafos de 12 derivaciones son ampliamente utilizados en entornos clínicos. Estos utilizan un conjunto de 10 electrodos colocados estratégicamente en el cuerpo del paciente, incluyendo las extremidades y el tórax.

Las 12 derivaciones del electrocardiograma se pueden clasificar en dos grupos: las que proporcionan información sobre el plano frontal y las que se enfocan en el plano horizontal, cada una con su forma particular de medir el potencial eléctrico del corazón. En el plano frontal, existen tanto derivaciones bipolares como monopolares. Las derivaciones I, II y III son bipolares, y miden la diferencia de potencial entre dos electrodos específicos. Por otro lado, las derivaciones monopolares en este plano son aVR, aVL, y aVF. En estas, se utilizan 3 electrodos (BI, BD y PI) y se mide la diferencia de potencial entre un solo electrodo y un punto de referencia, correspondiente al punto medio de los otros 2 electrodos. En el plano horizontal, se incluyen las derivaciones precordiales V1 a V6. Estas son monopolares, y al igual que en el caso anterior, miden la diferencia de potencial entre un solo electrodo y un punto de referencia. En este caso el punto de referencia corresponde a la central de Wilson, que representa el potencial promedio de los electrodos BI, BD y PI.

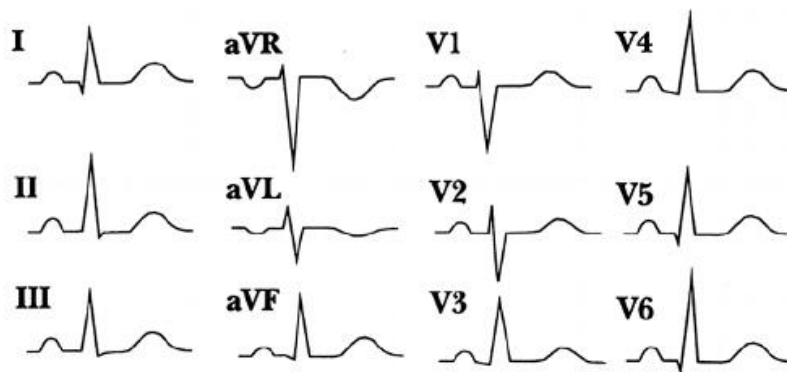


Figura 2.3. Morfología de las 12 derivaciones de ECG [8]

Cada derivación proporciona una representación única de la actividad eléctrica del corazón desde un ángulo específico. Esto implica que la morfología de las señales varía dependiendo de la derivación, como se muestra en la Figura 2.3 [8]. De este modo, los electrocardiogramas de 12 derivaciones brindan una perspectiva completa y detallada de la actividad eléctrica del corazón.

Por otro lado, los electrocardiógrafos de una derivación son dispositivos más simples que capturan la actividad eléctrica del corazón desde una única perspectiva. Son portátiles y adecuados para el monitoreo básico y seguimiento a largo plazo en entornos no clínicos. Sin embargo, su precisión diagnóstica es limitada y a menudo proporcionan señales de calidad inferior. Existen diversos tipos de estos electrocardiógrafos, incluyendo dispositivos de pulsera, para smartphones y parches adhesivos. Los electrocardiógrafos para smartphones suelen proveer la derivación I, ya que generalmente se registran mediante los dedos de las manos. Esto se debe a que la derivación I es la proyección del vector eléctrico cardíaco en la dirección determinada entre las extremidades superiores izquierda y derecha.

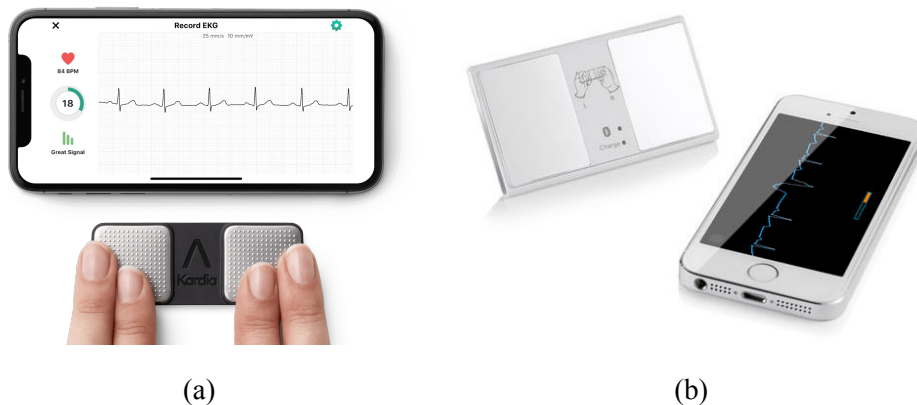


Figura 2.4. Electrocardiógrafos comerciales de 1 derivación para smartphone.
a) Kardiamobile de AliveCor [9]. b) WeCardio UN de Borsam Instruments [10].

El KardiaMobile de la empresa AliveCor es uno de los electrocardiógrafos para smartphone más comercializados. Este dispositivo permite a los usuarios registrar su ECG simplemente colocando los dedos (pulgar o índice y mayor) de cada mano en los electrodos integrados, durante 30 s. La señal generada corresponde a la derivación I, que mide la diferencia de potencial entre las extremidades superiores derecha e izquierda. Este dispositivo se conecta al smartphone a través de un adaptador o por bluetooth. Su aplicación asociada ofrece un trazado en tiempo real del ECG en la pantalla del smartphone y realiza análisis automatizados que permiten detectar la presencia de FA u otros trastornos cardíacos [9].

El WeCardio UN de Borsam Instruments, de origen chino, es otra opción de ECG para

smartphones. Este es el modelo que la empresa Virtual Sense adquirió para el telemonitoreo cardíaco de sus pacientes. Con un peso de 33 gramos y unas dimensiones de 88 x 49 mm, es práctico y fácil de usar. Funciona de manera similar al KardiaMobile, registrando señales de 30 segundos. Es una alternativa más económica, pero no proporciona alertas en caso de presencia de FA [11].

2.2.1 Tipos de ruido del ECG

Las señales de ECG tienen ruidos de distinta índole. Estos normalmente se dividen en 4 categorías: desvío de la línea de base, interferencia de electromiografía (EMG), ruido de línea y artefactos por movimientos de electrodos (Figura 2.5).

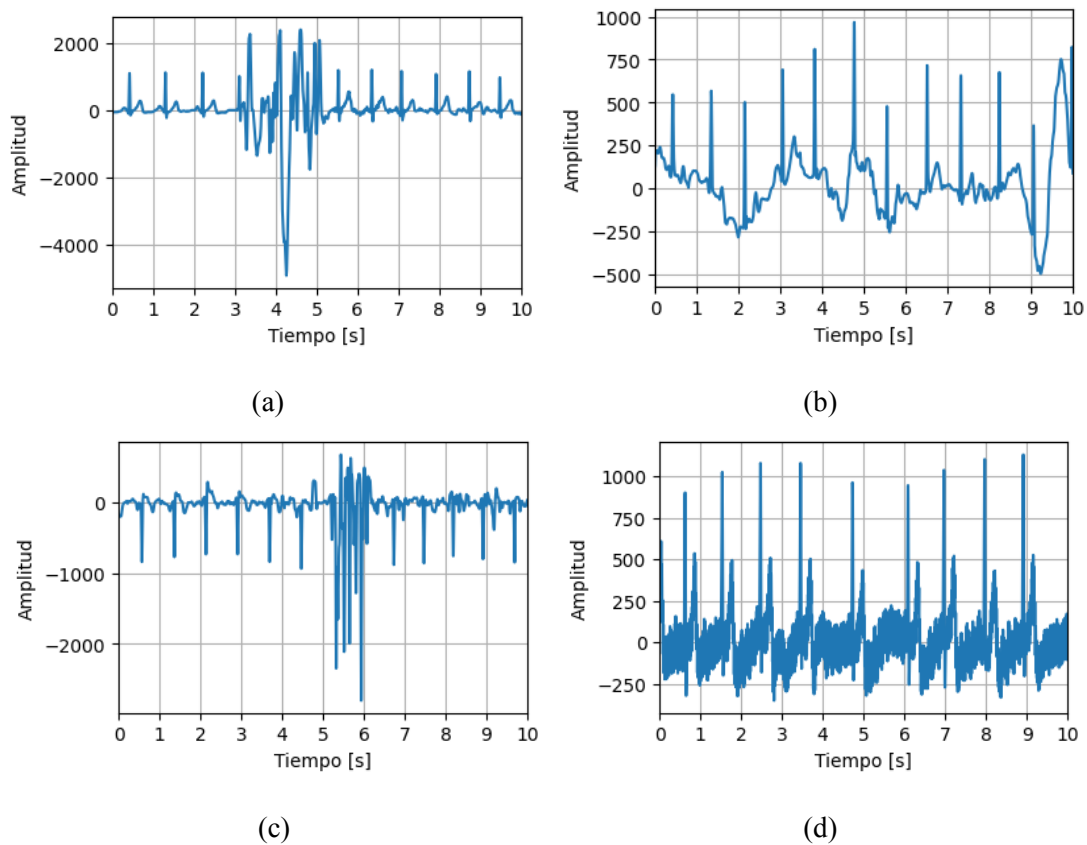


Figura 2.5. Ruidos clásicos presentes en un ECG. Estas señales fueron obtenidas de la base de datos SSLAFDB que se describe en la sección 3.1. a) Artefactos por movimiento de electrodos (señal n° 50). b) Desvío de la línea de base (señal n° 4197). c) Interferencia de EMG (señal n° 4328), d) Ruido de línea (señal n° 7107).

El desvío de la línea de base (Figura 2.5b) es un tipo de ruido de baja frecuencia que principalmente es causado por la respiración, problemas con la impedancia de los electrodos (normalmente por mal contacto electrodo-piel) o pequeños movimientos del paciente. Esto se traduce en alteraciones de la línea de base del ECG [12]. El contenido espectral de este artefacto suele encontrarse en bajas frecuencias llegando normalmente hasta 1 Hz pero en algunos casos superándolo (Figura 2.6). Este tipo de ruido suele eliminarse mediante el uso de filtros pasaaltos, filtros de mediana e interpolación cúbica.

El ruido de línea (Figura 2.5d) es un ruido de alta frecuencia que normalmente tiene componentes espectrales entre los 50 y 60 Hz. Este es causado por una incorrecta conexión a tierra del equipo e interferencia de equipos cercanos. Este se elimina fácilmente utilizando un filtro Notch, que descarta únicamente una frecuencia específica. Tanto el desvío de la línea de base como el ruido de línea son poco problemáticos dado que pueden ser normalmente removidos mediante distintos filtros lineales y no lineales [12].

Otro tipo de ruido existente en el ECG es el de la interferencia electromagnética. Este proviene de las ondas electromagnéticas emitidas por dispositivos eléctricos y electrónicos en el ambiente circundante, como luces fluorescentes, celulares, computadoras, red WiFi y cables de alimentación de otros dispositivos. Este ruido tiene un rango de frecuencias variable, dependiendo de la fuente del mismo. De todas formas, suele tener frecuencias altas, lo que hace que sea eliminable mediante un filtro pasabajos.

La interferencia por EMG (Figura 2.5c) es un ruido causado por la interferencia eléctrica de las contracciones de músculos esqueléticos (señal de EMG) en la señal de ECG. Si por ejemplo un músculo que atraviesa el vector eléctrico generado por los electrodos de una derivación se contrae, esta señal eléctrica que generó la contracción será también sensada por el equipo de ECG, mezclándose el registro de actividad eléctrica del corazón con el de los músculos. Este tipo de ruido puede ser tanto intermitente, debido a movimientos repentinos, o más estacionario. Además, su morfología depende del grado de contracción muscular. A diferencia del desvío de la línea de base y el ruido de línea, este tipo de interferencia presenta más dificultades para ser tratado. Por un lado, los componentes frecuenciales del EMG se solapan en gran parte con los componentes del complejo QRS y se extienden hasta rangos mayores (Figura 2.6).

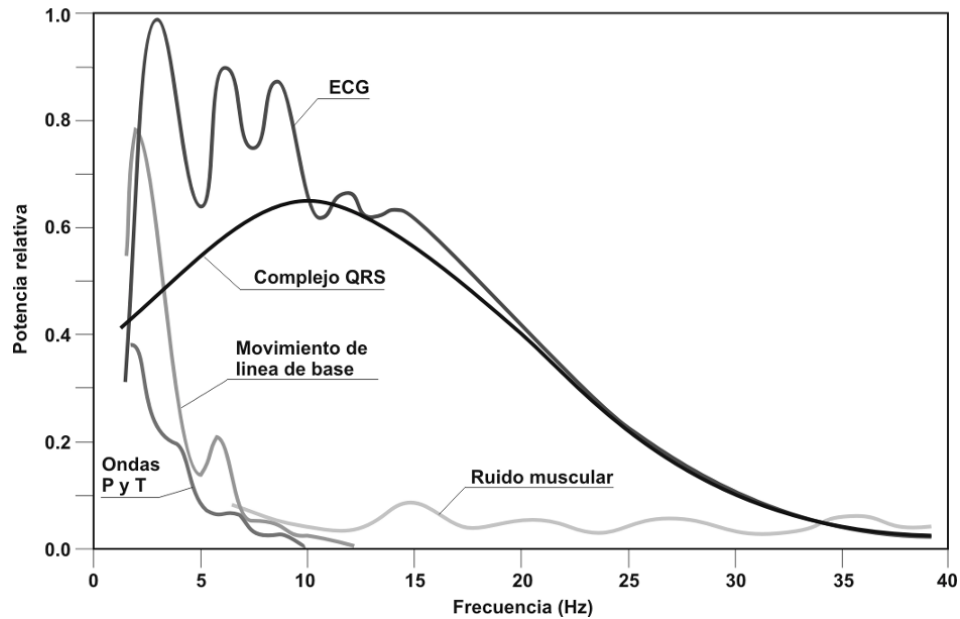


Figura 2.6. Componentes espectrales de los distintos elementos de la señal de ECG [13].

Finalmente, los artefactos por movimientos de electrodos (Figura 2.5a) son un tipo de ruido causado por cambios en la impedancia de la piel en la zona de contacto con el electrodo. Esto puede ser originado por distintas causas, entre las cuales el estiramiento de la piel en la zona de contacto y en el caso de dispositivos de una derivación que utilizan los dedos, por sudoración o suciedad en la zona de contacto. Este ruido, al igual que el de EMG, es difícil de tratar, dado que sus componentes espectrales se solapan con los de los complejos QRS, onda T y P (usualmente entre 1 y 10 Hz). En el ECG pueden verse como ondas de gran amplitud que pueden a veces confundirse con complejos QRS [12].

2.2.2 Latidos ectópicos

Al hablar de latidos ectópicos en el área de análisis del ritmo cardíaco, se suele referir a lo que se conoce como extrasístoles. Las extrasístoles son alteraciones en el ritmo, en el cual se produce un latido adicional fuera del ritmo normal. En condiciones normales, el corazón tiene un ritmo regular establecido por el nodo sinusal (llamado ritmo sinusal), que es el marcapasos natural del corazón. Sin embargo, en ocasiones, pueden presentarse impulsos eléctricos adicionales que se originan en otras partes del corazón, como los ventrículos o las aurículas.

Los dos tipos de latidos cardíacos ectópicos más comunes son las contracciones ventriculares prematuras (PVC, por sus siglas en inglés) o ectópicos ventriculares y las contracciones auriculares prematuras (PAC, por sus siglas en inglés) o ectópicos auriculares. Las PVC y las PAC se caracterizan por un latido prematuro y por ende un intervalo RR anterior más corto, seguido de una pausa compensatoria que lleva a un intervalo RR más largo de lo normal [14].

En el caso de los PVC, se dice que hay una pausa compensatoria completa, osea, una pausa tal que la suma del intervalo RR anterior y posterior al ectópico sea igual al doble de un intervalo RR normal del registro. En las PAC se dice que luego del latido prematuro hay una pausa compensatoria incompleta, osea, se genera un intervalo RR más largo de lo normal, pero la suma del intervalo anterior y el posterior es menor al doble de un intervalo RR normal [15].

Además, las PVC se generan en el ventrículo, lo que resulta en un complejo QRS con morfología anormal (Figura 2.7a). En el caso de las PAC, el impulso anormal eléctrico que lo generó se encuentra en la aurícula, por lo que la morfología del QRS en el ECG permanece normal (Figura 2.7b).

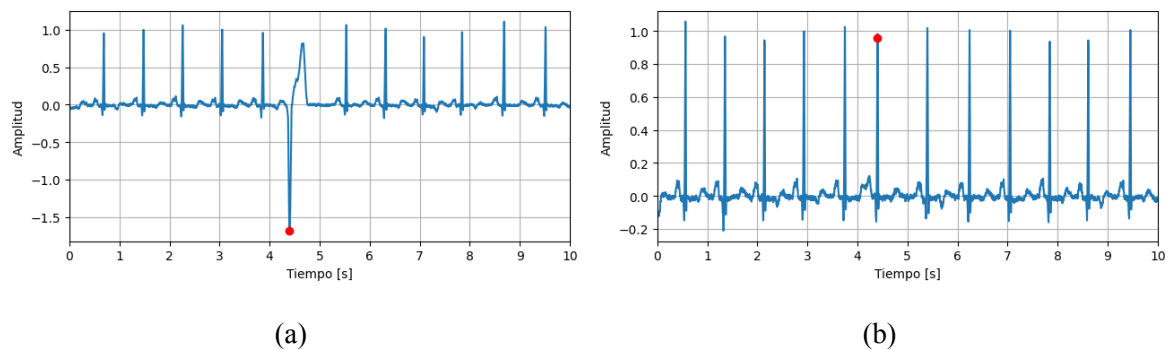


Figura 2.7. Ejemplos de latidos ectópicos anotados en la señal n° 100 de la base de datos MIT-BIH (que se describirá en la sección 3.1). a) PVC (minuto 0). b) PAC (minuto 25).

Estos latidos pueden aparecer aleatoriamente en pacientes sanos y las causas suelen ser desconocidas. Sin embargo, la aparición frecuente de latidos ectópicos en un registro de ECG puede ser un indicio importante de la ocurrencia de alguna arritmia cardíaca.

2.3 Aprendizaje automático

El aprendizaje automático es una disciplina de la inteligencia artificial que se basa en algoritmos y modelos estadísticos que permiten a los sistemas informáticos mejorar progresivamente su rendimiento en tareas específicas sin necesidad de programación explícita. Los algoritmos de ML se entrenan a partir de datos, identificando patrones y haciendo inferencias, o tomando decisiones basadas en estos patrones.

Existen tres categorías principales de aprendizaje automático: supervisado, no supervisado y por refuerzo. En el aprendizaje supervisado, el algoritmo se entrena con un conjunto de datos etiquetados, y luego aplica el conocimiento adquirido para predecir las etiquetas de nuevos datos. En el aprendizaje no supervisado, el algoritmo se entrena con un conjunto de datos no etiquetados, identificando independientemente patrones o estructuras en los datos. Por otro lado, el aprendizaje por refuerzo implica que un agente aprenda a tomar decisiones óptimas en un entorno a través de la experimentación y la retroalimentación de recompensas o penalizaciones.

El potencial del ML en la telemedicina es considerable. Puede emplearse para analizar grandes volúmenes de datos de salud, detectar patrones y hacer predicciones, lo que puede mejorar la precisión del diagnóstico, la predicción de enfermedades y el monitoreo de la salud de los pacientes. Los algoritmos de ML también pueden implementarse en dispositivos de telemedicina para proporcionar diagnósticos en tiempo real y alertas de salud a los pacientes y proveedores de atención médica.

Dentro del aprendizaje automático supervisado, algunos de los métodos de clasificación más empleados son las máquinas de vectores de soporte (SVM, por sus siglas en inglés), los K-vecinos más cercanos (KNN por sus siglas en inglés) y regresión logística (RL). Otros métodos más recientes son AdaBoost y XGBoost. Estos métodos se aplican en el presente trabajo y se detallan en la sección de métodos.

Materiales

Como se mencionó previamente, el algoritmo fue desarrollado para ser integrado a la plataforma de telemonitoreo de Virtual Sense. Dicha plataforma recoge señales de ECG portátiles, que son analizadas por el algoritmo para determinar la presencia de FA. Los dispositivos de ECG portátiles que utiliza Virtual Sense son el modelo WeCardio UN de Borsam Instruments. Estos dispositivos fueron adquiridos recientemente y aún no han sido distribuidos a los pacientes. En consecuencia, tanto el entrenamiento como la validación del algoritmo se llevaron a cabo utilizando bases de datos ya existentes.

3.1 Bases de datos

El primer paso del desarrollo fue la búsqueda de bases de datos para poder entrenar y validar el modelo. Existen diversas bases de datos de registros de ECG, pero para el desarrollo buscado se tuvieron que aplicar ciertos criterios de inclusión. El primer criterio es que la base de datos fuera de acceso y uso gratuito e irrestricto para fines comerciales. Esto es porque, como fue explicado anteriormente, este algoritmo no será utilizado únicamente con el fin de realizar un proyecto final de carrera, sino que será implementado en la plataforma comercial de telemonitoreo de Virtual Sense. El segundo criterio fue que posea registros de la derivación I, que es la de interés. La morfología de las señales de ECG difiere mucho dependiendo de la derivación tomada (Figura 2.3). De entrenar con señales que representen otro vector eléctrico, cualquier característica morfológica de la señal que quiera utilizarse ya sea para la detección de calidad de los segmentos o para la detección de FA sería problemática. El tercer criterio de inclusión fue que el ritmo de las señales esté anotado manualmente por un profesional y que haya anotaciones de señales catalogadas como FA.

Tomando en cuenta estos criterios, se buscaron posibles bases de datos. Para esto se recurrió a los repositorios de Physionet, Kaggle y a una búsqueda bibliográfica.

De las bases de datos encontradas se eligió la Short Single-Lead AF Database (SSLAFDB)

[16] para entrenar y validar. Luego, se realizó una validación por terceros pertenecientes al equipo de Virtual Sense utilizando un subconjunto de la base de datos PTB-XL [17] y una base de datos sintética. Para la detección de latidos ectópicos se utilizó la MIT-BIH Arrhythmia Database [18].

3.1.1 SSLAFDB

SSLAFDB es una base de datos de PhysioNet de señales de ECG obtenidas utilizando el KardiaMobile de AliveCor. Esta base de datos consta de 8528 señales de ECG equivalentes a la derivación I, con una frecuencia de muestreo de 300 Hz y una duración variable de entre 9 y 60 segundos (con duración mediana de 30 segundos). Los ritmos cardíacos anotados en esta base de datos se dividen en las siguientes clases:

- Ritmo normal o sinusal (N) : 5050 señales
- FA (A): 738 señales
- Otras arritmias distintas a la FA (O): 2456 señales
- Señales muy ruidosas para ser analizadas (\sim): 284 señales

Los ritmos \sim fueron utilizados únicamente para la etapa de detección de ruido del algoritmo. Por otro lado, los ritmos N y O se fusionaron en una única clase 0, y los A se renombraron clase 1.

Esta base de datos fue utilizada en la competencia “AF Classification from a short single lead ECG recording”, perteneciente al “PhysioNet/Computing in Cardiology Challenge” del año 2017. El objetivo de dicho desafío era clasificar correctamente la FA a partir de las señales de la SSLAFDB. Cabe destacar que las señales representadas en las figuras de este trabajo provienen de esta base de datos. Para cada una de ellas, se especifica el número de señal correspondiente.

3.1.2 PTB-XL Kaggle

PTB-XL-Kaggle es un subconjunto de la base de datos PTB-XL seleccionado y modificado por Kaggle [19]. Esta consta de 6528 señales de ECG de 12 derivaciones, con una frecuencia de muestreo de 500 Hz y una duración de 10 segundos. Los ritmos cardíacos anotados por Kaggle se dividen en 3 clases: ritmo normal o sinusal (SR, por sus siglas en inglés), FA y aleteo auricular (FA), y otras arritmias distintas a la FA (VA, por sus siglas en inglés). En este trabajo se utilizaron únicamente las señales de la derivación I, ya que es equivalente a la generada por los dispositivos de ECG de una derivación para smartphone. Los ritmos SR y VA se combinaron para formar la clase 0, y a los AF se los etiquetó clase 1.

3.1.3 Base de datos sintética

La base de datos sintética fue generada utilizando el simulador de Physionet “PhysioNet Model for Simulating ECG and PPG Signals with Arrhythmia Episodes” [20]. Este simulador produce señales de ECG y PPG que pueden incluir episodios de arritmia. Ajustando los parámetros del simulador se crearon 10000 registros de ECG de derivación I, de 30 intervalos RR de largo, con frecuencia de muestreo de 250 Hz. A las señales se les agregó ruido con raíz cuadrática media (RMS, por sus siglas en inglés) de 0.06 mV, proveniente de movimientos musculares, artefactos generados por el movimiento de los electrodos y desvío de la línea de base. Estos tipos de ruido explican en la sección 4.1.1. Se incluyeron ritmos de tipo normal, bradicardia, taquicardia y FA. Posteriormente, los de FA fueron asignados clase 1, y el resto se fusionaron en una única clase 0.

3.1.4 MIT-BIH Arrhythmia Database

La MIT-BIH Arrhythmia Database contiene 48 señales de media hora de registros de ECG ambulatorio de dos canales, obtenidos de 47 sujetos estudiados por el Laboratorio de Arritmias BIH (Boston’s Beth Israel Hospital), entre los años 1975 y 1979. La frecuencia de muestreo de las señales es de 360 Hz, con una resolución de 11 bits y un rango de 10 mV. Lo que la hace casi única entre las bases de datos de acceso libre es que posee anotaciones en cada latido de los registros, además de anotaciones de distintos ritmos, realizadas por dos o más cardiólogos

expertos. Entre los latidos anotados, se encuentran los PVC y los PAC, por lo que es muy útil para entrenar un algoritmo de detección automática de latidos ectópicos.

Métodos

El algoritmo diseñado comienza con un procesamiento de la señal de ECG, procede con una etapa de detección de ruido, luego sigue con la adquisición de las series RR, continua con la etapa de detección de ectópicos y adquisición de las series NN, y finaliza con la etapa de detección de FA propiamente dicha. El flujo del algoritmo se muestra en la Figura 4.1, y se explica con mayor detalle a continuación.

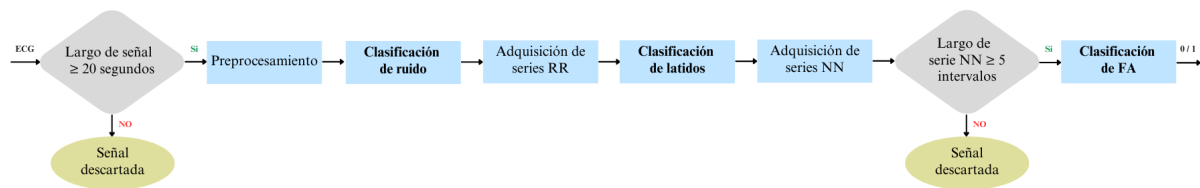


Figura 4.1. Flujo del algoritmo

En primer lugar, se evaluó la duración de la señal de ECG. Las señales que procesará este algoritmo una vez integrado en la plataforma de Virtual Sense serán de 30 segundos. De todas formas, a veces los pacientes registran señales más cortas o más largas, considerándose válidas señales con una duración mínima de 20 s. Por lo tanto, las señales de SSLAFDB con una duración menor a 20 s fueron descartadas.

Herramientas de programación

Este algoritmo se desarrolló en su totalidad en lenguaje de programación Python, versión 3.9.12. Se utilizaron diversas librerías, muchas de las cuales se mencionarán a lo largo del trabajo. En la Tabla B.1 del apéndice B.1 se muestra la lista completa de librerías implementadas y sus respectivas versiones.

En términos generales el código se estructuró en 3 clases y un archivo con funciones auxiliares nombrado `utils`:

- **NoiseClassifier:** realiza el análisis de ruido de una señal. Su método `classify`

devuelve la predicción de ruido segundo a segundo, y el método `cut_noise` devuelve los segmentos no ruidosos de la señal.

- **EctopicClassifier:** realiza el análisis de latidos ectópicos de una señal. Su método `classify` procesa una serie RR y devuelve la predicción de ectópicos latido a latido.
- **AFClassifier:** realiza el análisis de FA de una señal. Su método `classify` devuelve la predicción sobre la presencia o ausencia de FA. En esta clase se hace uso de la clase `NoiseClassifier` para eliminar el ruido, luego genera las series RR, elimina los intervalos RR asociados a latidos ectópicos utilizando la clase `EctopicClassifier`, para finalmente poder realizar la predicción de FA.

Estas 3 clases tienen un método nombrado `save_model` que permite guardar su respectivo modelo entrenado, *scaler* ajustado y atributos utilizados en archivos `.sav`. Al inicializar estas clases se cargan estos archivos para obtener los modelos listos para realizar predicciones, y evitar tener que entrenarlos cada vez que se inicia el algoritmo.

4.1 Preprocesamiento

Los objetivos de este preprocesamiento de las señales son la disminución del ruido y la estandarización de las señales.

4.1.1 Disminución de ruido

Como se mencionó en el marco teórico, los electrocardiógrafos para smartphone son dispositivos sencillos con una alta proporción de ruido en comparación con los de uso clínico. Una reducción en el ruido de la señal conduce a una detección de picos R y una serie RR confiable, que es fundamental para la correcta detección de FA. Por ende, es necesario reducir al máximo posible cualquier tipo de artefacto presente y ser muy cuidadoso en la detección del mismo.

Filtro de mediana

Los filtros de mediana resultan muy efectivos en la eliminación del desvío de la línea base del ECG y algunos ruidos residuales [21][22]. El filtro de mediana implementado consta de dos filtros de mediana móvil, implementados con la función `medfilt` de la librería `scipy.signal`, con tamaños de ventana de 200 y 600 milisegundos (ms). El filtro de 200 ms se aplica a la señal original y el de 600 ms se aplica a la salida del primer filtro. Luego, se obtiene la señal filtrada al restar la salida del segundo filtro a la señal original. En la Figura 4.2 se muestra un ejemplo de los pasos que realiza este filtro.

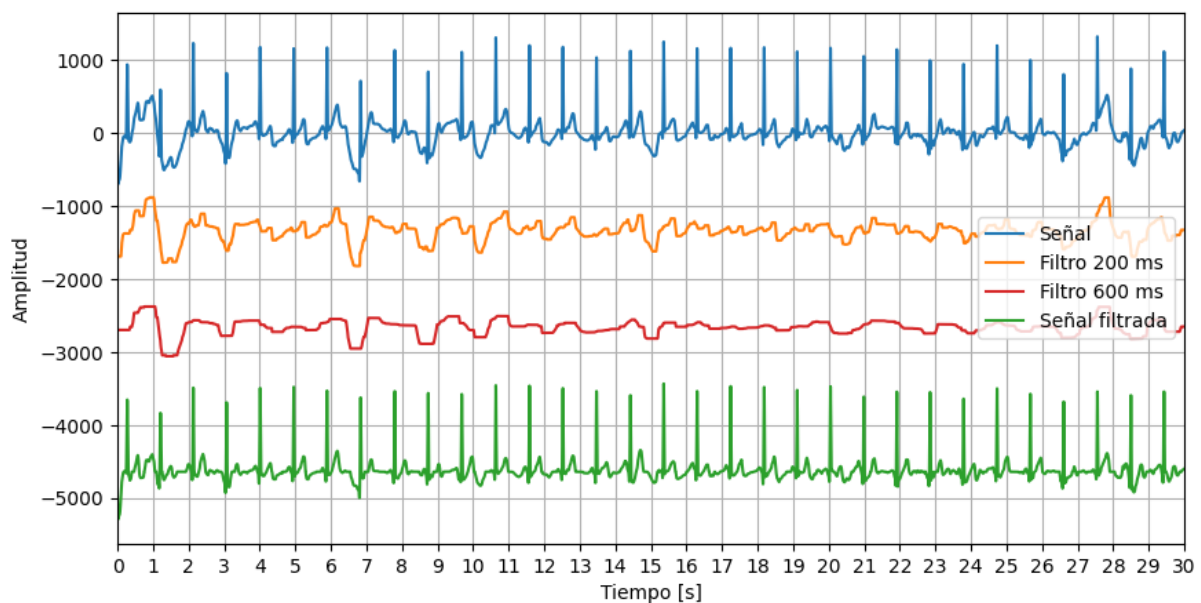


Figura 4.2. Pasos del filtro de mediana. Señal n° 26 original (azul), luego del primer filtro (naranja), luego del segundo filtro (rojo), y la señal filtrada (verde).

Filtro pasabanda

También se utilizó un filtro pasabanda para eliminar el ruido de línea, parte del ruido de EMG y para reducir la amplitud de la onda T que, en muchos casos, presenta gran amplitud. Inicialmente, se utilizaron las frecuencias de corte 5 y 15 Hz, sugeridas por el algoritmo de Pan-Tompkins. Sin embargo, dado que en numerosos casos la onda T seguía siendo muy pronunciada, se ajustaron de manera experimental estos valores hasta encontrar los óptimos. El

filtro pasabanda que se utilizó es un un filtro digital Butterworth de 5to orden, con frecuencias de corte de 8 y 25 Hz, implementado mediante la función `butter` de la librería `scipy.signal`. El filtrado se realizó dos veces, una hacia adelante y otra hacia atrás, con el objetivo de compensar la distorsión de fase producida en cada iteración, utilizando la función `filtfilt` de la librería `scipy.signal`. En la Figura 4.3 se muestra un ejemplo de señal con onda T prominente con filtro de mediana aplicado, antes y después de ser filtrada por el pasabanda. Cabe destacar que, a diferencia del filtro de mediana, este filtro se aplicó solo en ciertos casos que se discutirán más adelante.

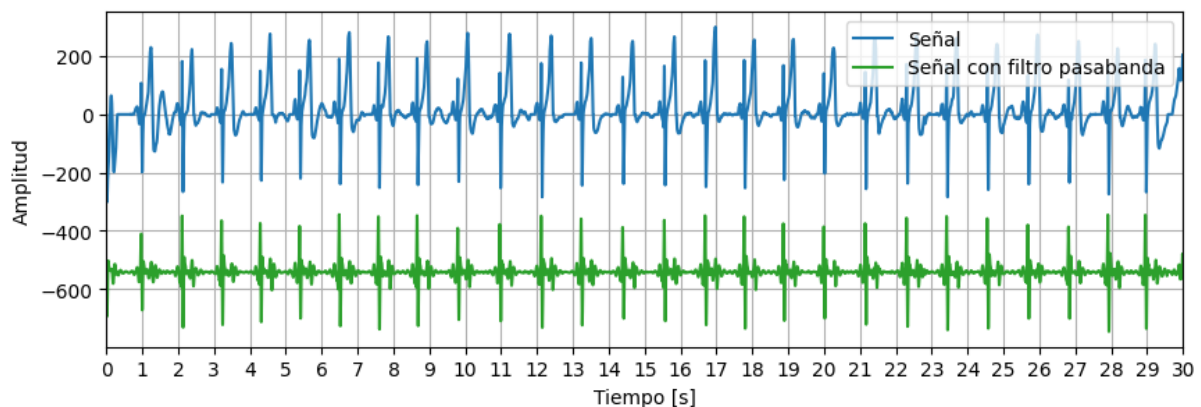


Figura 4.3. Filtro pasabanda. Señal n° 16 con filtro de mediana: Antes (azul) y después del filtro pasabanda (verde).

Filtro pasabajos

En los casos en los que no se utilizó el filtro pasabanda, se añadió un filtro pasabajos para evadir el ruido de línea y conservar solo las frecuencias de interés. Este pasabajos utilizado es un filtro digital de Butterworth de 5to orden, con una frecuencia de corte de 40 Hz. Al igual que en el pasabanda, el filtro se aplicó 2 veces en sentidos inversos para evitar la distorsión de fase. En la Figura 4.4 se muestra un ejemplo de señal con alta proporción de ruido de línea, antes y después de ser filtrada por el pasabajos.

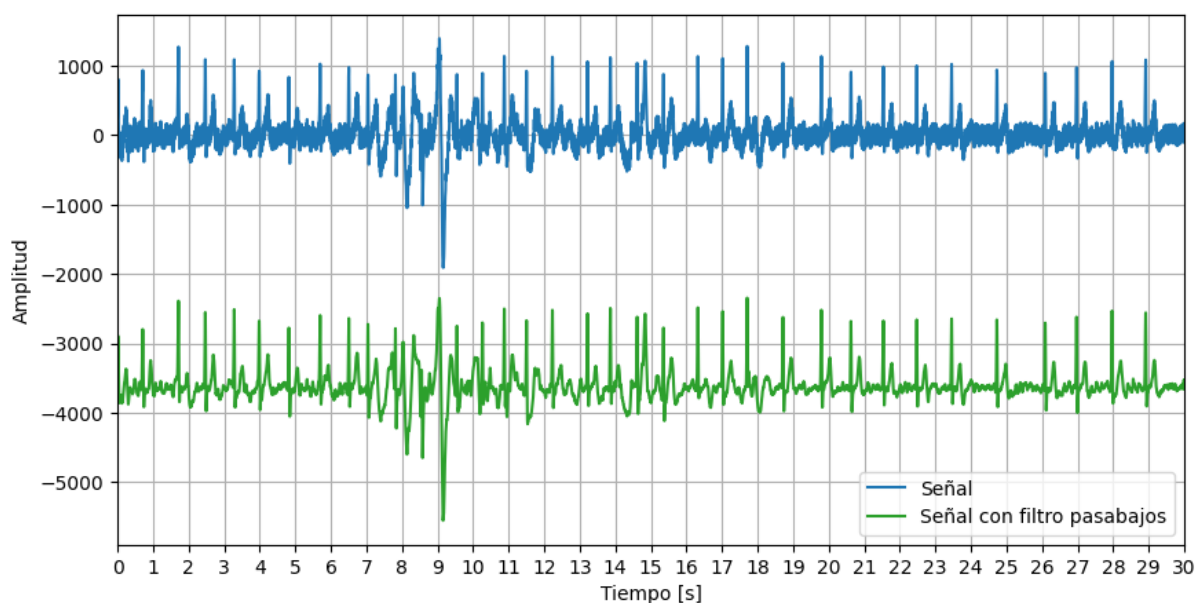


Figura 4.4. Filtro pasabajos. Señal nº 7107 con filtro de mediana: antes (azul) y después del filtro pasabajos (verde).

4.1.2 Estandarización de las señales

Corrección de la polaridad

Las señales de ECG de la derivación I suelen presentar una forma de onda en la que los picos R de los latidos normales son positivos. No obstante, si los usuarios de los ECGs para smartphones colocan los dedos al revés en los electrodos, la polaridad de la señal puede quedar invertida. Para estandarizar la polaridad de las señales, se realizó un análisis de los picos R de las mismas. Estos picos se detectaron mediante una implementación propia del algoritmo de Pan-Tompkins. Como esta detección de picos es meramente preliminar, el algoritmo de Pan-Tompkins implementado se describe en detalle más adelante en la sección de adquisición de las series RR.

Una vez obtenidos los picos R preliminares, se buscan las plantillas de cada latido. Esto se realiza extrayendo el segmento de señal que va desde 0.2 segundos antes hasta 0.4 segundos después del pico R. Obtenidas las plantillas de todos los latidos de la señal se calcula la plantilla mediana, alineando las plantillas y calculando la mediana de cada punto. Si el valor máximo de la plantilla mediana es mayor que el valor absoluto del mínimo, se considera que la polaridad de

la señal es positiva. De lo contrario (como se ve en la Figura 4.5a), se considera que la polaridad es negativa, y se corrige la señal multiplicándola por -1.

Estandarización de la amplitud

Finalmente, se normaliza la amplitud de las señales de manera que la amplitud de los picos R ronde la unidad. Esto se logró utilizando el valor absoluto máximo de la plantilla mediana. La señal de amplitud normalizada se obtiene dividiendo la señal por este valor máximo. En la Figura 4.5 se muestra un ejemplo de señal filtrada antes y después de la estandarización.

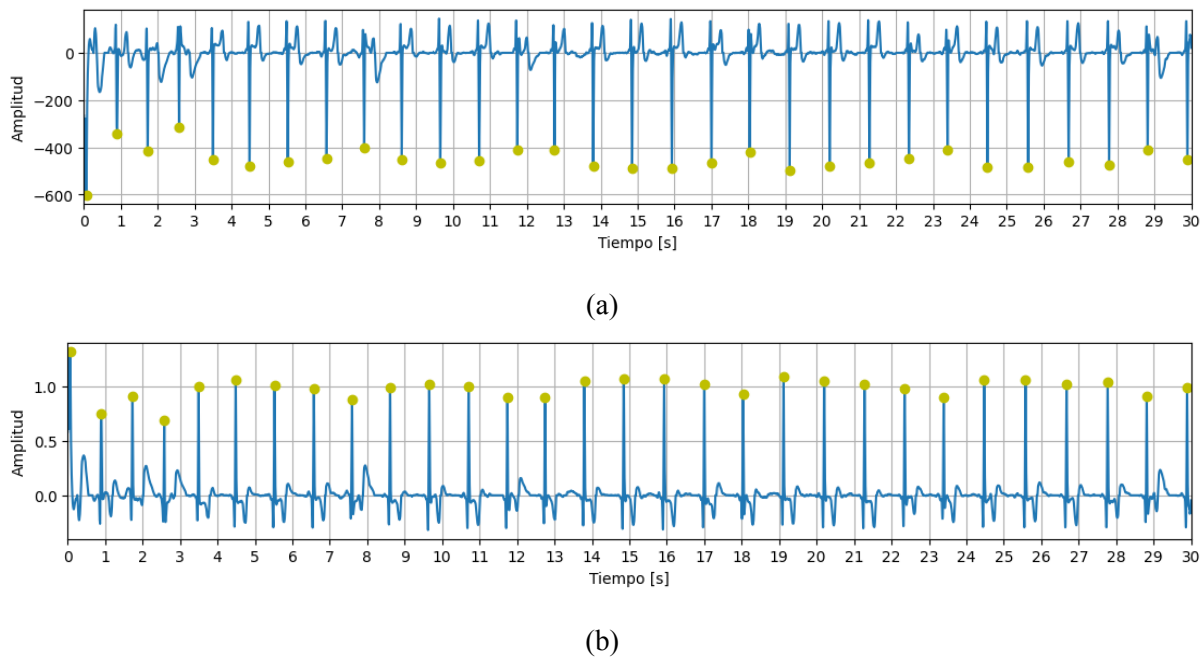


Figura 4.5. Estandarización de una señal con polaridad negativa, con los picos R preliminares (verde). Se muestra la señal nº 5: a) antes, y b) después de la corrección de la polaridad y la normalización de la amplitud.

4.2 Detección de ruido

En la etapa de preprocesamiento de la señal se buscó reducir al mínimo posible el ruido, eliminando el desvío de la línea de base, el ruido de línea y otros ruidos de baja frecuencia con el filtro de mediana y el pasabanda. Sin embargo, otros tipos de ruidos como la interferencia de EMG y los artefactos por movimientos de los electrodos, no pudieron ser filtrados. Para resolver esto se decidió agregar una etapa de detección de ruido de la señal ya filtrada.

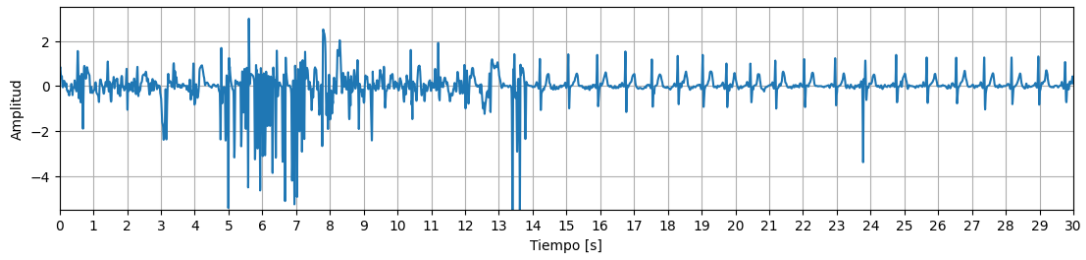
4.2.1 Selección de ventanas

Inicialmente, se intentó distinguir las señales entre ruidosas y no ruidosas utilizando la clasificación anotada en la base de datos, tomando la clase ruidosa \sim y combinando las clases N, A y O para conformar la clase no ruidosa.

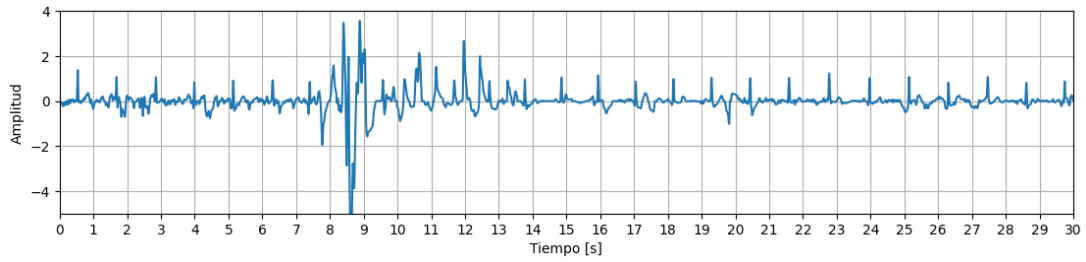
Al examinar visualmente las señales de la base de datos, se observó que existen segmentos altamente ruidosos en señales clasificadas como no ruidosas, y otros no tan ruidosos en señales anotadas como ruidosas. En la Figura 4.6a se muestra una señal clase N en la cual los segundos 0-14 son extremadamente ruidosos. Por el contrario, en la Figura 4.6b se ve una señal anotada como ruidosa cuyos segundos 0-7 y 14-24 tienen un nivel de ruido relativamente bajo.

Como resultado, se pierden por completo señales que solo tienen algunos segmentos ruidosos. De la misma forma, los segmentos ruidosos de señales clasificadas como no ruidosas no se identifican, e introducen error en el posterior análisis. Por esto, se identificó la necesidad de evaluar las señales por segmentos más pequeños o ventanas, para así poder separar partes ruidosas y no ruidosas dentro de una misma señal.

Para esto se optó por seleccionar manualmente ventanas de 5 segundos de señales ruidosas y no ruidosas. La clase ruidosa se formó escogiendo 364 ventanas notoriamente ruidosas de las señales clasificadas como tales, mientras que la clase no ruidosa se generó eligiendo 468 ventanas de apariencia no ruidosa de los ritmos N, A y O, en proporciones iguales. En la Figura 4.7 se muestran cuatro ejemplos de las 832 ventanas seleccionadas.



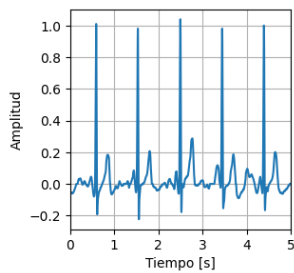
(a)



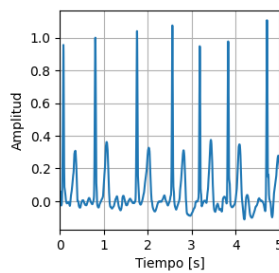
(b)

Figura 4.6. Señales con segmentos ruidosos y no ruidosos.

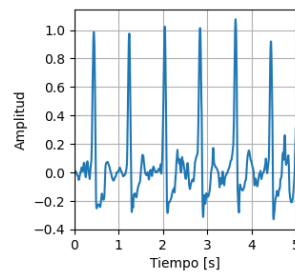
a) Señal n° 253, anotada como no ruidosa, ritmo N. b) Señal n° 1629, anotada como ruidosa.



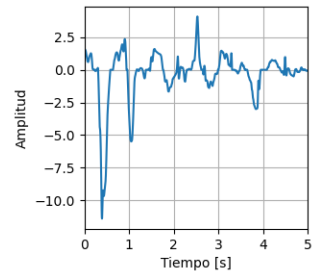
(a)



(b)



(c)



(d)

Figura 4.7. Ejemplo de ventanas preprocesadas y seleccionadas manualmente.

a) No ruidosa, ritmo N (ventana n° 5, proveniente de señal n° 9). b) No ruidosa, ritmo A (ventana n° 200, de señal n° 866). c) No ruidosa, ritmo O (ventana n° 350, de señal n° 185). d) Ruidosa (ventana n° 650, de señal n° 3564).

4.2.2 Índices de ruido

Como se explicó en el marco teórico, la señal de ECG tiene una forma característica, con ondas típicas y una periodicidad dada. En los ECGs ruidosos se distorsiona esta morfología y en los casos extremos se pierde la periodicidad. Contando con ruidos de distinta índole, la clase ruidosa se vuelve difícil de caracterizar. Por lo tanto, para lograr separar las ventanas ruidosas y no ruidosas se buscaron distintos índices que juntos logran distinguirlas.

Los índices implementados fueron índices de calidad de la señal (SQI, por sus siglas en inglés). Los SQIs más utilizados para ECG son los utilizados en Zhao & Zhang 2018 [23] y Li et al. 2007 [24]:

- **bsqi:** El porcentaje de latidos detectados simultáneamente por dos detectores de QRS con distinta sensibilidad al ruido. Estos detectores son `wqrs` y `ep_limited` (o `Hamilton`), ambos de código abierto y bien documentados. Estos se implementaron mediante las funciones `wqrs_detector` y `hamilton_detector`, respectivamente, de la clase `Detectors` de la librería `ecgdetectors` [25]. Señales de alta calidad tienen un bsqi del 100%.
- **ssqi:** El tercer momento de la distribución, conocido como *skewness* en inglés. Brinda una idea de la simetría de la señal. De acuerdo con el teorema central del límite, los procesos aleatorios no correlacionados tienden a conformar distribuciones normales, mientras que los procesos correlacionados suelen presentar distribuciones no normales. A su vez, las distribuciones normales se asocian con valores de *skewness* cercanos a cero. Por lo tanto, en señales con mayor ruido, se esperaría que una *skewness* cercana a cero, lo que indica una mayor similitud con una distribución normal teórica. Este índice se implementó mediante la función `skew` de la librería `scipy.stats`.
- **ksqi:** El cuarto momento de la distribución, conocido como *kurtosis* o *kurtosis* en inglés. Proporciona información sobre el apuntamiento o achatamiento de la distribución de probabilidad de una variable aleatoria. Al igual que la *skewness*, la *kurtosis* brinda detalles sobre el grado de normalidad de una distribución. Una señal cuyos valores tienen una distribución normal pura presenta una *kurtosis* de 3, mientras que en una señal de ECG

con ritmo sinusal limpio tiende a ser mayor a 5. Como las señales ruidosas se asocian con distribuciones más normales, se espera que tengan valores de curtosis menores a 5 [24]. Este índice fue implementado utilizando la función `kurtosis` de la librería `scipy.stats`.

- **fSQI:** La proporción de la señal que aparenta ser una línea plana. Se calcula como el porcentaje de la señal cuyo gradiente es menor a 1 % del máximo. Si una señal de ECG posee ruido, se espera que tenga pocas zonas que aparenten ser una línea plana, mientras que si es poco ruidosa debería tener mas zonas que aparentan una línea plana.
- **pSQI:** La potencia relativa del complejo QRS, utilizada para detectar interferencia del EMG. Se calcula como el cociente de la potencia presente en la banda de frecuencia que contiene la mayor parte de la energía del complejo QRS (5-15 Hz) y la totalidad del ECG (5-40 Hz), evitando las bajas frecuencias:

$$pSQI = \frac{\int_{5Hz}^{15Hz} P(f)df}{\int_{5Hz}^{40Hz} P(f)df} \quad (4.1)$$

- **basSQI:** Es la potencia relativa del desvío de la línea de base. Se calcula como el cociente de la potencia presente en la banda de frecuencia que contiene la línea de base (0-1 Hz) y la totalidad del ECG (0-40Hz):

$$basSQI = \frac{1 - \int_{0Hz}^{1Hz} P(f)df}{\int_{0Hz}^{40Hz} P(f)df} \quad (4.2)$$

Dado que las señales con las que se opera son de baja calidad y corta duración (entre 20 y 60 segundos), es crucial no solo detectar el ruido, sino también evitar la sobredetección que podría resultar en la pérdida de demasiados segmentos de la señal para su análisis. Cabe destacar que no todos los tipos de ruido son dañinos para el algoritmo en cuestión, ya que el objetivo principal no es eliminar todo el ruido, sino asegurar que se obtengan segmentos suficientemente limpios para una detección efectiva de los complejos QRS. Por ello, los índices se calcularon a partir de las señales preprocesadas con el filtro mediana, corrección de polaridad, normalización de la amplitud y el filtro pasabajos explicados en la sección 4.1. Además, el bSQI, sSQI y kSQI fueron calculados tanto de esta manera como reemplazando el filtro pasabajos por el pasabanda con frecuencias de corte de 8 y 25 Hz detallado en la sección de 4.1.1. Esto se debe a que este

pasabanda es el primer paso del algoritmo de Pan-Tompkins de implementación propia que será utilizado y explicado mas adelante para detección de latidos.

En el caso del bSQI, se consideró relevante ver si la detección de ambos algoritmos era certera una vez aplicado el pasabanda. De no aplicarlo antes del calculo de los índices, podría generarse que el detector mas sensible al ruido se vea afectado por ruidos de frecuencias fuera de estas bandas, por lo que no serían ruidos que afecten nuestra detección de latidos. Lo mismo para el sSQI y kSQI. Es posible que ruidos en otras bandas de frecuencia que son filtradas por el pasabanda puedan afectar la curtosis y *skewness*. Sin aplicar antes el pasabanda es posible que se detecte ruido que no interfiere con la robustez del detector. Para el caso del pSQI, sin embargo, no se aplicó este filtro pasabanda porque, como se explicó previamente, este índice tiene el objetivo de analizar diferencias de energías entre las bandas principales del complejo QRS (5-15 Hz) y de la señal de ECG entera (5-40Hz). De aplicar el pasabanda, se perdería la información de entre 25 a 40 Hz. Aunque el ruido presente en estas bandas no cause problemas *per se*, puede agregar contraste para detectar el ruido que esta superpuesto en la banda de interés (5-15Hz).

Algo similar aplica para el basSQI. Si se aplicaba el pasabanda, se perdería la información de la banda de frecuencia que se busca comparar (de 0 a 1 Hz), por lo que el índice carecería de sentido. Se pensó en excluir este índice a probar, dado que el desvío de la línea de base era eliminada casi en su totalidad por el filtro mediana y, lo restante, sería removido por el filtro pasabanda. Sin embargo, se decidió analizarlo de todos modos, debido a que es ampliamente utilizado en la bibliografía.

4.2.3 Clasificación de ruido

Resumiendo, la clasificación de ruido en este trabajo se trata de un problema de clasificación binaria de ventanas de 5 segundos de la señal de ECG. Los atributos son extraídos de las ventanas, y las 2 clases son la ruidosa (1) y la no ruidosa (0). Estos atributos se obtuvieron con el uso del filtro pasabajos y, cuando correspondía, el pasabanda y fueron estandarizados restándole a cada valor la media del atributo, y luego dividiéndolo por la desviación estándar del mismo. La estandarización se implementó utilizando el `StandardScaler` de la librería

`sklearn.preprocessing`.

Correlación de Kendall-Tau

El primer paso de la selección de la combinación de índices a utilizar como entrada del modelo fue ver la correlación que estos tienen entre sí. Para esto se usó la correlación de Kendall-Tau, una medida de la similitud entre dos conjuntos de clasificaciones o rangos. Este es un índice de correlación no paramétrico para datos tanto ordinales como continuos que, a diferencia del coeficiente de Pearson, mide correlación monotónica y no únicamente lineal, y no requiere que los datos estén normalmente distribuidos. Esta medida de correlación se implementó mediante el método `corr(method="kendall")` de la librería `pandas`.

Dado un conjunto de observaciones de variables X e Y , el coeficiente Tau de Kendall mide las relaciones en función de la concordancia direccional de los pares observados. Específicamente, se considera si los pares de observaciones (X_i, Y_i) y (X_j, Y_j) son concordantes, es decir, tienen el mismo orden, o discordantes, con un orden opuesto. Si X_i es igual a X_j , entonces el par se excluye del cálculo. El coeficiente se calcula como el número de pares concordantes menos el número de pares discordantes, todo dividido por el número total de pares.

La correlación de Kendall-Tau varía de -1 a 1, donde -1 indica una correlación negativa perfecta, 1 una correlación positiva perfecta y 0 ninguna correlación. Los valores negativos indican que cuando una variable aumenta, la otra disminuye, y los valores positivos indican que las variables tienden a aumentar juntas [26]. Se consideró que una correlación con valor absoluto de 0 a 0.3 era correlación débil, de 0.3 a 0.6 correlación moderada y de 0.6 a 1 correlación fuerte [27].

Wrapper

Luego, se pasó a la segunda etapa de selección de índices para encontrar la combinación óptima. Para esto se implementó el método conocido como *wrapper*. Este método implica el usar para

la selección de variables el mismo algoritmo de clasificación que se planea usar en el problema. Se experimenta con todas las combinaciones posibles de las variables y se elige la mejor en base a distintas métricas de evaluación. Este método suele ser bastante eficaz ya que proporciona una combinación óptima de características que se adapta bien al modelo elegido. Sin embargo, tiene un alto tiempo de cómputo y puede conducir al sobreajuste del modelo. Como la cantidad de índices totales en este caso es baja, el tiempo de cómputo no debería ser tan prolongado, por lo que se consideró un método apropiado.

Este método experimental se usó tanto para la selección de índices como para la elección del método de clasificación. En el caso de los índices, se probaron todas las combinaciones posibles, exceptuando las que incluían el mismo índice calculado con y sin el pasabanda (por ejemplo, en una misma combinación nunca podrían estar presentes bSQI y bSQI con pasabanda). También se evitaron las combinaciones de índices que tuvieran un coeficiente de correlación alto (mayor a 0.6). De esta manera se probaron 313 combinaciones distintas, mediante una función de implementación propia.

Tipos de clasificadores

Para elegir el tipo de clasificador, se evaluaron varios algoritmos de aprendizaje supervisado ampliamente utilizados. Estos fueron:

- **KNN:** Es un algoritmo no paramétrico y de aprendizaje perezoso, lo que significa que no hace ninguna suposición sobre la distribución subyacente de los datos y que no tiene una fase de entrenamiento explícita. KNN funciona encontrando los K vecinos más cercanos en el espacio de las características para cada nueva observación y asignándole la clase que predomina entre estos vecinos. Un hiperparámetro importante es el número de vecinos a considerar. Otra elección importante es si se ponderan o no las distancias entre los vecinos y la observación, y en ese caso la métrica de distancia (como la distancia euclidiana o Manhattan, entre otras). En el caso de no ponderar las distancias, la probabilidad de una clase se estima simplemente como la proporción de vecinos de esa clase entre los K vecinos más cercanos. En cambio, si se ponderan las distancias, la probabilidad se

calcula como la suma de los pesos de los vecinos de la misma clase, donde los pesos son inversamente proporcionales a la distancia. Este clasificador se implementó mediante la clase `KNeighborsClassifier` de la librería `sklearn.neighbors`.

- **SVM:** Este algoritmo busca un hiperplano de separación óptimo que maximiza el margen entre las clases. En el caso no lineal, utiliza funciones kernel para mapear los datos a un espacio de características de mayor dimensión donde las clases son linealmente separables. Los hiperparámetros más relevantes son el C , que controla el compromiso entre la maximización del margen y minimización del error de clasificación, y el tipo de función kernel (que puede ser lineal, polinómica, radial y sigmoidea, entre otras). Las SVMs originales no producen estimaciones de probabilidad directamente y deben ser modificadas (por ejemplo, usando la técnica de Platt) para obtenerlas. Para implementar este método se utilizó la clase `SVC` de la librería `sklearn.svm`.
- **Regresión logística:** Este método modela la probabilidad logarítmica de una clase (por defecto la clase 1 en la clasificación binaria) como una combinación lineal de las características. Los coeficientes de este modelo lineal se aprenden a partir de los datos utilizando la máxima verosimilitud. Un hiperparámetro importante es la regularización ($L1$, $L2$ o ambas), que puede añadirse para evitar el sobreajuste. La función de decisión es la función logística, que proporciona una probabilidad directa de la clase. Este clasificador se implementó a través de la clase `LogisticRegression` de la librería `sklearn.linear_model`.
- **XGBoost:** Este algoritmo construye una serie de árboles de decisión en un enfoque aditivo, donde cada árbol subsiguiente se construye para corregir los errores residuales del modelo anterior, minimizando así una función de pérdida. Los hiperparámetros más relevantes incluyen la tasa de aprendizaje (`learning rate`), la profundidad máxima de los árboles (`max_depth`), el número de árboles (`n_estimators`) y la regularización. En cuanto a la estimación de las probabilidades, XGBoost calcula puntuaciones para cada clase, sumando las puntuaciones de todas las hojas de los árboles que caen en esa clase. Luego, las puntuaciones se transforman en probabilidades mediante la función softmax. Para implementar este clasificador se usó la clase `XGBClassifier` de la librería `xgboost` [28].

Grid-search

Para cada combinación de índices, se llevó a cabo una búsqueda de la combinación óptima de hiperparámetros para los clasificadores previamente mencionados. Para esto se utilizó una técnica conocida como *grid-search*, que permite automatizar el proceso de ajuste de hiperparámetros de un modelo de aprendizaje automático. Este método consta de realizar una búsqueda exhaustiva probando todas las combinaciones posibles de valores de hiperparámetros y calcular las métricas de evaluación. De esta forma se identifican los hiperparámetros que proporcionan el mejor rendimiento del modelo. En este sentido se asemeja al método *wrapper*.

Los hiperparametros probados para cada clasificador pueden verse en la Tabla 4.1.

| Clasificadores | Parámetros del <i>grid-search</i> |
|----------------|---|
| SVM | C: [0.1, 1, 10, 100, 1000] gamma: [1, 0.1, 0.01, 0.001, (1 / (# atributos) * varianza(atributos))] kernel : ['rbf', 'linear', 'poly', 'sigmoid'] |
| KNN | k: [0 - 30] weights: ['uniform', 'distance'] p: [1, 2] |
| RL | penalty: [L1, L2, 'elasticnet'] l1_ratio: [0.01, 0.1, 0.5] C : [0.001, 0.01, 0.1, 1.0, 10, 100] solver: ['newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga'] |
| XGBoost | learning_rate: [1, 0.1, 0.01, 0.001] max_depth: [3, 5, 7, 9] n_estimators: [100, 200, 300, 400] |

Tabla 4.1. Tabla con todos los hiperparámetros que probará el método *grid-search* para los 4 clasificadores seleccionados.

Validación cruzada

Para obtener resultados más confiables del rendimiento del modelo, el método de *grid-search* se aplicó usando validación cruzada. La validación cruzada es una técnica que permite evaluar la capacidad de un modelo para generalizar su aprendizaje a nuevos conjuntos de datos. En este método, primero se mezclan los datos de manera aleatoria para eliminar la posible influencia de algún orden específico. Luego, los datos se dividen en k subconjuntos de igual tamaño. Se realiza una serie de corridas, en cada una de las cuales se utiliza un subconjunto diferente para la validación y los restantes para el entrenamiento. Las métricas obtenidas en cada corrida se promedian para obtener un conjunto de métricas finales, que son robustas frente al azar y por ende más confiables.

Métricas de evaluación

El rendimiento de los distintos clasificadores con cada combinación de índices e hiperparámetros se evaluó utilizando las siguientes 4 métricas:

$$Exactitud = \frac{VP + VN}{VP + FP + VN + FN} \quad (4.3)$$

$$Sensibilidad = \frac{VP}{VP + FN} \quad (4.4)$$

$$Especificidad = \frac{VN}{VN + FP} \quad (4.5)$$

$$F1 - score = \frac{VP}{VP + \frac{1}{2}(FP + FN)} \quad (4.6)$$

Donde VP (verdaderos positivos) es el número de casos positivos correctamente clasificados como positivos por el modelo, VN (verdaderos negativos) representa la cantidad de casos negativos correctamente clasificados como tales, FP (falsos positivos) se refiere al número de casos negativos incorrectamente clasificados como positivos y FN (falsos negativos) es la cantidad de casos positivos incorrectamente clasificados como negativos.

Obtenidos los resultados del *grid-search*, se tomó 1 candidato, con el conjunto de hiperparámetros que mostró la mejor exactitud, para cada tipo de clasificador. De esta forma

se generaron 4 clasificadores candidato. Finalmente, se eligió el clasificador que resultó tener mejores métricas, priorizando la exactitud y el F1-score.

Cabe destacar que todas estas pruebas se realizaron utilizando únicamente el 80% de los datos. El 20% restante se separó para ser utilizado como conjunto de validación y poder reportar las métricas finales. Esta separación realizó con la función `train_test_split` de la librería `sklearn.model_selection`, que incluye un mezclado aleatorio de los datos con el parámetro `random_state` que se fijó arbitrariamente en 212.

Se recuerda que el propósito del clasificador de ruido es facilitar una óptima detección de FA. Por lo tanto, el desempeño final del detector de ruido se verá reflejado en el rendimiento del clasificador de FA.

Clasificación de ruido de una señal completa

A la hora de determinar el ruido de una nueva señal, se evalúa la calidad de cada segundo de la señal. Esto se logra analizando la señal preprocesada utilizando una ventana móvil de 5 segundos con un desplazamiento de 1 segundo. La probabilidad de que una ventana sea ruidosa se obtiene a través del método `predict_proba` del clasificador entrenado, y representa la probabilidad de que la predicción del clasificador sea clase 1 (en este caso ruidosa). Cualquiera de los clasificadores probados en este trabajo, tanto de la librería `sklearn` como `xgboost`, devuelven esta probabilidad mediante `predict_proba`.

Luego, se calcula la probabilidad P de que cada segundo de la señal sea ruidoso utilizando un promedio ponderado de las probabilidades de las cinco ventanas que lo abarcan. La fórmula que se desarrolló para este promedio ponderado es la siguiente:

$$P(i) = \frac{p(i-2) + 2p(i-1) + 3p(i) + 2p(i+1) + p(i+2)}{9} \quad (4.7)$$

Donde $p(i)$ es la probabilidad de que la ventana centrada en el segundo i sea ruidosa. Como los primeros y últimos 4 segundos de la señal no poseen 5 ventanas que los abarquen, $i \in [4, n-4]$ donde n es la duración de la señal en segundos. La probabilidad P de los segundos faltantes, $[0, 1, 2, 3, n-3, n-2, n-1, n]$, se calculó mediante las siguientes adaptaciones de la fórmula

anterior:

$$P(0) = p(2) \quad (4.8)$$

$$P(1) = \frac{2p(2) + p(3)}{3} \quad (4.9)$$

$$P(2) = \frac{3p(2) + 2p(3) + p(4)}{6} \quad (4.10)$$

$$P(3) = \frac{2p(2) + 3p(3) + 2p(4) + p(5)}{8} \quad (4.11)$$

$$P(n-3) = \frac{p(n-5) + 2p(n-4) + 3p(n-3) + 2p(n-2)}{8} \quad (4.12)$$

$$P(n-2) = \frac{p(n-4) + 2p(n-3) + 3p(n-2)}{6} \quad (4.13)$$

$$P(n-1) = \frac{p(n-3) + 2p(n-2)}{3} \quad (4.14)$$

$$P(n) = p(n-2) \quad (4.15)$$

Una vez calculada la probabilidad de ruido P para cada segundo de la señal, este valor se redondea al entero más cercano, dando como resultado 0 (sin ruido) o 1 (ruido). A partir de esta clasificación la señal se separa en segmentos ruidosos y no ruidosos de la señal. Los segmentos no ruidosos se generan tomando partes de la señal donde la clasificación de ruido vale 0 de forma constante.

La fórmula de promedio ponderado y sus variantes no solo permiten encontrar valores de probabilidad de ruido más precisos sino que realizan un suavizado de los mismos. Este suavizado es de importancia ya que no es conveniente tener segmentos ruidosos y no ruidosos cortos e intercalados en una misma señal ya que se pierden intervalos RR y la serie se vuelve menos fiel.

4.3 Adquisición de la serie RR

La serie de intervalos RR de una señal se obtuvo a partir de sus segmentos no ruidosos. Para lograrlo, se comenzó realizando una detección de complejos QRS a cada segmento no ruidoso de forma individual.

4.3.1 Detección de complejos QRS

La detección de complejos QRS se llevó a cabo aplicando un algoritmo de Pan-Tompkins de implementación propia.

Algoritmo de Pan-Tompkins

El algoritmo de Pan-Tompkins, desarrollado por Jiapu Pan y Willis J. Tompkins en 1985, es un método ampliamente utilizado en la detección de complejos QRS en señales de ECG. Este algoritmo es conocido por su robustez y precisión, lo que lo hace adecuado para su uso en una variedad de aplicaciones de ECG, incluida la detección de arritmias como la FA.

El algoritmo de Pan-Tompkins utiliza una serie de 5 pasos para procesar la señal de ECG y detectar los complejos QRS: filtro pasabanda, derivación, cuadrado de la señal, integración y detección de QRSs.

1. **Filtro pasabanda:** Se aplica para realzar los complejos QRS de la señal, y atenuar las ondas P y T y ciertos tipos de ruido, como el ruido de alta frecuencia debido a la actividad muscular y el ruido de baja frecuencia debido a la respiración y el movimiento del paciente.

Convencionalmente, este filtro tiene frecuencias de corte de 5 y 15 Hz. Sin embargo, como se mencionó en la sección 4.1.1, algunas señales exhiben una onda T prominente que no se conseguía atenuar con dicho filtro pasabanda. Es esencial para la detección de los complejos QRS minimizar la amplitud de las ondas T prominentes mediante el filtro pasabanda, ya que si estas tienen una amplitud similar o mayor al complejo QRS, es

probable que el algoritmo detecte ambos picos. Esto introduciría errores en los intervalos adyacentes. Para contrarrestar este problema, se ajustaron las frecuencias de corte del filtro hasta que se logró un filtrado óptimo, estableciendo finalmente las frecuencias en 8 y 25 Hz, tal como se expuso anteriormente. La respuesta en frecuencia de este filtro se muestra en la Figura 4.8a.

2. **Derivación:** Se deriva la señal filtrada para enfatizar las pendientes de la onda QRS y reducir las de las ondas P y T.

Este paso se implementó mediante un filtro digital FIR derivador que se diseñó aplicado la siguiente fórmula:

$$\frac{dx[n]}{dn} = \frac{x[n+L] - x[n-L]}{2LT} \quad (4.16)$$

Donde $x[n]$ representa el valor de la señal preprocesada en la muestra n , L es el skip factor del filtro y T es el período de muestreo, es decir, $1/f_s$, siendo f_s es la frecuencia de muestreo. Se utilizó $L = 1$, y considerando que $T = 1/300$, la fórmula para el filtro derivador resultó ser la siguiente:

$$\frac{dx[n]}{dn} = 150 (x[n+1] - x[n-1]) \quad (4.17)$$

Entonces, los pesos del filtro son: $b_k = [150, 0, -150]$. Este filtro se aplicó a la señal mediante su convolución con la respuesta al impulso del filtro, utilizando la función `convolve` de la librería `numpy`. Su respuesta en frecuencia se ilustra en la Figura 4.8b.

3. **Cuadrado de la señal:** Se eleva al cuadrado la señal derivada para hacer que todos los valores sean positivos para en la siguiente etapa de integración no se reste al realizar el promedio móvil sino que se sume todo.

Este paso se implementó sencillamente aplicando la siguiente fórmula:

$$y(nT) = x(nT)^2 \quad (4.18)$$

En este punto del método de detección de complejos QRS se introdujo una etapa adicional al modelo original de Pan-Tompkins para eliminar los grandes artefactos de ruido que se

denominó **filtrado de artefactos**. En esta etapa se nivela a cero la amplitud de aquellos picos de la señal que resultan demasiado elevados. Esta modificación se implementó debido a que se observaron casos en los que el ruido generaba artefactos tan grandes que modificaban sustancialmente el valor medio de la señal, lo que obstaculizaba la correcta detección de los picos en la etapa final del algoritmo de Pan-Tompkins.

El filtrado de artefactos se efectuó mediante una detección preliminar de los picos para determinar la mediana de su amplitud. A continuación, se examinó la señal con una ventana móvil de 200 ms, avanzando en desplazamientos de 100 ms. Si la amplitud máxima de la ventana excede 2.5 veces la altura mediana de los picos de la señal, los valores de esa ventana se reducen a cero, lo que resulta en la eliminación del pico. El valor de 2.5 se encontró de forma experimental, probando valores entre 1.5 y 4, hasta dar con el que generaba la eliminación de picos más precisa.

A lo largo del proyecto, el rendimiento del detector de ruido fue mejorando, logrando identificar ruido en segmentos de señal con picos excesivos, por lo que esta etapa de filtrado de artefactos fue perdiendo relevancia. Al concluir el proyecto, se probó obtener los resultados finales con y sin el filtrado de artefactos, y se encontró que utilizándolo los resultados eran levemente mejores.

4. **Integración de la ventana móvil:** Se aplica una ventana móvil integradora, también conocida como promedio móvil, que se utiliza para eliminar elementos de la señal que tengan una duración menor a la del complejo QRS, como por ejemplo un pico aislado de alta amplitud pero muy corta duración, posiblemente generado por ruido.

Esta integración de la señal se implementó mediante un filtro integrador que se diseñó utilizando la siguiente fórmula:

$$y(nT) = \frac{1}{N} (x(nT - (N - 1)T) + x(nT - (N - 2)T) + \dots + x(nT)) \quad (4.19)$$

Donde N es la cantidad de muestras en la ventana de integración. Convencionalmente se utiliza una ventana de integración de 150 ms, por lo que en este caso $N = 0.150 \times f_s = 45$. Este filtro también se aplicó utilizando la función `convolve`. La respuesta en frecuencia de este filtro se muestra en la Figura 4.8c. Finalmente, en la Figura 4.8d se muestra la respuesta en frecuencia de los 3 filtros representados aplicados en serie.

5. **Detección de complejos QRS con umbrales relativos:** Se establecen umbrales relativos basados en la media de la señal para detectar los complejos QRS.

La detección de los complejos QRS a partir de la señal integrada se implementó utilizando el detector de picos `find_peaks(height = h, distance = d)` de la librería `scipy.signal`. Esta función identifica un pico si la amplitud del mismo supera un umbral h y si se ubica a una distancia mínima d del pico anterior. El umbral se fijó en la media de la señal, y la distancia mínima en tiempo se estableció en 240 ms.

El valor de tiempo mínimo entre 2 picos se obtuvo a partir del valor de ritmo cardíaco establecido como máximo. Según médicos cardiólogos del Instituto Cardiológico de Buenos Aires (ICBA), se pueden tomar como valor mínimo y máximo de ritmo cardíaco humano, 30 y 250 bpm, respectivamente. Pasando estos valores a intervalos de tiempo se obtiene un intervalo RR mínimo de 240 ms y un máximo de 2 s [29]. Por lo tanto, si se encuentra un pico a una distancia menor a 240 ms de otro, es altamente probable que alguno de ellos no se trate de un complejo QRS.

Analizando visualmente las señales, se observó que el algoritmo de Pan-Tompkins implementado ubicaba los complejos QRS de forma correcta, generalmente marcando exactamente el pico R. Es de importancia que el punto encontrado se ubique en el pico R ya que la idea es generar la serie de intervalos RR, delimitada por estos. Sin embargo, en algunas situaciones en las que las ondas Q o S eran muy destacadas, se encontró que la detección no marcaba el pico R, sino que se ubicaba ligeramente a los lados, marcando una de estas otras dos ondas. Para mejorar la precisión de la detección de estos picos, se decidió corregir su ubicación. Este ajuste se realizó tomando una ventana de 100 ms de la señal, centrada en el punto identificado como pico. Luego se ubica el valor máximo de la ventana, el cual pasa a ser el pico R corregido.

La validación de la detección de complejos QRS se realizó con la base de datos de MIT-BIH mencionada en la sección 3.1, ya que a diferencia de SSLAFDB, esta tiene todos los latidos anotados. Para esto se buscó la ubicación de los latidos anotados en la base de datos. Dentro de las anotaciones sobre latidos, se consideraron picos R las anotaciones con los siguientes símbolos: ['N', 'L', 'R', 'A', 'J', 'S', 'V', 'E', 'F', 'Q', '/', 'a', 'e', 'j', 'f']. Las series de picos

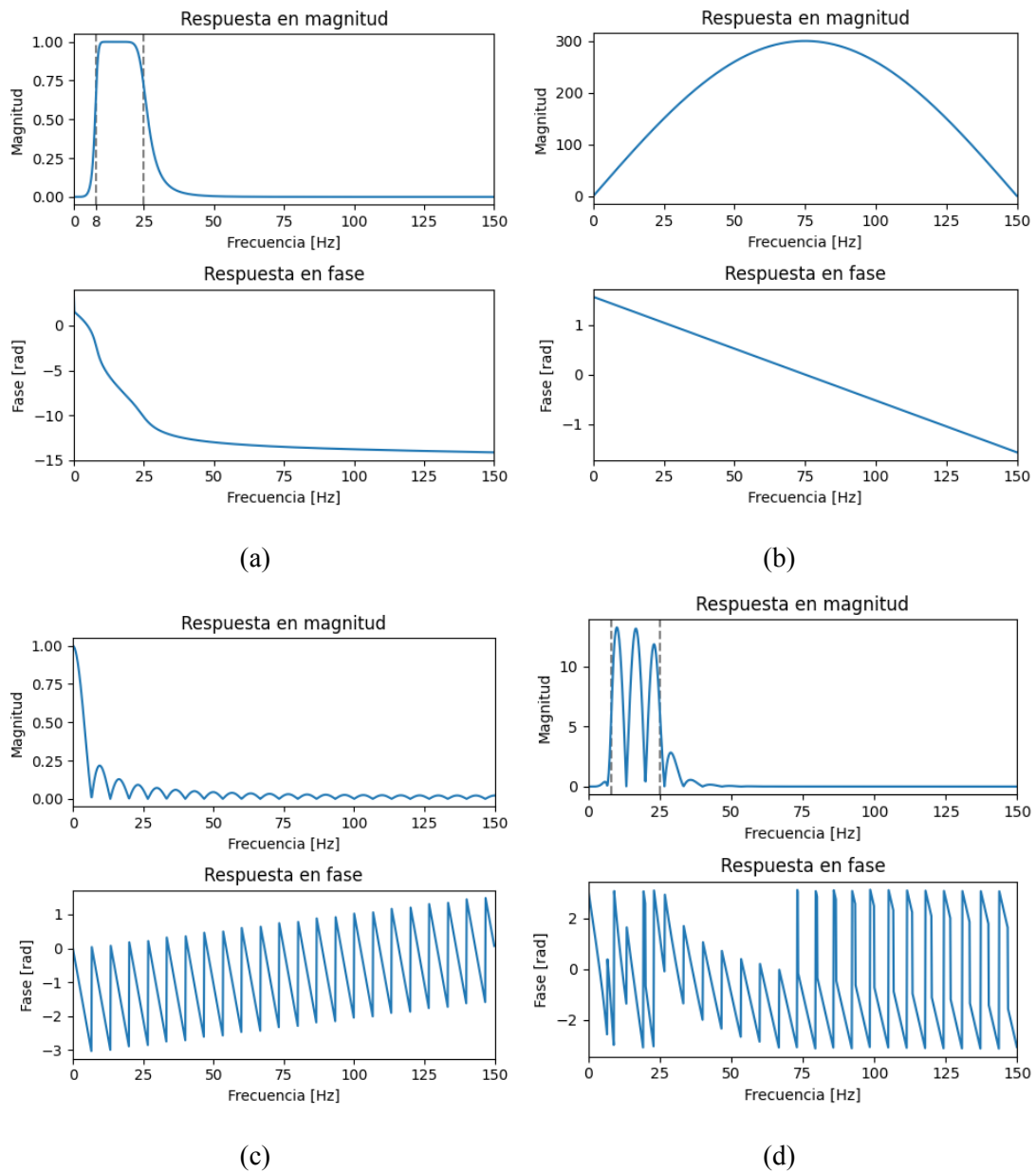


Figura 4.8. Respuesta en frecuencia de magnitud y fase de los filtros digitales aplicados en el algoritmo de Pan-Tompkins: a) filtro pasabanda, b) filtro derivador, c) filtro integrador, d) superposición de los 3 filtros anteriores.

R obtenidas por el algoritmo de Pan y Tompkins de implementación propia se compararon con las provenientes de las anotaciones. Según la bibliografía, se puede considerar que 2 latidos coinciden si difieren temporalmente en menos de 150 ms. Teniendo esto en cuenta se buscó el número de latidos VP , FP y FN totales y se calcularon las métricas de sensibilidad, valor

predictivo positivo (PPV, por sus siglas en inglés) y F1-score. La fórmula del PPV, también conocido como precisión, es la siguiente:

$$PPV = \frac{VP}{VP + FP} \quad (4.20)$$

Luego de haber ubicado los picos R de cada segmento no ruidoso de señal, la serie de intervalos RR del segmento se crea calculando el tiempo transcurrido entre cada par de picos R, y listándolos. Si bien este método de detección de picos R es en su mayoría preciso, existen casos en los que se pierden latidos o se detectan dos veces. Para abordar este problema, la serie RR se procesa mediante un método que se detalla a continuación.

4.3.2 Procesamiento de las series RR

Una vez armada la serie RR de cada segmento no ruidoso, se realiza un procesamiento de la misma para depurar errores de detección que hayan llevado a intervalos RR incorrectos. Estos errores de detección pueden ser de dos tipos: que el detector de latidos haya omitido un latido, lo que resulta en un intervalo RR anormalmente largo, o que el detector haya detectado un latido adicional, lo que llevaría a dos intervalos RR más cortos en lugar del latido real. Estos intervalos RR anormales causados por errores en la detección de latidos, pueden generar una falsa variabilidad de la frecuencia cardiaca, conduciendo a problemas en la detección de FA (como se mencionó previamente, el ritmo irregular es uno de los principales indicadores de FA).

El procesamiento consta de 2 etapas. La primera etapa consta en la identificación de intervalos RR demasiado largos o cortos para ser fisiológicamente posibles. Para esto, se tomaron los límites de 240 ms y 2 s, explicados en la sección 4.3.1.

De encontrarse un intervalo RR de duración mayor a 2 s, se supone que el detector omitió un latido. Para corregirlo se consideraron diversas opciones, como reemplazar ese intervalo por dos intervalos de la mitad de duración. Sin embargo, estos nuevos intervalos agregados no serían necesariamente iguales a los perdidos, lo que podría añadir error. Por esta razón, se optó por simplemente eliminar el intervalo largo de la serie. La Figura 4.9 muestra un ejemplo de una señal en la que, debido a la amplitud y morfología de cuatro de los picos R, estos no se detectan,

generando intervalos RR superiores a los 2 segundos. Estos intervalos se eliminan exitosamente de la serie a través de las consideraciones mencionadas previamente.

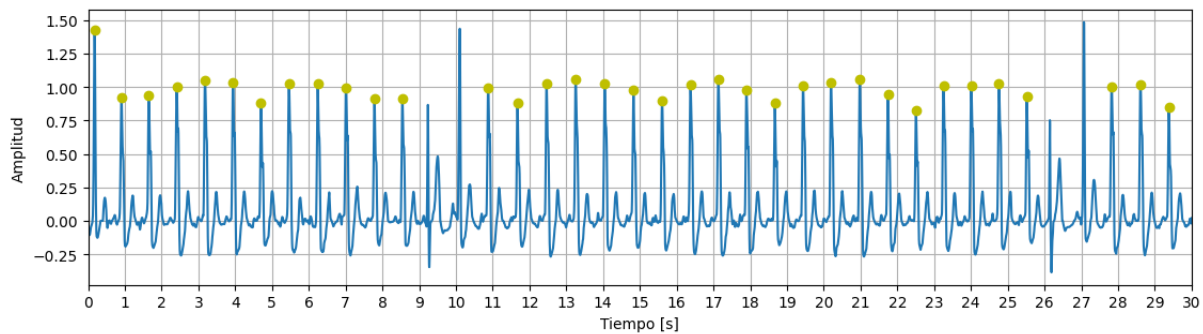
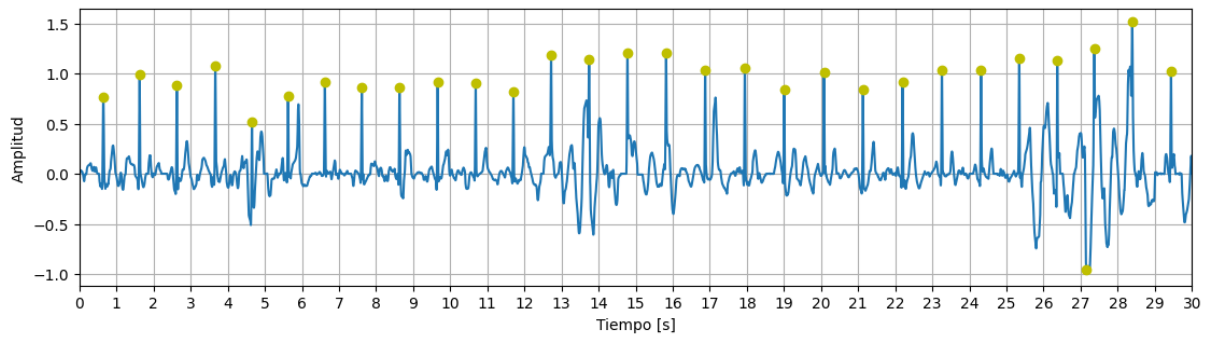


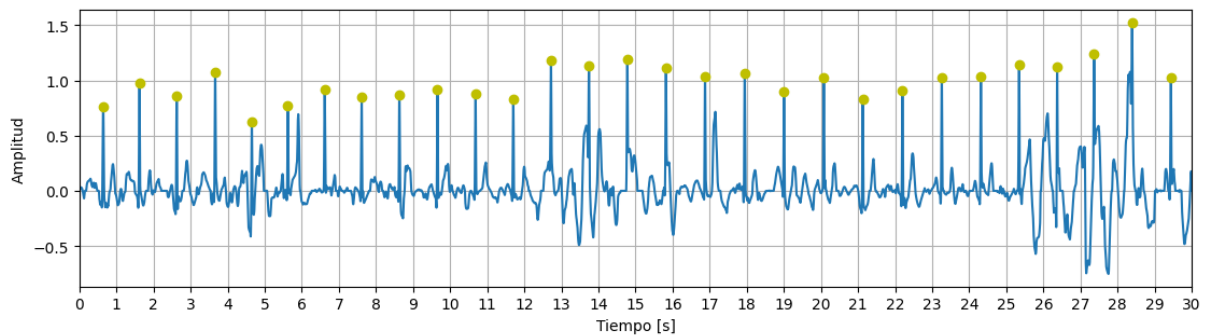
Figura 4.9. Señal n° 496 con picos R detectados (verde)

En el caso de encontrar un intervalo menor a 240 ms, se asume que el detector realizó una doble detección, lo que genera dos intervalos cortos en vez de uno regular. Eso puede darse en casos en los que se identifica como un complejo QRS a una onda T de gran amplitud, o algún artefacto de ruido con una amplitud similar a los picos, que el filtrado de artefactos no logró eliminar. Para abordar esta doble detección, los dos intervalos cortos se unen en uno solo, con una duración equivalente a la suma de ambos. Este proceso se implementa identificando un intervalo RR inferior al umbral y analizando los dos intervalos adyacentes a este, seleccionando el más corto. A continuación, se suman el intervalo menor a 240 ms y el adyacente más corto, y se reemplazan por dicha suma. De esta forma, se recupera el intervalo real, eliminando los incorrectos y restaurando la información perdida. Este método permite no solo ajustar los intervalos RR involucrados en una doble detección, sino que eliminar el pico detectado erróneamente de la lista de picos. La Figura 4.10a muestra un caso donde se detecta un pico incorrecto debido a ruido de frecuencia similar al ECG. Como se puede observar en la Figura 4.10b, el procesamiento de la serie RR identifica y elimina exitosamente este pico.

Una vez depurada la serie de esta manera, se pasa a la segunda etapa del procesamiento. En esta segunda fase, también se buscan intervalos extremadamente cortos y largos. Sin embargo, a diferencia de la primera etapa donde se usaban umbrales fijos, en esta etapa se utilizan valores adaptativos basados en las características de la serie RR analizada. Para esto se calcula la mediana de los intervalos RR de la serie ya procesada en la primera etapa. Luego, se sigue un



(a)



(b)

Figura 4.10. Señal nº 392 con picos R detectados (verde): a) antes, y b) después del procesamiento de la serie RR

proceso similar al de la primera etapa, con la diferencia de que los umbrales inferior y superior se establecen en relación al intervalo RR mediano obtenido. Se probaron distintos valores para los umbrales y se definió el inferior como un tercio del intervalo RR mediana y el superior como tres veces el mismo.

4.3.3 Armado y viabilidad de la serie RR

Para obtener la serie RR de la señal completa, se concatenan en orden las series RR ya procesadas de cada segmento no ruidoso. Si la serie RR resultante es más corta que 5 intervalos, su posterior análisis se considera no confiable y, por lo tanto, se descarta.

La serie RR final es una serie de intervalos RR en segundos, ordenados según su ocurrencia temporal. En las próximas secciones de este trabajo, la serie se denotará como $x(n)$, donde $n \in [1, N]$ y N representa el número total de intervalos en la serie.

4.4 Detección de latidos ectópicos

En la detección automática de arritmias, una etapa común es la detección de latidos ectópicos, con el objetivo de depurar las series RR. Estos latidos, que no se originan en el nodo sinusal, son considerados anomalías en el ritmo cardíaco, y suelen introducir irregularidades en un ritmo regular, dificultando la clasificación de la FA. Al depurar las series RR se obtienen las series NN, que se utilizan para extraer diferentes índices para la clasificación de la FA. La eliminación de los latidos ectópicos en este trabajo tiene como fin principal evitar que estos introduzcan ruido y afecten negativamente la detección de FA al incrementar los parámetros de irregularidad en señales con un ritmo predominantemente regular. Por tanto, en señales de FA, pasar por alto un latido ectópico no tendría un impacto significativo, ya que solo intensificaría la irregularidad del ritmo. En base a estas consideraciones, se procedió a la detección de latidos ectópicos en las series RR.

4.4.1 Clasificación de latidos

Como se explicó en la sección 2.2.2, los latidos ectópicos se caracterizan por un intervalo RR corto seguido de un intervalo RR más largo que uno normal de la serie, por lo que estas son las características más comúnmente utilizadas para su detección.

Los métodos de detección de latidos ectópicos que se describen a continuación se evaluaron utilizando la base de datos MIT-BIH, la cual incluye anotaciones de latidos. Sin embargo, según un estudio A. Khalaf y S. Mohammed, el uso de esta base de datos para la validación puede ser problemático debido a la diversidad en los tipos de anotaciones seleccionadas para representar los latidos [30]. Dentro de los símbolos considerados como latidos que se mencionaron en la sección 4.3.1, ‘V’ representa un ectópico ventricular mientras que ‘A’ un ectópico auricular. En este trabajo, ambos tipos de ectópicos se unieron para formar la clase ectópico. Todos los otros latidos de la lista fueron etiquetados como 0, que representa la clase no ectópico.

A diferencia de SSLAFDB, las señales de esta base de datos son más extensas, con duraciones de varias horas. Durante estos períodos prolongados, el ritmo cardíaco de un individuo puede experimentar significativas variaciones, alternando entre episodios de arritmia

y ritmos normales, y presentando o no latidos ectópicos de diferentes tipos. Por esta razón, las señales fueron divididas en varios segmentos de acuerdo con las anotaciones de cambio de ritmo presentes en la base de datos (identificadas con el símbolo '+'). Para cada paciente, se extrajo una serie RR correspondiente a cada ritmo en su registro, y estas series fueron analizadas de manera independiente. Esta separación es fundamental debido a que en todos los métodos se emplearán parámetros estadísticos de las series. Si se mezclaran los ritmos, los resultados, como el intervalo RR mediano o los percentiles de la serie, serían ser poco confiables.

El primer método que se intentó se basa en una simple regla de umbrales y parámetros estadísticos de la serie de intervalos RR. Se identifica un latido ectópico cuando a un intervalo RR $x(n)$ corto le sigue un intervalo $x(n+1)$ más largo debido a la pausa compensatoria, y se cumplen simultáneamente las siguientes 3 condiciones:

$$r_1 = \frac{x(n)}{x(n-1)} < \gamma_1 \quad (4.21)$$

$$r_2 = \frac{x(n+1)}{x(n)} > \gamma_{99} \quad (4.22)$$

$$r_3 = \frac{x(n+1)}{x(n+2)} > \gamma_{25} \quad (4.23)$$

Donde $\gamma_1, \gamma_{25}, \gamma_{99}$ representan los percentiles 1, 25 y 99 de la serie de razones RR, respectivamente. La serie de razones RR se calcula realizando el cociente de un intervalo RR y el siguiente, es decir:

$$r(n) = \frac{x(n)}{x(n+1)} \quad (4.24)$$

Donde $n \in [0, N-2]$, N siendo el largo de la serie.

Al estar operando con señales de corta duración, se decidió probar cambiar los percentiles por 10, 35 y 90, dado que al tener muchos menos intervalos, los percentiles de los extremos no debían poseer información estadística confiable de la serie. De todas formas, validando con la base de datos de MIT-BIH se encontró que por más de que se ajusten los percentiles, el método era poco útil.

Luego, se creó una regla de umbrales de elaboración propia, basado en las mismas características de los ectópicos en las series RR. En esta nueva regla de umbrales, se identifica

un latido ectópico si se cumplen las siguientes 3 condiciones de forma simultánea:

$$y(n-1) = \frac{x(n-1)}{x_{median}} \geq \rho_1 \quad (4.25)$$

$$y(n) = \frac{x(n)}{x_{median}} < \rho_2 \quad (4.26)$$

$$y(n+1) = \frac{x(n+1)}{x_{median}} > \rho_3 \quad (4.27)$$

Donde x_{median} es la mediana de la serie RR, y ρ_1 , ρ_2 y ρ_3 son 3 umbrales diferentes. Estos umbrales se consiguieron iterando y buscando el valor que maximizaba la sensibilidad en MIT-BIH, siempre tratando de mantener alta la especificidad. Los valores encontrados fueron $\rho_1 = 0.7$, $\rho_2 = 0.7$ y $\rho_3 = 1.1$, pero aún así los resultados demostraron que el método era poco eficaz.

Finalmente, se optó por utilizar un modelo de aprendizaje supervisado para clasificar cada latido como ectópico o no ectópico. Los índices que se utilizaron para alimentar el clasificador se basaron en las ideas planteadas en los dos algoritmos descriptos anteriormente. Entonces, se usaron r_1 , r_2 , r_3 , $y(n-1)$, $y(n)$, $y(n+1)$, γ_{10} , γ_{35} , γ_{90} , y se agregaron los siguientes índices:

$$y(n+2) = \frac{x(n+2)}{x_{median}} \quad (4.28)$$

$$y_{mean} = \frac{x(n-1) + x(n)}{2 x_{median}} \quad (4.29)$$

Con la idea de reducir las falsas detecciones de ectópicos en señales con FA, se sumó un último índice que representa de irregularidad del ritmo cardíaco P_{NZPP_2} . Esto se debe a que, en algunos casos, la variabilidad aleatoria del ritmo en FA puede producir patrones de un intervalo RR corto seguido de uno largo que se asemejen a los de un latido ectópico. Además, este índice muestra diferentes valores en presencia de latidos ectópicos (para su explicación, ver sección 4.5.1).

Al igual que para el clasificador de ruido, los índices de ectópicos fueron estandarizados restándoles la media y dividiéndolos por la desviación estándar de cada índice.

La detección de latidos ectópicos en este trabajo se aborda como un problema de clasificación binaria, donde cada latido representa una instancia y las dos clases posibles son ectópico y

no ectópico. Los clasificadores seleccionados para análisis fueron nuevamente KNN, SVM, regresión logística, y XGBoost, con y sin tener en cuenta el peso relativo de cada clase.

Además, se evaluó el desempeño de otro algoritmo de clasificación llamado **AdaBoost**, especialmente reconocido por su eficacia en el manejo de conjuntos de datos desbalanceados. Al igual que XGBoost, AdaBoost funciona mediante la combinación de varios clasificadores débiles (normalmente árboles de decisión) para construir un clasificador fuerte ponderado. En cada iteración del proceso, AdaBoost incrementa el peso de las instancias que han sido clasificadas erróneamente, incentivando así a los clasificadores subsiguientes a concentrarse en ellas. Con la recalibración iterativa de los pesos de las instancias de entrenamiento, AdaBoost presta especial atención a las instancias difíciles de clasificar, que suelen corresponder a la clase minoritaria en conjuntos de datos desbalanceados. Los hiperparámetros más significativos en este algoritmo son el número de clasificadores débiles y la tasa de aprendizaje. Este clasificador se implementó mediante la clase `AdaBoostClassifier` de la librería `sklearn.ensemble`.

Se sospecha que varios de los índices propuestos se encuentren correlacionados ya que algunos son cocientes de otros, o hasta una combinación lineal en el caso de y_{mean} . Estas correlaciones se examinaron mediante la matriz de correlación de Kendall-Tau. Si se identificaban valores superiores a 0.6 en pares de índices, se consideró la posibilidad de remover uno de estos y evaluar el impacto en el desempeño del clasificador. Además, se consideró la aplicación del análisis de componentes principales (PCA, por sus siglas en inglés) sobre los índices altamente correlacionados.

PCA

El análisis de componentes principales es una técnica de gran valor al trabajar con conjuntos de datos que contienen variables altamente correlacionadas. En esencia, el PCA es una herramienta de reducción de dimensionalidad que transforma un conjunto original de variables correlacionadas en un conjunto nuevo de variables no correlacionadas llamadas componentes principales. Estos componentes se organizan de tal manera que el primero captura la mayor variación en los datos, seguido por el segundo componente, que captura la segunda mayor

variación, y así sucesivamente, así manteniendo la información más importante. Con PCA, es posible mantener la mayor parte de la información original de los datos, mientras se reduce el número de variables, lo que facilita la interpretación y potencialmente mejora el rendimiento de los modelos predictivos. El número de componentes a utilizar es variable y puede seleccionarse de distintos modos. En este trabajo se utilizó la cantidad mínima de componentes que logren mantener el 90% de la varianza de los datos. Esto se realizó seleccionando los primeros componentes principales hasta que la suma acumulativa de la varianza explicada alcance al menos el 90%. Este análisis de componentes principales se implementó usando la clase PCA de la librería `sklearn.decomposition`.

Por otra parte, la identificación de los hiperparámetros óptimos para el clasificador se llevó a cabo utilizando nuevamente el método *grid-search*. Se evaluaron los mismos hiperparámetros que en el caso del clasificador de ruido, añadiendo la consideración de si balancear o no el conjunto de datos de entrenamiento. Dado que en la base de datos de ectópicos, la cantidad de latidos no ectópicos supera notablemente a los latidos ectópicos, el conjunto de datos es altamente desbalanceado. Por lo tanto, es probable que el balanceo del conjunto de datos pueda tener un impacto significativo en los resultados del clasificador.

Las métricas de evaluación utilizadas para la clasificación de latidos fueron la precisión, sensibilidad, especificidad y F1-score descritas anteriormente.

En la detección de latidos ectópicos, minimizar los falsos positivos (latidos normales clasificados como ectópicos) es crucial para evitar intervenciones innecesarias y alarmas falsas. De ahí se decidió que la precisión, que mide la proporción de identificaciones de latidos ectópicos que son efectivamente correctas, sea la métrica prioritaria en este caso. Sin embargo, también se tuvieron en cuenta otras métricas como la sensibilidad y el F1-score, para proporcionar una visión más integral del rendimiento del modelo y ayudar a garantizar un equilibrio en su desempeño.

Cabe destacar que las pruebas descritas anteriormente se realizaron con el 80% de los datos, garantizando la independencia entre los pacientes. Para esto, se separaron las señales de 20% de los pacientes, es decir, 7 individuos, y se reservaron exclusivamente para la validación del

algoritmo. Todas estas pruebas se realizaron utilizando validación cruzada en k partes, con $k = 5$, y se mantuvo la independencia de los pacientes gracias a una implementación propia de la validación cruzada.

Dada la falta de otras bases de datos con un número suficiente de latidos normales y ectópicos anotados, no fue posible validar este detector con una base de datos distinta.

Finalmente, dado que el objetivo principal del detector de ectópicos es depurar las series para una clasificación más precisa de la FA, la utilidad de este detector se evaluó indirectamente a través del rendimiento del detector de FA sin y con detección de ruido y ectópicos.

4.4.2 Armado y viabilidad de la serie NN

Una vez que se detectan los latidos ectópicos en la señal, los intervalos RR asociados a estos latidos — es decir, el intervalo RR corto anterior y el compensatorio posterior — se eliminan de la serie RR, dando lugar a la serie NN. Si esta serie NN resultante consta de menos de 5 intervalos, se considera que cualquier análisis subsiguiente no sería confiable, por lo que se descarta.

4.5 Detección de FA

Como se mencionó en el marco teórico, la FA presenta tres características principales visibles en la señal de ECG: ritmo irregular y rápido, ausencia de ondas P y presencia de ondas F. Por lo tanto, la detección de esta arritmia se puede realizar mediante el análisis de la regularidad del ritmo cardíaco y/o a través de análisis morfológico, en el que se estudia la presencia o ausencia de las ondas mencionadas. Este detector se implementó a través del análisis de ritmo.

Inicialmente, se consideró añadir información morfológica para mejorar la detección, pero después de analizar las señales a utilizar, esto no se consideró factible. La utilidad de los parámetros morfológicos de la señal depende del nivel de ruido presente, razón por la que la mayoría de los detectores de FA existentes los ignoran [31]. Las ondas P son de muy baja amplitud y las ondas F lo son aún más. En señales con altos niveles de ruido, como las obtenidas por dispositivos de smartphone, es muy difícil distinguir la presencia o ausencia de estas ondas, ya que se suelen confundir con ruidos de amplitud similar. Al poseer solo información de una derivación, no se pueden utilizar los otros vectores eléctricos para corregir falsas detecciones. Además, la derivación con la que se cuenta no es la ideal para el análisis morfológico necesario para la FA, ya que la onda P es más visible en las derivaciones II y V1 [32]. Todo esto conduce a un alto nivel de falsas detecciones que, en última instancia, solo empeoraría el rendimiento del clasificador.

Este trabajo tiene como objetivo detectar episodios de FA en pacientes monitoreados a distancia, los cuales no serían fácilmente detectables en chequeos rutinarios. Dado que las señales utilizadas son de corta duración, alrededor de 30 segundos, es poco probable observar una transición de ritmo en dicho intervalo. Por lo tanto, se eligió evaluar la presencia de FA en las señales completas, sin subdividir las en ventanas de tiempo más pequeñas.

4.5.1 Índices de detección de FA

Para el análisis de las características del ritmo cardíaco, se usaron las series RR, de las que se extrajeron diversos parámetros de irregularidad. Estos parámetros se pueden clasificar en las siguientes categorías: parámetros de dispersión estadística, basados en puntos de inflexión,

basados en histogramas, de entropía, basados en el gráfico de Poincaré y basados en la función de coherencia espectral tiempo-variante [31]. Sin embargo, se descartaron los basados en histogramas y en la función de coherencia, ya que requieren señales largas con series RR de gran número de intervalos. En una señal de 60 segundos, que es el máximo en SSLAFDB, se necesitaría un ritmo rápido de 100 latidos por minuto (bpm, por sus siglas en inglés) para obtener apenas 100 intervalos RR. A continuación, se describen los distintos parámetros analizados. Todos estos fueron calculados mediante funciones de implementación propia.

Índices de dispersión estadística

El Coeficiente de Variación (CV) de una serie RR se define como [31][33]:

$$P_{CV} = \frac{\sigma_x}{m_x} \quad (4.30)$$

Donde σ_x y m_x representan el desvío estándar y la media de la serie RR, respectivamente. Este parámetro sirve para caracterizar la dispersión de la serie, y a la vez los cambios en la frecuencia cardíaca, ya que los acortamientos en los intervalos RR que usualmente existen en un episodio de FA llevan a una media menor.

Otro parámetro de dispersión ampliamente usado en la literatura [31][34] es la raíz cuadrada de la media de las diferencias sucesivas al cuadrado (RMSSD, por sus siglas en inglés), definido como:

$$P_{RMSSD} = \sqrt{\frac{1}{N-1} \sum_{n=1}^{N-1} \Delta x^2(n)} \quad (4.31)$$

Donde $\Delta x(n)$ es la serie de las diferencias sucesivas de la serie RR, calculada como $\Delta x(n) = x(n+1) - x(n)$, siendo $x(n)$ es la serie RR y N su largo. Además, existe una variación de este parámetro, llamada media normalizada de las diferencias sucesivas absolutas (NMA SD, por sus siglas en inglés), y se calcula de la siguiente forma:

$$P_{NMA SD} = \frac{\frac{1}{N-1} \sum_{n=1}^{N-1} |\Delta x(n)|}{m_x} \quad (4.32)$$

Al ser el cociente de un parámetro de dispersión dividido la media, este parámetro describe información similar al P_{CV} . Por ende, caracteriza tanto dispersión como cambios en el

ritmo cardíaco, y también al P_{RMSSD} . Estos tres parámetros aportan información muy similar, y presenten valores más elevados en caso de un episodio de FA, donde se espera una mayor variabilidad (lo que conduce a una mayor dispersión) y una mayor frecuencia cardíaca (resultando en una media menor).

Un último parámetro dentro de esta categoría es el porcentaje de intervalos RR sucesivos que difieren en más de k ms [35]. Este índice se define de la siguiente manera:

$$P_{NN_k} = \frac{\#(\Delta x(n) > k)}{N} \quad (4.33)$$

Donde $\#(\Delta x(n) > k)$ es la cantidad de intervalos RR sucesivos que difieren en más de k milisegundos con respecto al siguiente. Se probó utilizar 2 variantes de este índice: una con una diferencia de 50 ms ($P_{NN_{50}}$) y otra con una diferencia de 100 ms ($P_{NN_{100}}$).

Cantidad de puntos de inflexión

La prueba de puntos de inflexión es un test estadístico no paramétrico que sirve para determinar si un conjunto de variables aleatorias son independientes e idénticamente distribuidas [31][36]. Un punto de inflexión en una serie se produce donde la serie pasa de crecer a decrecer, o viceversa, identificándose como un máximo o mínimo local, respectivamente.

En una serie con N muestras, se pueden contar la cantidad de puntos de inflexión (N_{IP}), calcular el número esperado de puntos de inflexión (m_{IP}) y comparar a cuantos desvíos estándar (σ_{IP}) se encuentra la cantidad de puntos de inflexión real, de la esperada.

En una serie completamente aleatoria, cualquier conjunto de 3 muestras sucesivas tiene la misma posibilidad de ocurrir en cualquiera de los 6 ordenes posibles. En cuatro de estos ordenes, la muestra central es un máximo o un mínimo local (osea, un punto de inflexión), por lo que la probabilidad de que haya un punto de inflexión cada 3 muestras es $2/3$. Teniendo esto en mente, m_{IP} y σ_{IP} se pueden calcular de la siguiente forma:

$$m_{IP} = \frac{2(N - 2)}{3} \quad (4.34)$$

$$\sigma_{IP} = \sqrt{\frac{16N - 29}{90}} \quad (4.35)$$

Este test consiste en que, si N_{PI} se encuentra fuera del intervalo $m_{IP} \pm 1.96 \sigma_{IP}$, la serie no se puede considerar completamente aleatoria. Si la serie tiene ritmo FA, entonces la serie tiende a ser aleatoria. En casos de otros ritmos, la serie debería ser más determinística (osea, N_{IP} y m_{IP} deberían diferir por un margen representativo).

Los índices de FA empleados en este trabajo son variables numéricas continuas, que serán introducidas en un clasificador de aprendizaje automático. Dado que este test arroja resultados binarios (serie aleatoria o no aleatoria), se pierde información respecto a qué grado de aleatoriedad presenta la serie, por lo que se decidió que su inclusión no era la opción más adecuada. No obstante, la idea subyacente de este test se consideró útil y se propuso un índice de elaboración propia. Este se nombró la diferencia relativa de puntos de inflexión P_{IPD} y se calcula mediante la siguiente fórmula:

$$P_{IPD} = \frac{N_{IP} - m_{IP}}{\sigma_{IP}} \quad (4.36)$$

Parámetros de entropía

La entropía de Shannon (ShanEn) es un parámetro ampliamente utilizado. Esta cuantifica la cantidad de incertidumbre de la información que brinda un mensaje, que en este caso sería la serie RR [31]. La entropía de Shannon de una variable x define del siguiente modo:

$$P_{ShanEn} = - \sum_{i=1}^B p(x_i) \log_2(p(x_i)) \quad (4.37)$$

Donde B representa la cantidad de posibles valores que puede tomar x y $p(x_i)$ es la probabilidad de cada valor. Esta última se estima calculando la frecuencia relativa de cada uno de los posibles valores de x :

$$p(x_i) = \frac{\# x_i}{N} \quad (4.38)$$

Donde $\# x_i$ es la cantidad de veces que x_i aparece en la serie y N es el largo de la misma. Esta estimación se suele realizar mediante un histograma donde el número de *bins* equivale a B . Este número se determinó de forma experimental, empezando con 5 y aumentándolo hasta encontrar el número que permitiese una separación óptima de clases, que en este caso fue 10.

Valores elevados de entropía indican una alta incertidumbre, mientras que valores bajos representan una baja incertidumbre. En el análisis de ritmos cardíacos, la presencia de FA se asocia con valores más altos de P_{ShanEn} en comparación con el ritmo sinusal [31].

La entropía de muestreo (SampEn, por su nombre en inglés) refleja la autosimilitud (o autosemejanza) de la señal. Dado que los ritmos de FA son inherentemente irregulares, los intervalos RR varían considerablemente entre sí, resultando en una baja autosimilitud y, en consecuencia, en una alta entropía de muestreo. En contraste, el ritmo sinusal, que es más regular, tiende a presentar una alta autosimilitud y una entropía de muestreo menor. Este parámetro se define de la siguiente manera:

$$P_{SampEn} = -\ln\left(\frac{B(m+1, r)}{B(m, r)}\right) \quad (4.39)$$

Donde $B(m, r)$ es la probabilidad condicional de que pares de secuencias coincidan por m muestras dentro de un rango de tolerancia r . $B(m, r)$ se puede aproximar con la siguiente formula:

$$B(m, r) = \frac{1}{(N-m)(N-m-1)} \sum_{i=0}^{N-m} \sum_{j=0, j \neq i}^{N-m-1} H(r - \|x(i) - x(j)\|_{\infty}) \quad (4.40)$$

Donde $\|\cdot\|_{\infty}$ representa la norma máxima y N el tamaño de la serie. Se elió usar el valor de $m = 1$ para simplificar los cálculos, ya que la norma correspondiente es la norma 1, y además porque en la bibliografía se encontró como el valor que da mejores resultados en series RR cortas [37]. Finalmente, $B(m = 1, r)$ queda:

$$B(m = 1, r) = \frac{2}{(N-1)(N-2)} \sum_{i=0}^{N-2} \sum_{j=0, j \neq i}^{N-1} H(r - |x(i) - x(j)|) \quad (4.41)$$

El valor de r se encontró iterando, iniciando en 0.03 y aumentando en incrementos de 0.005. El valor óptimo se determinó como 0.05.

De P_{SampEn} se desprende un coeficiente similar, llamado coeficiente de la entropía de muestreo (CSampEn, por sus siglas en inglés). Este se define de la siguiente forma:

$$P_{CSampEn} = P_{SampEn} + \ln(2r) - \ln(m_x) \quad (4.42)$$

Donde m_x representa la media de la serie RR y se lo incluye para tomar en cuenta el ritmo cardíaco rápido característico de la FA.

Un índice más sencillo relacionado a los anteriores es la entropía de muestreo simplificada (SSamEn, por sus siglas en inglés). En este índice, no se utiliza $B(m + 1, r)$ sino que solo $B(m, r)$, y la norma máxima se sustituye por la norma Euclideana. Además, se sugiere usar un $m = 1$, por lo que la norma Euclideana queda igual a la norma 1 [38]. La fórmula de la entropía de muestreo es la siguiente:

$$P_{SSampEn} = \frac{B(m = 1, r)}{m_x} \quad (4.43)$$

Índices basados en el gráfico de Poincaré

El gráfico de Poincaré es un gráfico de dispersión de intervalos sucesivos de una serie numérica $(x(n), x(n + 1))$. Alternativamente, también puede tomarse los gráficos $(\Delta x(n), \Delta x(n + 1))$ o $(x(n), \Delta x(n))$. Los gráficos de Poincaré de las series RR de un ECG permiten caracterizar distintos ritmos cardíacos.

Como las series RR de señales con ritmo sinusal son regulares, los puntos en el gráfico de Poincaré se aglomeran en una región pequeña (Figura 4.11 a, b, y c). En el gráfico $(x(n), x(n + 1))$, esta acumulación se ubica en algún punto sobre la línea de identidad del gráfico, es decir, $x(n) = x(n + 1)$, y este punto depende de la magnitud del ritmo cardíaco (Figura 4.11a). En el caso de $(\Delta x(n), \Delta x(n + 1))$, los valores suelen ser cercanos a cero y por ende, los puntos del gráfico se aglomeran en el origen (Figura 4.11b). Por último, en el gráfico $(x(n), \Delta x(n))$, los puntos se acumulan en un punto sobre el eje x, es decir, la línea $\Delta x(n) = 0$, y el valor de x de este punto depende nuevamente de la magnitud del ritmo cardíaco (Figura 4.11c). En contraste, como los ritmos FA tienen un grado alto de variabilidad, su gráfico de Poincaré tiende a estar mucho más disperso que el de un ritmo sinusal (Figura 4.11 d, e, y f) [31][39].

Los gráficos de Poincaré se analizan a través de parámetros que reflejan la densidad de puntos en las diferentes regiones del gráfico y parámetros que caracterizan la geometría de los puntos en el mismo.

Para evaluar la densidad de puntos en las diversas regiones del gráfico, éste se divide con una grilla cuadrada, cuyas celdas se utilizan como *bins* de un histograma bidimensional. El

tamaño del *bin* debe ser lo suficientemente pequeño para que se forme un alto número de *bins*, pero no tanto como para que demasiados *bins* resulten vacíos. Entonces, el tamaño ideal del *bin* depende de la longitud de la serie, ya que series de menor número de intervalos logran poblar menor cantidad de *bins*. Para determinar este tamaño óptimo, se inició con un valor pequeño de 0.025 s, incrementándolo en pasos de 0.025 s hasta encontrar el valor más adecuado para este tipo de señales, que resultó ser 0.125 s.

Los parámetros sobre la densidad de puntos en diferentes regiones del gráfico de Poincaré se basan en el número de *bins* poblados por al menos un punto, o *bins* no vacíos (“non-zero bins”), y se designan P_{NZPP} por sus siglas en inglés. Como las señales con FA producen gráficos de Poincaré con mayor grado de dispersión, estas se asocian a un número de *bins* poblados mucho mayor que las de ritmo sinusal, cualquiera sea la versión del gráfico.

El gráfico de Poincaré más utilizado para la detección de FA es el de $(\Delta x(n), \Delta x(n+1))$, donde las series con ritmo sinusal suelen poblar la región del origen (Figura 4.11b), mientras que las de FA suelen estar más dispersas por todo el gráfico (Figura 4.11e). Entonces, el parámetro utilizado para estudiar la presencia de FA en este gráfico (P_{NZPP_2}) se calcula restando al número de *bins* no vacíos, el número de *bins* no vacíos de la región del origen [31]. Como en este caso se opera con un tamaño de *bin* relativamente grande, se considera que la región del origen se constituye por los 4 *bins* centrales.

Como se explicó en la sección 4.4, P_{NZPP_2} también se usa para la detección de ectópicos. Un ejemplo es la Figura 4.11 h), donde se muestra el gráfico de Poincaré de una señal con ritmo O que incluye latidos ectópicos. Se observa que los latidos predominan en el centro, pero también se encuentran en dos zonas adicionales, representando el patrón de intervalos RR cortos seguidos por intervalos largos, lo cual es indicativo de un ritmo con latidos ectópicos.

En el caso de los gráficos $(x(n), x(n+1))$ y $(x(n), \Delta x(n))$, los parámetros utilizados son simplemente el número de *bins* no vacíos (P_{NZPP_1} y P_{NZPP_3} , respectivamente).

Los parámetros que caracterizan la geometría de los puntos del gráfico de Poincaré se basan en el concepto de que en ritmos sinusales, los puntos del gráfico $(x(n), x(n+1))$ se suelen ubicar al rededor de la línea de identidad ($(x(n) = x(n+1))$), agrupándose en una forma similar a una

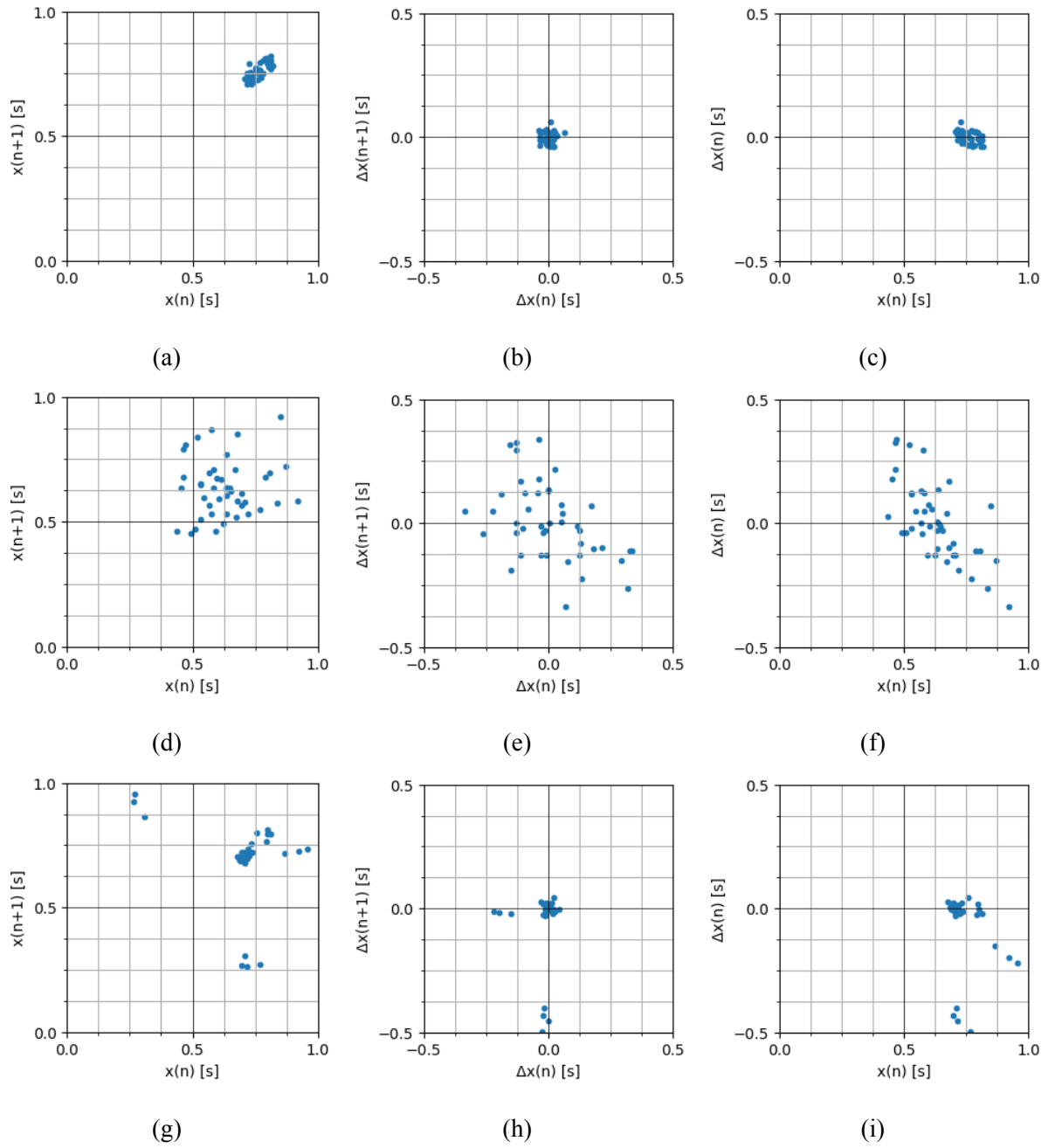


Figura 4.11. Distintos gráficos de Poincaré de las series RR provenientes de las señales:
a-c) n° 0, ritmo N. d-f) n° 8, ritmo FA. g-i) n° 7, ritmo O.

elipse [31]. Estos parámetros geométricos buscan estudiar la forma de esta agrupación.

El eje mayor de esta elipse coincide con la línea de identidad del gráfico. Entonces, la dispersión de los puntos sobre los ejes de la elipse puede estudiar realizando una rotación de

45°, definida de la siguiente forma:

$$\begin{bmatrix} y(n+1) \\ y(n) \end{bmatrix} = \begin{bmatrix} \sin(\pi/4) & \cos(\pi/4) \\ \cos(\pi/4) & -\sin(\pi/4) \end{bmatrix} \begin{bmatrix} x(n+1) \\ x(n) \end{bmatrix} \quad (4.44)$$

Donde $y(n)$ y $y(n+1)$ son transformaciones de $x(n)$ y $x(n+1)$, respectivamente [31].

Realizando la multiplicación anterior, las fórmulas para calcular $y(n)$ y $y(n+1)$ quedan:

$$y(n+1) = \frac{1}{\sqrt{2}} (x(n+1) + x(n)) \quad (4.45)$$

$$y(n) = \frac{1}{\sqrt{2}} (x(n+1) - x(n)) = \frac{\Delta x(n)}{\sqrt{2}} \quad (4.46)$$

En principio, la forma de la elipse se puede caracterizar mediante la desviación estándar de $y(n)$ y $y(n+1)$. Estos parámetros ($\sigma_{y,0}$ y $\sigma_{y,1}$, respectivamente) se calculan mediante la siguiente ecuación:

$$\sigma_{y,j} = \sqrt{\frac{1}{N-1} \sum_{n=0}^{N-2} (y(n+j) - m_y)^2}, \quad j = 0, 1 \quad (4.47)$$

Donde m_y es la media de $y(n)$. $\sigma_{y,0}$ se asocia con la irregularidad de la serie RR en el corto plazo mientras que $\sigma_{y,1}$ se relaciona con la variabilidad en el largo plazo. Como la irregularidad en señales con FA se manifiesta en el corto plazo, $\sigma_{y,1}$ pierde relevancia y se descarta [31].

Por otro lado, $\sigma_{y,0}$ si se suele utilizar para detección de FA. Dada la regularidad de los intervalos RR en señales con ritmo sinusal y la naturaleza aleatoria de la irregularidad de los intervalos en señales con FA, la media de $\Delta x(n)$ resulta muy cercana a cero. Por la relación establecida en (4.46), m_y , la media de $y(n)$, también se acerca a cero. Entonces, $\sigma_{y,0}$ se puede aproximar de la siguiente forma:

$$\sigma_{y,0} \approx \sqrt{\frac{1}{N-1} \sum_{n=1}^{N-1} \left(\frac{\Delta x(n)}{\sqrt{2}} \right)^2} = \sqrt{\frac{1}{2(N-1)} \sum_{n=1}^{N-1} \Delta x^2(n)} \quad (4.48)$$

Observando la fórmula anterior, se puede notar que esta aproximación de $\sigma_{y,0}$ equivale al parámetro P_{RMSSD} , multiplicado por un factor de $\frac{1}{\sqrt{2}}$. Esto es razonable ya que ambos terminan siendo parámetros de dispersión de la serie RR.

Otro parámetro geométrico del gráfico de Poincaré es la media de las distancias Euclidianas de la serie RR (σ_{ED}). Este se basa en las distancias Euclidianas entre 2 puntos sucesivos

$(x(n), x(n+1))$ y $(x(n+1), x(n+2))$, y representa la tasa de cambio local de la serie RR. Este parámetro se calcula de la siguiente manera:

$$\sigma_{ED} = \frac{1}{N-2} \sum_{n=1}^{N-2} \sqrt{\Delta x^2(n) + \Delta x^2(n+1)} \quad (4.49)$$

Al igual que en el caso de $P_{CSampEn}$, para que estos parámetros relacionados a la irregularidad absoluta de la serie RR puedan usarse exitosamente para la detección de FA, deben normalizarse por la media de la serie RR [31]. Como consecuencia, se definen los últimos dos parámetros:

$$P_{\sigma_{y,0}} = \frac{\sigma_{y,0}}{m_x} \quad (4.50)$$

$$P_{\sigma_{ED}} = \frac{\sigma_{ED}}{m_x} \quad (4.51)$$

4.5.2 Clasificación de FA

Una vez calculados los índices de FA de todas las señales de las bases de datos, se procedió a la elección de la combinación de índices, tipo de clasificador e hiperparámetros óptimos para la clasificación de FA. Esto se realizó de manera análoga a la etapa de clasificación de ruido (sección 4.2.3), donde primero se descartaron los índices con un índice de correlación de Kendall-Tau mayor a 0.60, luego se procedió a la combinación óptima de parámetros y el clasificador utilizando el método *wrapper* y eligiendo en base a la exactitud, especificidad, sensibilidad y F1-score. Finalmente, los hiperparámetros óptimos se encontraron mediante *grid-search*. Los clasificadores probados fueron SVM, regresión logística, KNN, XGBoost y AdaBoost.

Cabe destacar que las iteraciones de los métodos *wrapper* y *grid-search* se realizaron con el 80% de los datos separados para entrenamiento, ya que el 20% restante se reservó para la validación.

El conjunto de datos completo consta de 639 muestras de clase FA y 6939 de clase no FA, de los cuales 4668 son ritmo N y 2271 son O. El notable desbalance puede dificultar la clasificación de FA, ya que esta clase está subrepresentada. Además, existe la diversidad en la categoría no FA, compuesta por los ritmos N y O en distintas proporciones. Por lo tanto, se optó por optimizar

y entrenar dos clasificadores diferentes: uno con el conjunto de entrenamiento completo y otro con una versión balanceada del mismo. El conjunto de entrenamiento balanceado se obtuvo seleccionando todas las muestras de FA de éste y un número equivalente de muestras no FA, escogiendo ritmos N y O en proporciones iguales de forma aleatoria.

Por último, para evaluar la utilidad del detector de ruido y de ectópicos en el algoritmo, se testeó la detección de FA sin utilizar el detector de ruido ni el de ectópicos.

4.5.3 Tiempo de cómputo y uso de memoria

Para analizar el tiempo de cómputo y uso de memoria, el algoritmo se corrió en una MacBook Pro (13-inch, M1, 2020) con memoria RAM de 16 GB y versión OS 12.2.1. Para calcular el tiempo se usó la función `time` de la librería `time`, y para analizar el uso de memoria se utilizó el decorador `@profile` de la librería `memory_profiler`. Estos valores se calcularon para cada una de las señales del conjunto de validación, y los valores reportados representan sus respectivos promedios.

4.6 Funcionalidades adicionales

A lo largo de las reuniones llevadas a cabo con la empresa Virtual Sense, se identificó la utilidad de implementar funcionalidades adicionales a la detección de FA. Al igual que el detector de FA, estas nuevas funcionalidades fueron desarrolladas para integrarse en la plataforma de telemedicina de Virtual Sense. A continuación se detallan las funcionalidades añadidas.

Ritmo cardíaco

El ritmo cardíaco (HR, por sus siglas en inglés), también conocido como frecuencia cardíaca, es la cantidad de latidos por minuto que realiza un corazón. Es una medida que refleja la velocidad de estos latidos y se expresa en bpm.

Tradicionalmente, el HR se mide contando la cantidad de pulsaciones que se perciben al palpar la arteria radial en la muñeca o la arteria carótida en el cuello, utilizando los dedos índice y medio [40]. No obstante, el ECG proporciona un método más preciso para medir el HR, al observar la frecuencia y regularidad de los complejos QRS en el trazado del ECG. En este contexto, el HR se calcula a partir de los intervalos RR de la señal.

Se distinguen dos tipos de HR: el instantáneo (iHR) y el promedio (HR). El iHR se calcula dividiendo 60 por el intervalo RR correspondiente a cada instante de tiempo, es decir:

$$iHR = \frac{60}{x(i)} \quad (4.52)$$

Donde $x(i)$ representa el intervalo RR en el instante i . Sin embargo, como en este trabajo se cuenta con señales de ECG de aproximadamente 30 segundos de duración, se planteó calcular el HR promedio. Este se calcula de forma análoga, utilizando el intervalo RR promedio:

$$HR = \frac{60}{m_x} \quad (4.53)$$

Donde m_x es la media de la serie de intervalos RR.

El ritmo cardíaco de una persona varía debido a numerosos factores como la actividad física, el estrés, las respuestas emocionales, la dieta, la hidratación, el nivel de condición

física, la genética, la presencia de ciertas patologías y el uso de medicación. De todas formas, normalmente se considera que un ritmo cardíaco en reposo es normal si cae dentro del rango de 60 a 100 bpm [41].

Variabilidad del ritmo cardíaco

La variabilidad del ritmo cardíaco (HRV, por sus siglas en inglés) se refiere a la variación en los intervalos RR. Esta medida es influenciada por el sistema nervioso autónomo, a través de sus componentes simpático y parasimpático. La HRV es un indicador importante de la adaptabilidad del corazón al estrés y puede proporcionar información valiosa sobre la salud cardiovascular y el funcionamiento autonómico. Un HRV bajo puede sugerir un sistema cardiovascular menos saludable y una capacidad de respuesta al estrés disminuida. En contraste, un HRV alto se considera más saludable [42].

Existen diversas medidas de HRV, entre ellas SDNN, RMSSD y pNN50. Las dos últimas métricas son algunos de los índices de dispersión estadística de FA mencionados en la sección 4.5.1. La medida de HRV más ampliamente utilizada es la desviación estándar de los intervalos NN (SDNN por sus siglas en inglés). Esta se asemeja al índice P_{CV} , pero difiere en que es una medida absoluta, expresada en milisegundos, en vez de relativa.

El SDNN se calcula de la siguiente forma:

$$SDNN = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x(i) - m_x)^2} \quad (4.54)$$

Donde $x(i)$ es el intervalo i de la serie NN, m_x es la media de la serie y N es el largo de la misma.

Alarmas de otros ritmos anormales

Virtual Sense tiene como objetivo integrar a su plataforma la función de detectar de arritmias varias a partir de la señal de ECG. Para ello, se desarrollaron algoritmos que procesan esta señal,

detectan posibles arritmias y etiquetan la señal con la arritmia correspondiente. Estas etiquetas activan alarmas que notifican a los responsables del telemonitoreo de pacientes.

Uno de los planteos fue la detección de bradicardia y la taquicardia. La bradicardia se refiere a un ritmo cardíaco lento, en el cual el HR en reposo es inferior a 60 bpm. En contraposición, la taquicardia hace referencia a un ritmo cardíaco acelerado, que se da cuando el HR excede los 100 bpm [43].

Para lograr esta detección, se comparó el HR promedio calculado con estos dos umbrales. Si el HR promedio es menor a 60 bpm, se asigna una etiqueta de bradicardia. De forma similar, si supera los 100 bpm, se asigna una etiqueta de taquicardia.

Otro aspecto del ECG que se quiso analizar fue la irregularidad del ritmo. Como se observó a lo largo del trabajo, la FA se asocia con un ritmo altamente irregular, con una naturaleza aleatoria. Además, el detector de FA desarrollado en este trabajo se basa únicamente en la irregularidad del ritmo cardíaco. Por consiguiente, se propuso utilizar el mismo detector de FA para identificar el ritmo irregular. El clasificador de FA, basado en la regresión logística, puede proporcionar un valor entre 0 y 1, que representa la probabilidad de que la señal pertenezca a la clase FA. Según la teoría detrás de la regresión logística, cuando esta probabilidad supera el umbral de 0.5, se clasifica la señal como FA. Sin embargo, cuando la probabilidad no alcanza 0.5 pero sí supera 0.3, en lugar de catalogar la señal como no FA, se decidió asignarle la etiqueta de ritmo irregular. Esta decisión se tomó con la intención de identificar y marcar aquellos ritmos que, sin llegar a ser FA, presentan un nivel de irregularidad relevante.

Resultados

5.1 Detección de ruido

5.1.1 Índices de ruido

En la Figura 5.1 se muestran los histogramas de los índices de ruido, para las ventanas ruidosas (rojo) y no ruidosas (azul) escogidas. La mediana de cada histograma se muestra mediante una línea punteada roja o azul según la clase a la que corresponde. Se utilizó el subíndice *bp* para diferenciar a los índices calculados con el filtro pasabanda, de los calculados sin el pasabanda y con el filtro pasabajos.

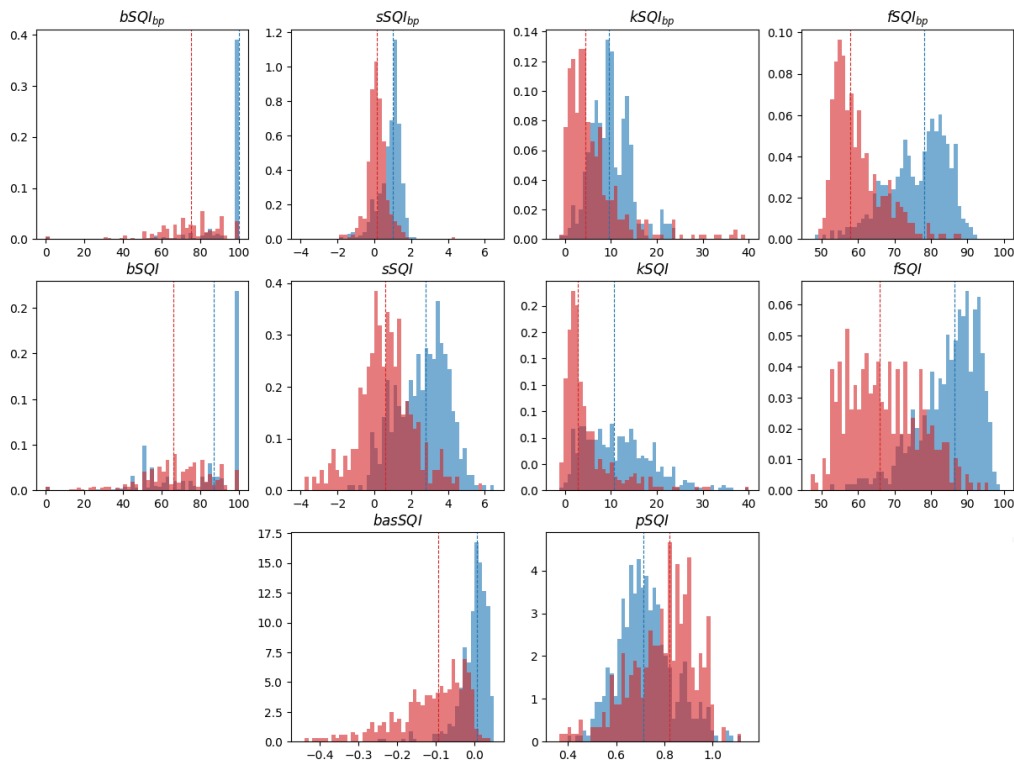


Figura 5.1. Histograma de los índices de ruido de implementación propia usando ventanas de 5 segundos ruidosas (rojo) y no ruidosas (azul). Las líneas punteadas muestran la mediana de cada clase.

Los histogramas muestran que bSQI con pasabanda es el índice que parece separar mejor las clases. Como se esperaba, la concordancia en la detección de latidos es del 100% en la mayoría de los casos para las ventanas no ruidosas. Se observa también una mejor separación que en el caso de bSQI sin pasabanda, donde la mediana cae cerca del 85% para las ventanas no ruidosas y se observa una presencia más amplia en casi todos los valores. Esto tiene sentido, ya que el filtro pasabanda está diseñado para seleccionar las bandas de frecuencia del complejo QRS, eliminando así mucha información que podría interferir con la detección y facilitando la correcta identificación de los latidos.

En el caso de los índices kSQI y sSQI, tanto con filtro pasabanda como sin él, se observa el comportamiento esperado de ruido aleatorio con distribución normal. Para las ventanas ruidosas, los valores de *skewness* se agrupan principalmente alrededor del 0 y la curtosis se acerca más al 3. Por otro lado, para las ventanas no ruidosas, la *skewness* se agrupa alrededor del 1 y la curtosis cerca del 10 cuando se aplica el pasabanda. Sin el pasabanda, la *skewness* se agrupa cerca del 2.5 y mientras que la curtosis también ronda el valor 10.

El comportamiento del fSQI, tanto con filtro pasabanda como sin él, cumple también con las expectativas. En las ventanas no ruidosas, la señal se asemeja más a una línea recta con una mediana cercana a 80 y 90 con y sin el filtro pasabanda respectivamente. En contraste, en las ventanas ruidosas, la mediana se sitúa alrededor de 60 y 65 con y sin el filtro pasabanda correspondientemente. Se esperaba una disminución en el porcentaje de la señal que asemeja a una línea recta al aplicar el pasabandas debido al fenómeno conocido como *ringing*. Este se refiere a un artefacto indeseable que ocurre cuando se aplica un filtro a una señal. El *ringing* se caracteriza por la presencia de oscilaciones o sobrepicos en la señal filtrada, que suelen aparecer justo después de un cambio o borde abrupto en la señal original [44], lo que lleva a menos zonas como líneas rectas.

Sorprendentemente, el basSQI muestra un alto grado de separación, por lo que parece ser un índice muy prometedor para la detección del ruido y debería considerarse en las etapas posteriores. Como se explicó en la sección de métodos, es posible que este índice, aunque refleje información de zonas de baja frecuencia que son filtradas por el filtro pasabanda luego, sea útil ya que estos ruidos de baja frecuencia pueden proporcionar información sobre ruidos

que se encuentran en la banda de interés y ayudar al contraste para su detección. Como era de esperar, los valores de basSQI para las ventanas no ruidosas se aproximan a cero, mientras que para las ventanas ruidosas estos se alejan de dicho punto.

Finalmente, el histograma que parece tener la mayor superposición de clases es el del pSQI. Sin embargo, las medianas de cada clase muestran una cierta separación, por lo que será considerado para los análisis posteriores, al igual que todos los demás índices.

En la Figura 5.2 se ilustran los resultados de la correlación de Kendall-Tau entre los índices, con y sin filtro pasabanda. Como puede verse en la barra de colores en la derecha de la figura, las correlaciones positivas se muestran en tonos rojos, las nulas en blanco y las negativas en tonos azules. Recordemos que los valores cercanos a 0 representan que no hay correlación mientras que mientras más cercana a 1 en valor absoluto la correlación es mayor.

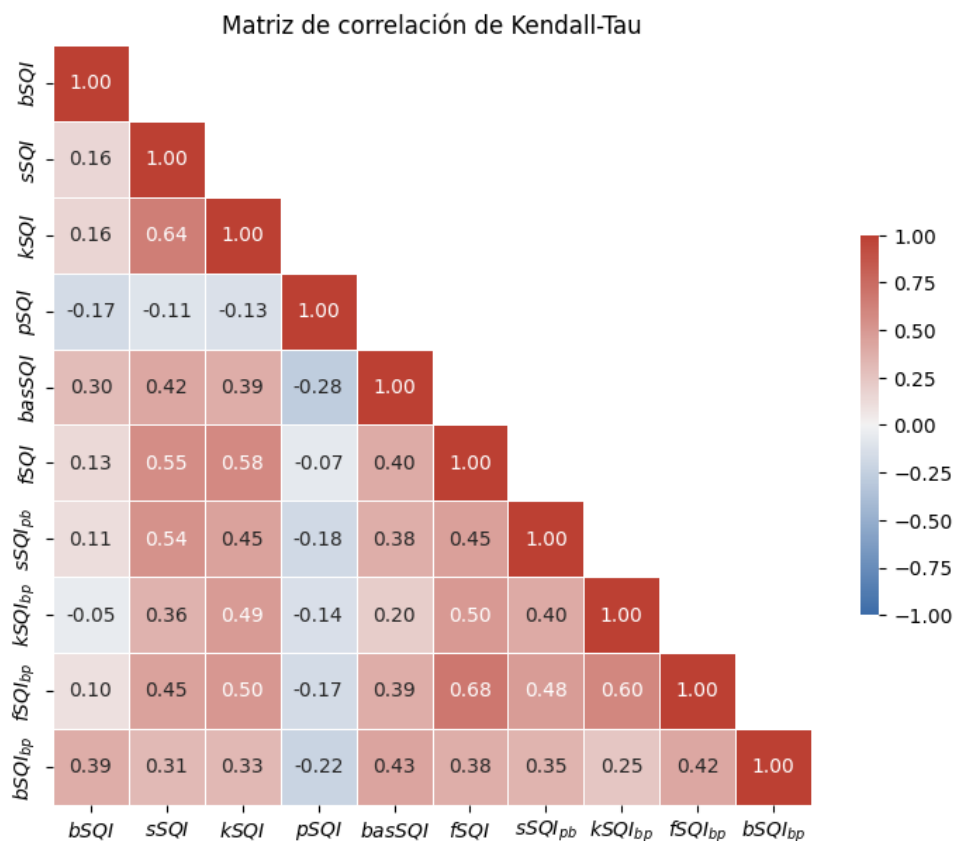


Figura 5.2. Matriz de correlación de Kendall-Tau de todos los índices con y sin filtro pasabanda, cuando corresponde.

En esta matriz, es evidente que existen correlaciones fuertes entre el fSQI y su versión con pasabanda. Esto es razonable y no representa un problema, ya que, como se explicó previamente, se descartaron de antemano las combinaciones que involucran a un índice y su versión con pasabanda. Otra correlación notable se da entre el sSQI y el kSQI sin pasabanda, que presenta un valor de Kendall-Tau de 0.64. Por lo tanto, se evitarán combinaciones que incluyan estos dos índices simultáneamente. Sin embargo, el resto de las combinaciones de estos índices con y sin pasabanda muestran una correlación moderada que no llega al umbral de 0.6, por lo que sí se tendrán en cuenta. Por ejemplo, se podría seleccionar una combinación de índices que incluya el sSQI sin pasabanda y el kSQI con pasabanda, o incluso ambos con pasabanda, dado que su correlación es de 0.4. Se decidió tomar todo el resto de las combinaciones en cuenta, aún para los casos donde se encontraron valores moderados cercanos al umbral de correlación fuerte, como el kSQI y fSQI sin pasabanda con 0.58.

Es importante destacar que el pSQI, basSQI y el bSQI con y sin pasabanda mostraron bajas correlaciones con todos los otros índices, sin alcanzar en ningún caso valores absolutos de 0.5. Finalmente, se observó que el pSQI y el bSQI sin pasabanda presentaron correlaciones débiles con las demás variables (solo la correlación del bSQI con el basSQI llegó hasta 0.3). Por lo tanto, parecen ser los índices que más información independiente brindan de todo el conjunto.

5.1.2 Optimización de índices, hiperparámetros y clasificador

Se recuerda que la selección de la combinación de índices e hiperparámetros óptima para cada tipo de clasificador evaluado se realizó mediante los métodos *wrapper* y *gridsearch*. Estos son métodos exhaustivos que se basan en entrenar un modelo con cada combinación de índices e hiperparámetros, y calcular las métricas de evaluación usando validación cruzada, con el fin de encontrar la combinación con mejor rendimiento. En la Tabla 5.1 pueden verse los mejores modelos para cada uno de los clasificadores, con las combinaciones de índices e hiperparámetros utilizados en cada caso.

En RL, se obtuvo un empate en rendimiento para la misma combinación de índices, tanto con el solver 'saga' como con 'liblinear' de la librería sklearn. Se decidió tomar este último dado que es más veloz que 'saga' en el caso de datasets pequeños, según la documentación

de `sklearn.linear_model.LogisticRegression` [45]. En XGBoost y RL, la combinación óptima de índices incluye a $kSQI_{bp}$ y $fSQI_{bp}$ simultáneamente. En la Figura 5.2 se puede observar que la correlación de Kendall-Tau para estos dos índices es de 0.6, al límite del umbral elegido. De todas formas, se decidió usarlos en conjunto ya que estaban presentes en por lo menos las mejores 6 combinaciones de cada uno de estos clasificadores, priorizando el rendimiento del clasificador por sobre la correlación.

| Clasificadores | Índices | Hiperparámetros | Exactitud | Especificidad | Sensibilidad | F1-Score |
|----------------|---|---|--------------------|--------------------|--------------------|--------------------|
| SVM | $bSQI_{bp}$ $kSQI_{bp}$ $fSQI$ $basSQI$ | $C = 1$ $\gamma = 1$ kernel = 'rbf' | $93.68 \pm 2.26\%$ | $92.80 \pm 1.81\%$ | $94.80 \pm 3.46\%$ | $93.59 \pm 2.57\%$ |
| KNN | $bSQI_{bp}$ $sSQI_{bp}$ $kSQI_{bp}$ $fSQI$ $pSQI$ $basSQI$ | $k = 8$ weights = 'distance' $p = 1$ | $94.38 \pm 2.59\%$ | $94.67 \pm 2.80\%$ | $94.10 \pm 4.23\%$ | $93.61 \pm 2.95\%$ |
| XGBoost | $bSQI_{bp}$ $kSQI_{bp}$ $fSQI_{bp}$ $pSQI$ $basSQI$ | learning_rate = 0.01 max_depth = 7 n_estimators = 200 | $93.52 \pm 2.12\%$ | $93.96 \pm 2.08\%$ | $92.89 \pm 4.26\%$ | $93.37 \pm 2.17\%$ |
| RL | $kSQI_{bp}$ $fSQI_{bp}$ $pSQI$ $basSQI$ | penalty = 'L1' $C = 1.0$ solver = 'liblinear' | $92.77 \pm 2.01\%$ | $93.60 \pm 1.31\%$ | $91.67 \pm 4.51\%$ | $92.63 \pm 2.51\%$ |

Tabla 5.1. Resultados de las mejores combinaciones de atributos e hiperparámetros para cada clasificador.

Observando la Tabla 5.1 puede verse que el clasificador que mejor exactitud y F1-score tuvo (utilizando validación cruzada) fue KNN. En segundo lugar queda SVM, seguido por XGBoost, dejando a RL en último lugar.

Como se explicó en la sección 4.2.3, el modelo de clasificación de ruido no se utilizará para hacer una predicción binaria sobre si una ventana es ruidosa o no, sino para estimar la probabilidad de que la misma sea ruidosa. Las clasificaciones de ruido se obtienen luego de

suavizar y redondear estas probabilidades. En los modelos SVM, las probabilidades de las clases se determinan ajustando una regresión logística a los puntajes de decisión. No obstante, este procedimiento puede ser computacionalmente exigente y las probabilidades resultantes pueden no ser tan precisas como en modelos que estiman intrínsecamente las probabilidades. En consecuencia, dichas probabilidades pueden ser inconsistentes con la predicción del modelo, lo que hace que las métricas obtenidas no sean completamente representativas del rendimiento de SVM para predecir probabilidades. Por estas razones, y debido a que SVM no generó las mejores métricas, se decidió descartarlo.

Por otra parte, el hecho de que tanto XGBoost como RL utilizan conjuntamente los índices $kSQI_{bp}$ y $fSQI$ no fue considerado como motivo suficiente para su descarte. Sin embargo, dado que estos modelos tampoco mostraron las métricas más destacadas, también se optó por descartarlos.

Finalmente, el clasificador elegido fue KNN, con 8 vecinos, distancias ponderadas, y $p = 1$, que implica que las distancias calculadas son distancias Manhattan. Las métricas de evaluación reportadas para este clasificador, utilizando el 20% de los datos que se habían apartado como conjunto de validación, fueron: exactitud del 91.62%, especificidad del 92.55%, sensibilidad del 90.41% y F1-score del 90.41%. La matriz de confusión y la curva ROC obtenidas se muestran en la Figura 5.3.

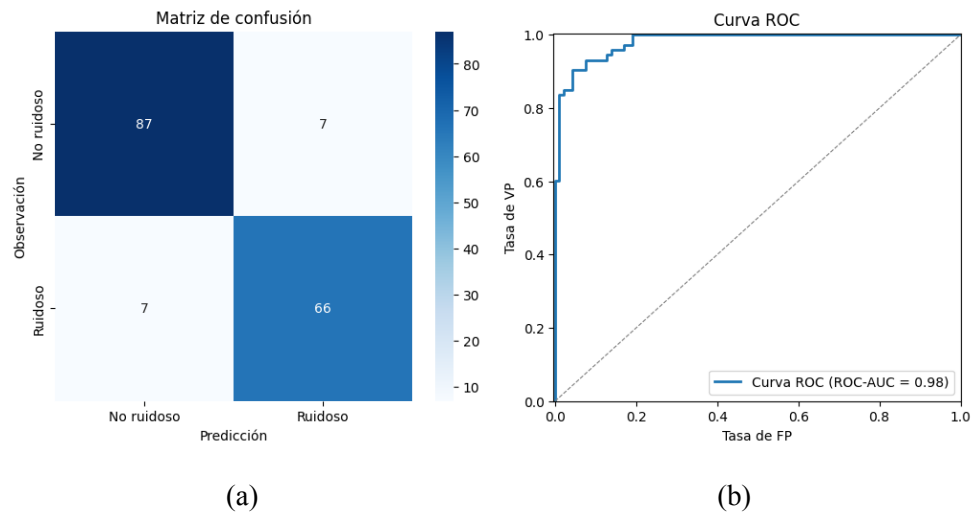
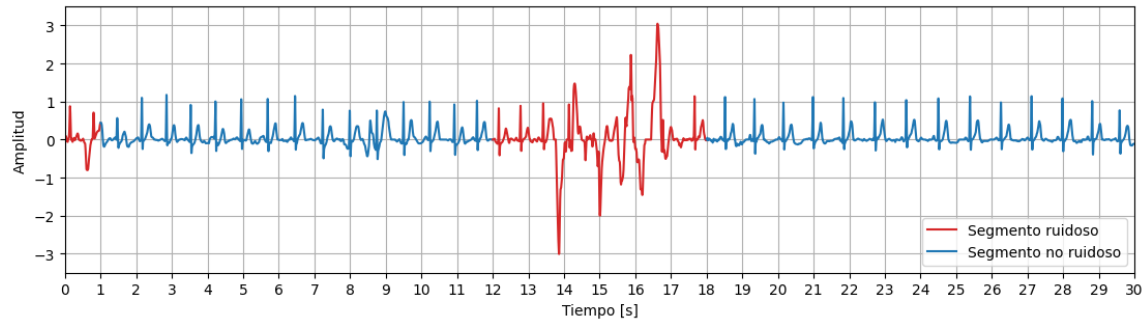


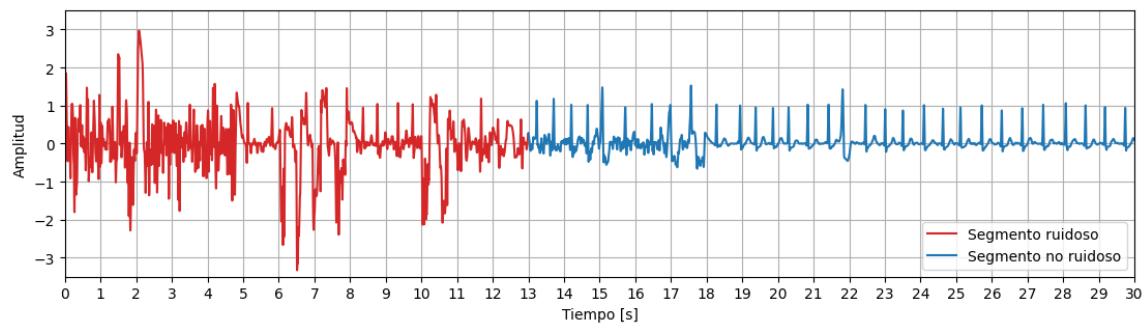
Figura 5.3. Resultados del rendimiento del clasificador de ruido KNN sobre el conjunto de validación. a) Matriz de confusión. b) Curva ROC, con ROC-AUC score de 0.98.

5.1.3 Clasificación de ruido de una señal completa

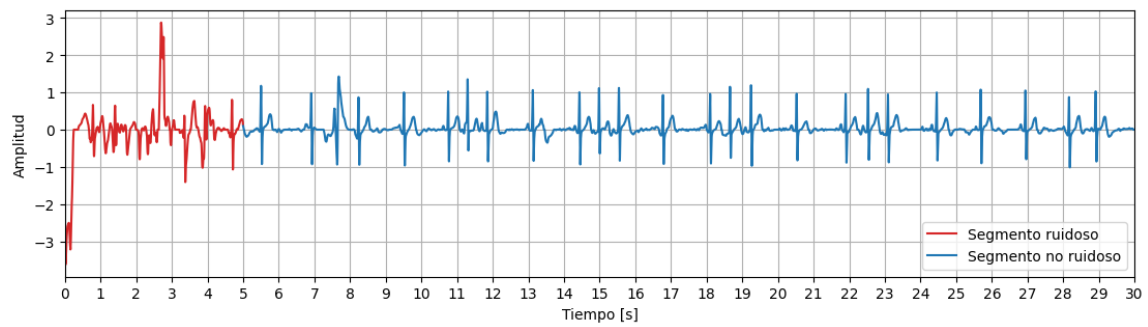
En la Figura 5.4 se ilustran 3 ejemplos de señales de ritmos distintos con clasificación de ruido, donde en rojo se ven los segmentos encontrados como ruidosos y en azul los no ruidosos.



(a)



(b)



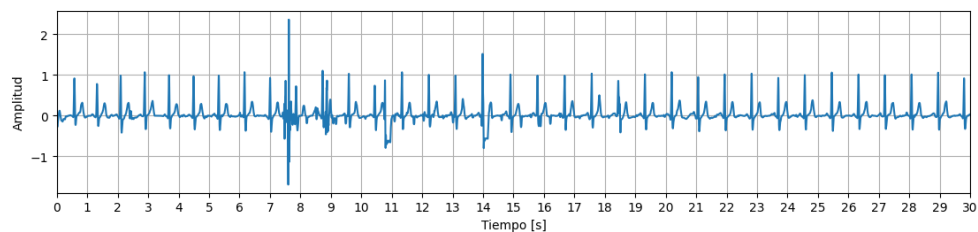
(c)

Figura 5.4. Ejemplo de señales clasificadas en segmentos ruidosos (rojo) y no ruidosos (azul).

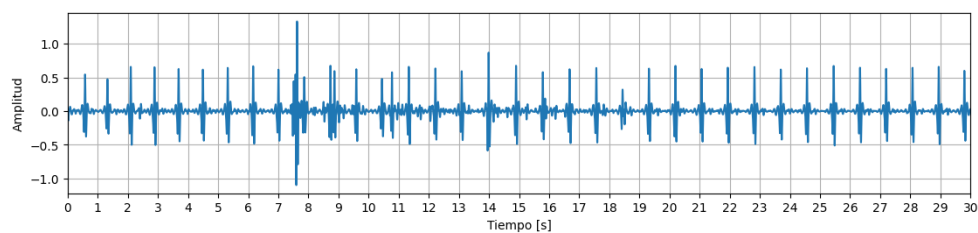
a) Señal nº 66, ritmo N. b) Señal nº 651, ritmo A. c) Señal nº 387, ritmo O.

5.2 Detección de complejos QRS

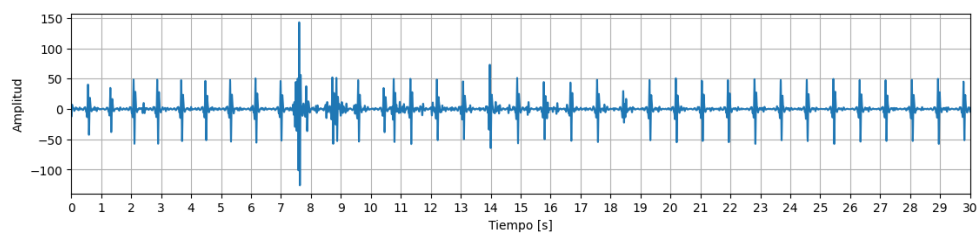
En la Figura 5.5 se muestra un ejemplo de señal con los pasos del algoritmo de Pan-Tompkins.



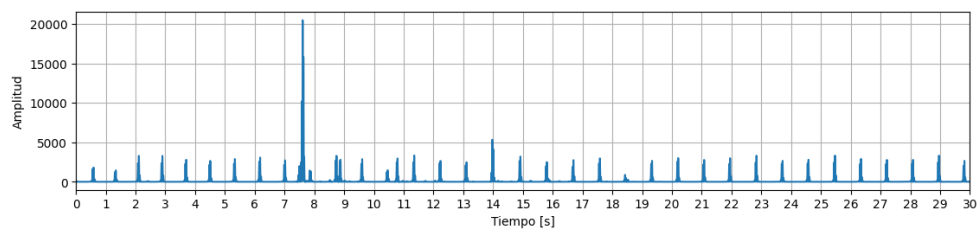
(a)



(b)

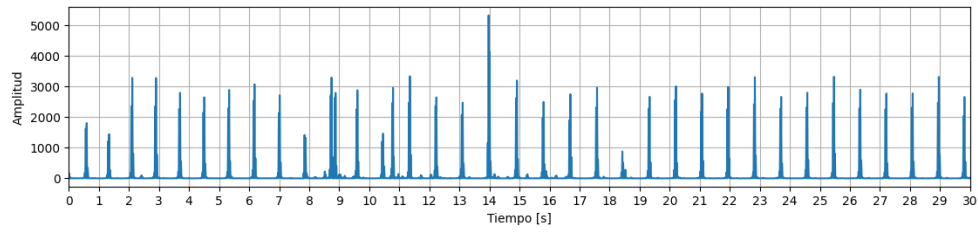


(c)

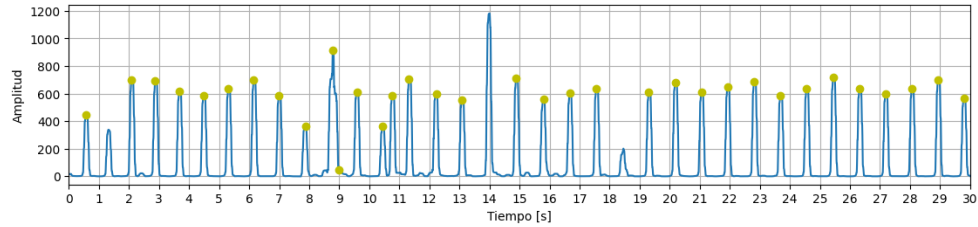


(d)

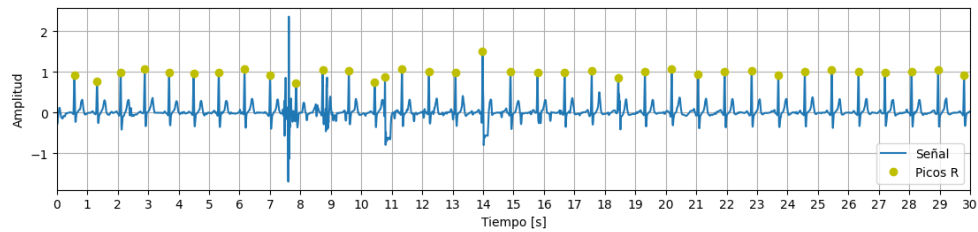
Figura 5.5. Pasos del algoritmo de Pan-Tompkins de implementación propia. Señal n° 1986: a) original, b) luego del filtro pasabanda, c) luego de la derivación, y d) luego de elevarla al cuadrado.



(e)



(f)



(g)

Figura 5.5. Continuación de los pasos del algoritmo de Pan-Tompkins de implementación propia. Señal n° 1986: e) luego del filtrado de artefactos, f) luego del la integración y con la detección de complejos QRS (verde), y g) señal original con la detección de complejos QRS

Los resultados obtenidos para la validación del detector de complejos QRS de implementación propia utilizando la base de datos de MIT fueron: exactitud del 96.82%, precisión del 99.37%, sensibilidad del 97.35% y F1-score del 98.30%.

5.3 Detección de latidos ectópicos

5.3.1 Índices de ectópicos

En la Figura 5.6 se muestran los histogramas de los índices de latidos ectópicos (rojo) y no ectópicos (azul).

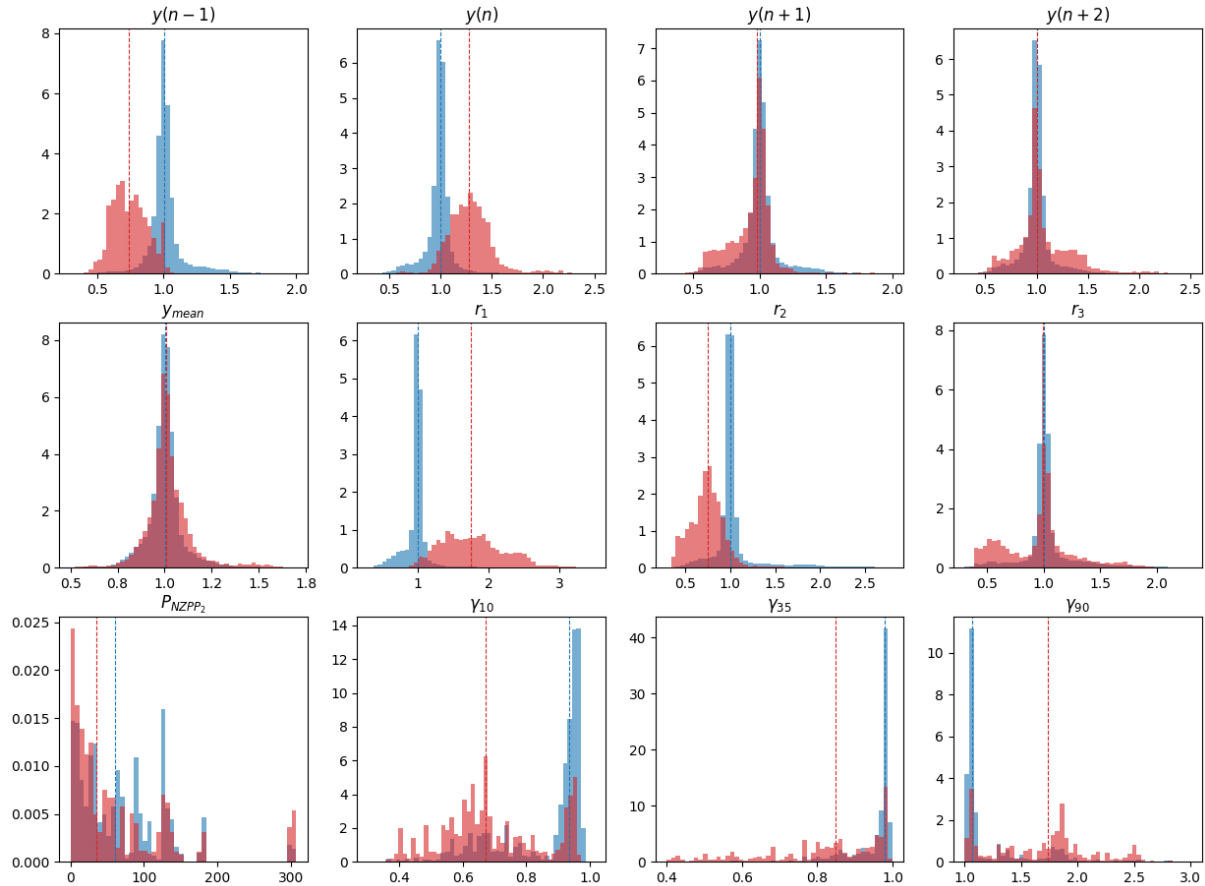


Figura 5.6. Histograma de los índices de ectópicos para las clases ectópicos (rojo) y no ectópicos (azul). La líneas punteadas muestran la mediana de cada clase.

Los histogramas que mejor separación tienen parecen ser $y(n-1)$, $y(n)$, r_1 y r_2 ; presentan una forma definida y distinta para los latidos ectópicos en comparación con los no ectópicos. En los latidos no ectópicos la mediana se sitúa en 1, mientras que en los ectópicos la mediana se aleja de este valor, siendo menor en el caso de $y(n-1)$ y r_2 , y mayor para $y(n)$ y r_1 . Esto hace sentido ya que estos atributos hacen referencia a los dos intervalos RR afectados por un

latido ectópico. En particular, $y(n - 1)$ y r_2 representan el intervalo corto anterior, y $y(n)$ y r_1 al intervalo largo posterior, característicos de un ectópico.

Otros índices que también parecen presentar una marcada separación de clases son γ_{10} , γ_{35} y γ_{90} , en donde los valores se acumulan cerca de la unidad en el caso de los latidos no ectópicos. En los latidos ectópicos, estos índices presentan valores distribuidos en un rango mayor. Esto indica que las series sin ectópicos son más regulares.

Por otro lado, los histogramas de $y(n + 1)$, $y(n + 2)$, y_{mean} , y r_3 muestran alto grado de superposición, y las medianas de ambas clases se ubican en el valor de la unidad. Esto sugiere que estos índices son poco útiles para la clasificación de ectópicos. Sin embargo, como resultan importantes desde un punto de vista teórico, se decidió incluirlos en los análisis posteriores donde su utilidad será puesta en evidencia mediante el desempeño del clasificador.

Finalmente, P_{NZPP_2} también muestra un alto grado de superposición, lo que era esperable ya que este no es un indicador de ectópicos directo, sino que se incluye con el fin de representar la irregularidad de la serie RR a la que pertenece el latido para reducir las falsas detecciones de ectópicos en ritmos altamente irregulares. Al igual que para los 4 índices anteriores, la importancia del mismo se verá reflejada en los siguientes análisis.

En la Figura 5.7 se muestra la matriz de las correlaciones de Kendall-Tau de todos los atributos utilizados para la detección de ectópicos.

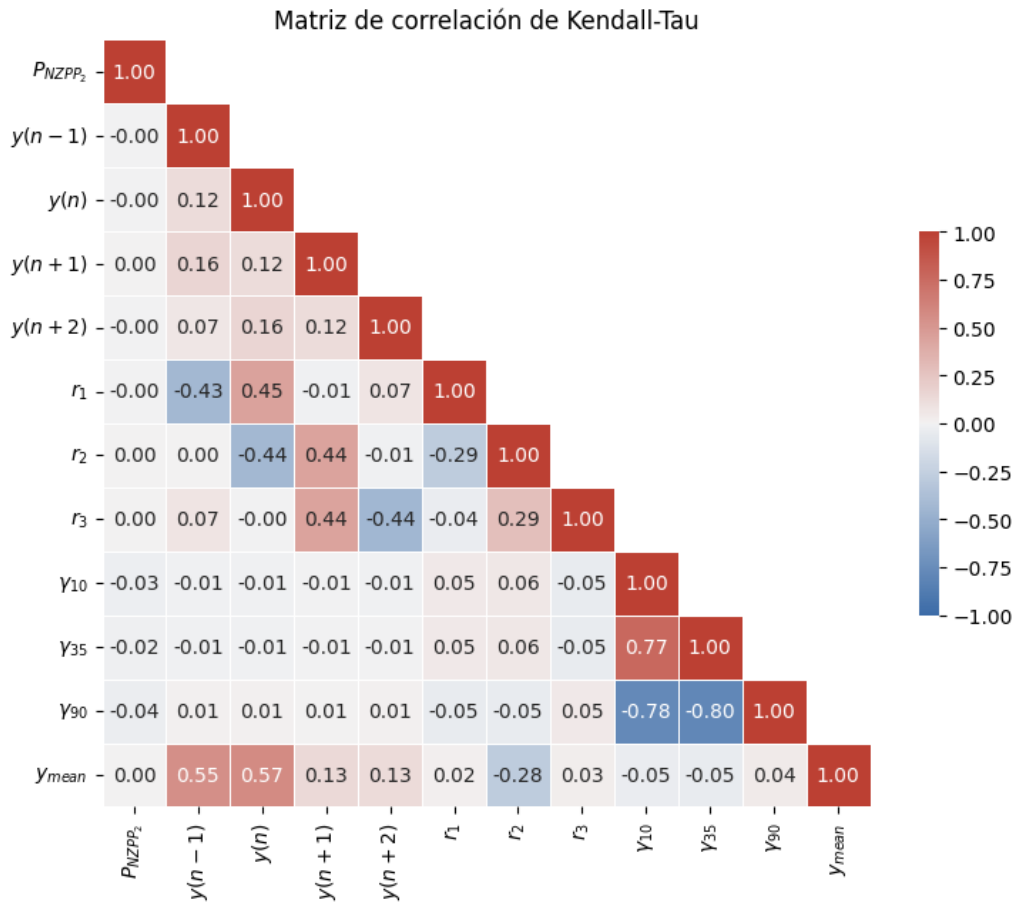


Figura 5.7. Matriz de correlación de Kendall-Tau entre los índices utilizados para la detección de ectópicos

Los resultados de la figura anterior muestran que γ_{10} , γ_{35} y γ_{90} tienen correlaciones muy altas, lo que era esperable dado que son los percentiles de la serie de razones RR y suelen variar de manera similar. Luego, el único valor medianamente elevado de correlación fue y_{mean} con $y(n)$ e $y(n-1)$, que mostró una correlación de 0.57 y 0.55 respectivamente. Esto es lógico dado que y_{mean} es el promedio entre los dos con los que se correlaciona.

Al ser tan elevados los valores de correlación entre los 3 percentiles, se realizó PCA sobre estos, y dio que el numero de componentes que mantenían al menos el 90% de la varianza eran 2. Estos se denominaron PCA_1 y PCA_2 .

La matriz de correlación sin los percentiles y con los dos nuevos componentes principales se muestra en la Figura 5.8. En esta se puede verse que PCA_1 y PCA_2 no se correlacionan entre

sí. En particular, PCA_1 presenta una correlación moderada con $y(n-1)$ e $y(n)$, que no existía en el caso de los percentiles.

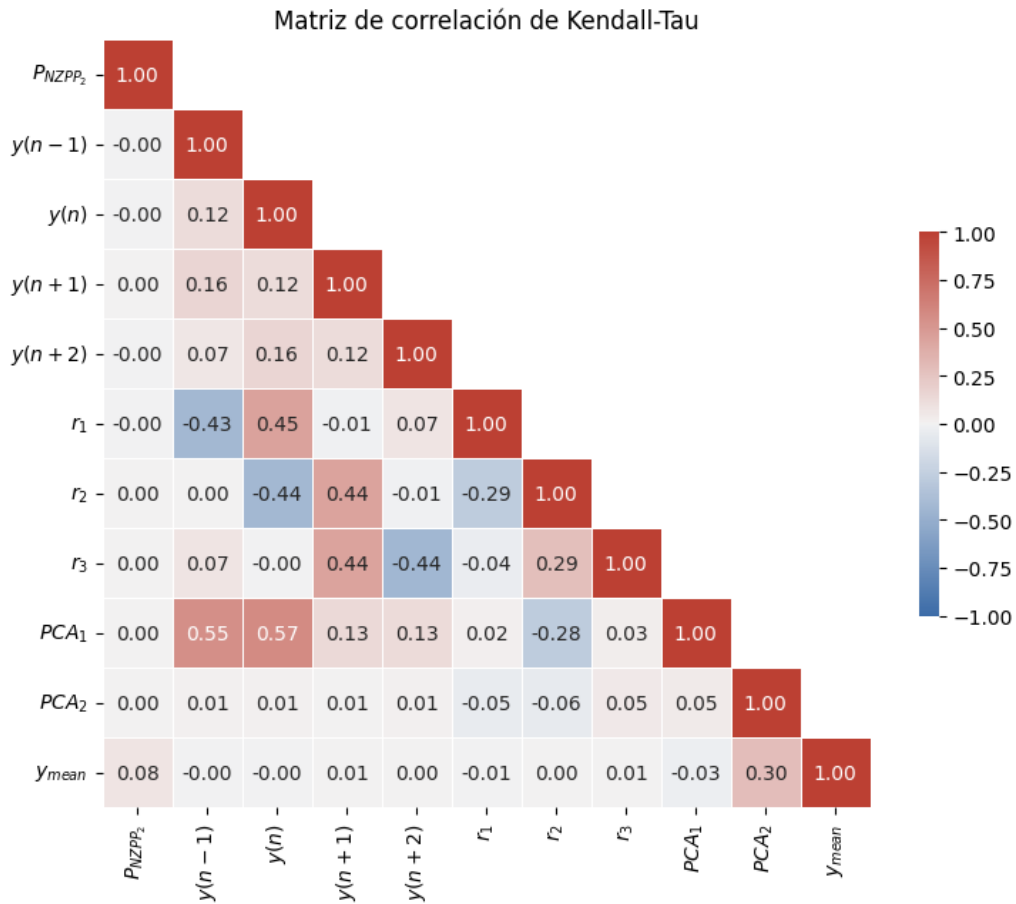


Figura 5.8. Matriz de correlación de Kendall-Tau entre los índices utilizados para la detección de ectópicos con PCA

5.3.2 Optimización de índices, hiperparámetros y clasificador de latidos

Finalmente se buscó la combinación óptima de hiperparámetros para los 5 clasificadores, con y sin la utilización de PCA para los atributos de entrada. Los modelos que mostraron mejor rendimiento y se consideraron como candidatos para cada tipo de clasificador pueden verse en la Tabla 5.2.

| Clasificador | Atributos | Hiperparámetros | Precisión | Sensibilidad | Especificidad | F1-Score |
|--------------|--------------|---|---------------|---------------|----------------|---------------|
| SVM | Todos | C = 10 gamma = 0.1 kernel = 'rbf' class_weight = None | 88.13 ± 1.48% | 81.49 ± 0.73% | 99.23 ± 0.095% | 84.67 ± 0.78% |
| KNN | Sin γ | k = 10 weights = 'distance' p = 1 | 88.77 ± 1.56% | 82.40 ± 0.94% | 99.27 ± 0.10% | 85.45 ± 0.67% |
| RL | Con PCA | penalty = 'l1' solver = 'liblinear' C = 1 class_weight = None | 77.99 ± 1.77% | 69.34 ± 2.65% | 98.62 ± 0.15% | 73.36 ± 1.36% |
| XGBoost | Todos | learning_rate = 0.1 max_depth = 7 n_estimators = 300 scale_pos_weight = None | 89.35 ± 1.41% | 84.41 ± 1.83% | 99.29 ± 0.11 | 86.78 ± 0.82% |
| AdaBoost | Todos | learning_rate = 0.1 max_depth = 3 n_estimators = 200 | 87.76 ± 1.91% | 82.40 ± 0.94% | 99.18 ± 0.14% | 85.62 ± 0.98% |

Tabla 5.2. Mejores resultados para cada clasificador de latidos. En la columna ‘Atributos’: ‘Todos’ indica que se utilizaron los atributos inicialmente comentados, ‘Sin γ ’ significa que se removieron los percentiles y ‘Con PCA’ implica que se reemplazaron estos por sus 2 componentes principales. Las métricas de evaluación fueron obtenidas mediante validación cruzada.

Como puede verse en la Tabla, entre los modelos de XGBoost se eligió uno que no tiene atributos PCA y si tiene los 3 percentiles altamente correlacionados, ya que fue la versión que demostró mejor desempeño. Utilizar los percentiles en este caso no representa un problema ya que XGBoost, al ser un modelo basado en arboles de decisión, es robusto a los datos altamente correlacionados [46].

Como se explicó previamente, si bien priorizar la precisión en este caso es muy importante, no se deben descuidar tampoco el F1-score y la sensibilidad. De todas formas, el clasificador que se destacó por presentar mejor precisión, sensibilidad, F1-score y especificidad es XGBoost. Por esta razón, se eligió el clasificador XGBoost.

En la Figura 5.9 se muestra el gráfico de la importancia adjudicada por el clasificador de XGBoost a cada atributo de detección de latidos ectópicos.

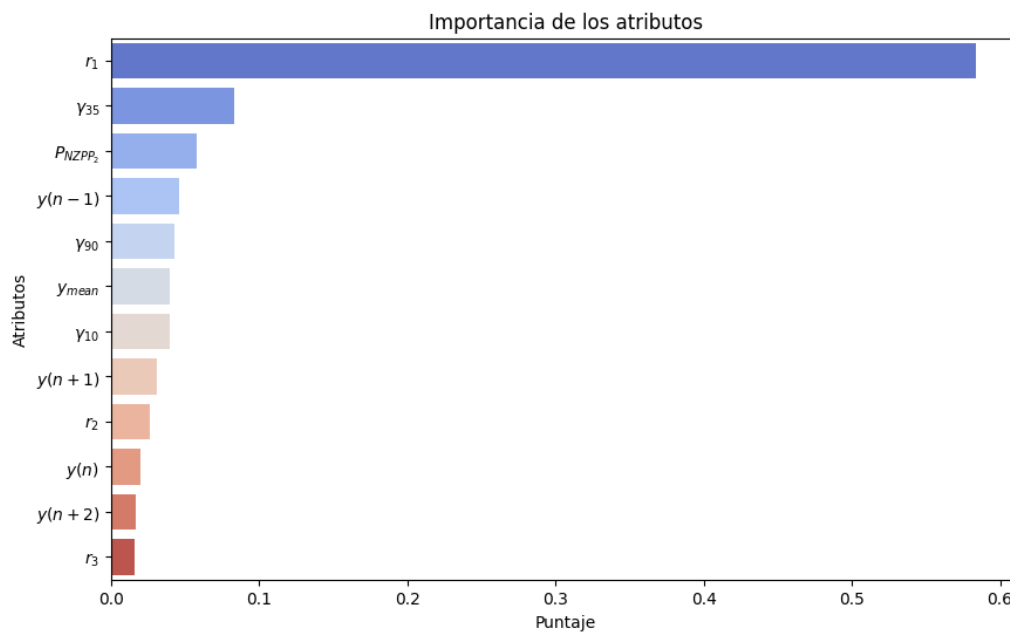


Figura 5.9. Gráfico de la importancia relativa de cada atributo

El atributo que presenta mayor importancia resulta ser r_1 . Por el contrario, $y(n)$ parece tener muy baja importancia, cuando se sabe que es un atributo relevante desde el punto de vista teórico. Sin embargo, es probable que esto se deba a que $y(n)$ comparte información con r_1 , ya que este último es el cociente de $y(n)$ e $y(n-1)$. A su vez, esto confirma que XGBoost opera distribuyendo el peso entre variables relacionadas de forma acertada, lo que sugiere que incluir

los 3 percentiles con alta correlación no representa un problema.

Contando con la importancia de cada atributo se implementó un análisis adicional en el que se removieron los atributos uno a uno en orden de importancia creciente. Se comenzó eliminando r_3 y $y(n+2)$, y se encontró que el rendimiento general del clasificador mejoró cuando se eliminó r_3 pero empeoró cuando se removió $y(n+2)$ o la combinación de estos dos. En particular, las métricas de validación cruzada sin r_3 fueron: precisión del 89.48%, sensibilidad del 84.70%, especificidad del 99.30% y F1-score del 87.01%. Por lo tanto, los atributos elegidos finalmente fueron todos menos r_3 .

En resumen, el clasificador elegido para la detección de latidos ectópicos fue XGBoost, con un learning rate de 0.1 y 300 árboles con profundidad máxima de 7, utilizando todos los atributos exceptuando r_3 , y sin considerar el peso relativo de las clases.

Las métricas de evaluación reportadas para este clasificador, utilizando el 20% de los datos que se habían apartado como conjunto de validación, fueron: precisión del 92.55%, sensibilidad del 88.23%, especificidad del 99.51% y F1-score del 90.34%. La matriz de confusión y la curva ROC de esta validación se muestran en la Figura 5.10.

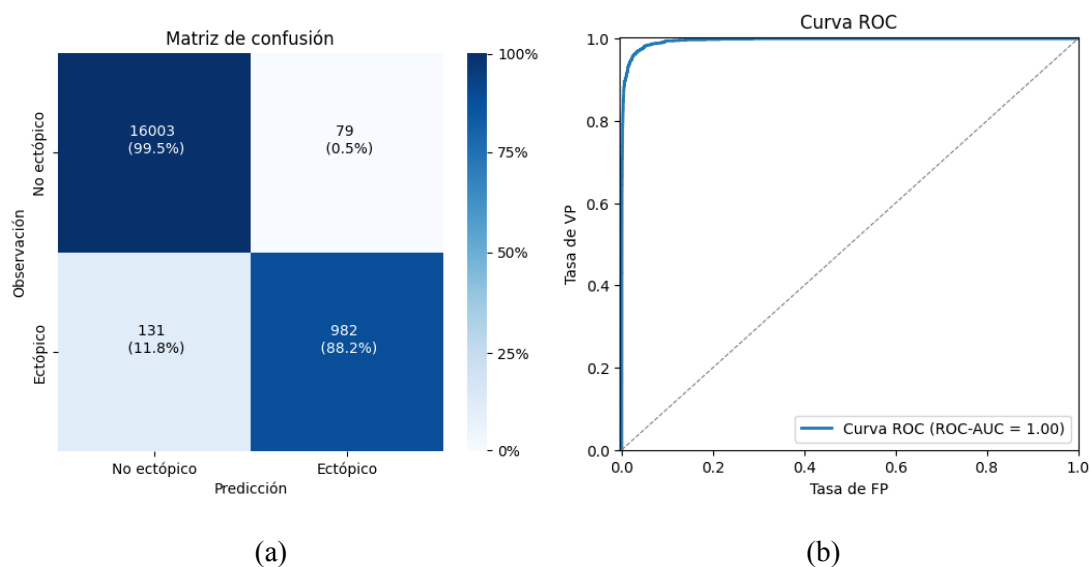
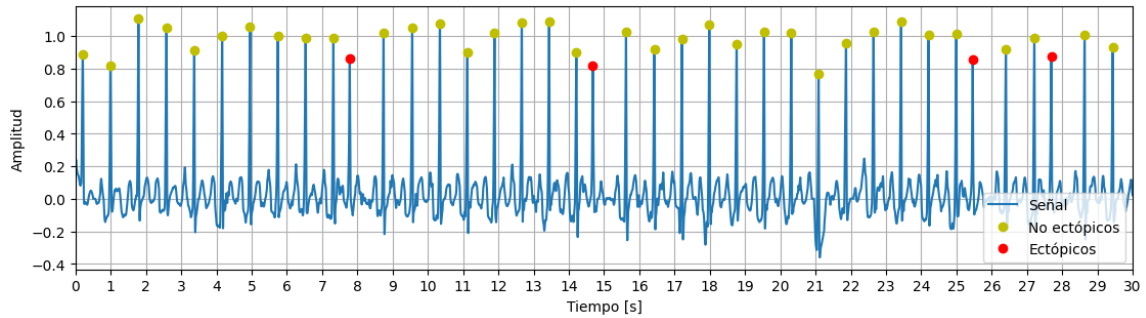


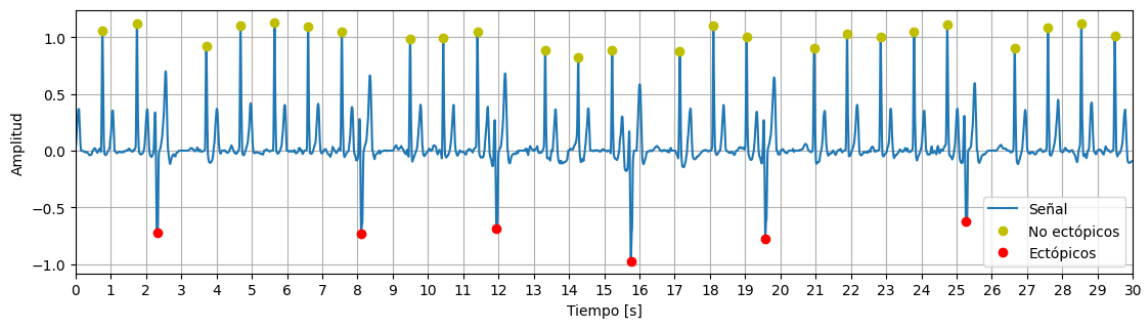
Figura 5.10. Resultados del rendimiento del clasificador de ruido XGBoost ganador sobre el conjunto de validación. a) Matriz de confusión. b) Curva ROC.

5.3.3 Clasificación de latidos de señales ejemplo

En la Figura 5.11 se ilustran 2 ejemplos de señales con detección de ectópicos, donde en verde se ven los latidos no ectópicos y en rojo los ectópicos, encontrados mediante el clasificador de implementación propia.



(a)



(b)

Figura 5.11. Ejemplo de señales con clasificación de latidos ectópicos (rojo) y no ectópicos (verde). a) Señal nº 6069, ritmo O. b) Señal nº 3334, ritmo O.

5.4 Detección de FA

5.4.1 Índices de FA

En la Figura 5.12 se muestran los histogramas de todos los índices de FA provenientes de señales de clase FA (rojo) y no FA (azul). Se recuerda que la clase no FA esta compuesta no solo por señales de ritmo de sinusal sino también por otro tipo de arritmias (ritmo O). Las líneas punteadas representan la mediana de cada clase. Luego, en la Figura 5.13 se muestra la matriz de correlación de los atributos de FA.

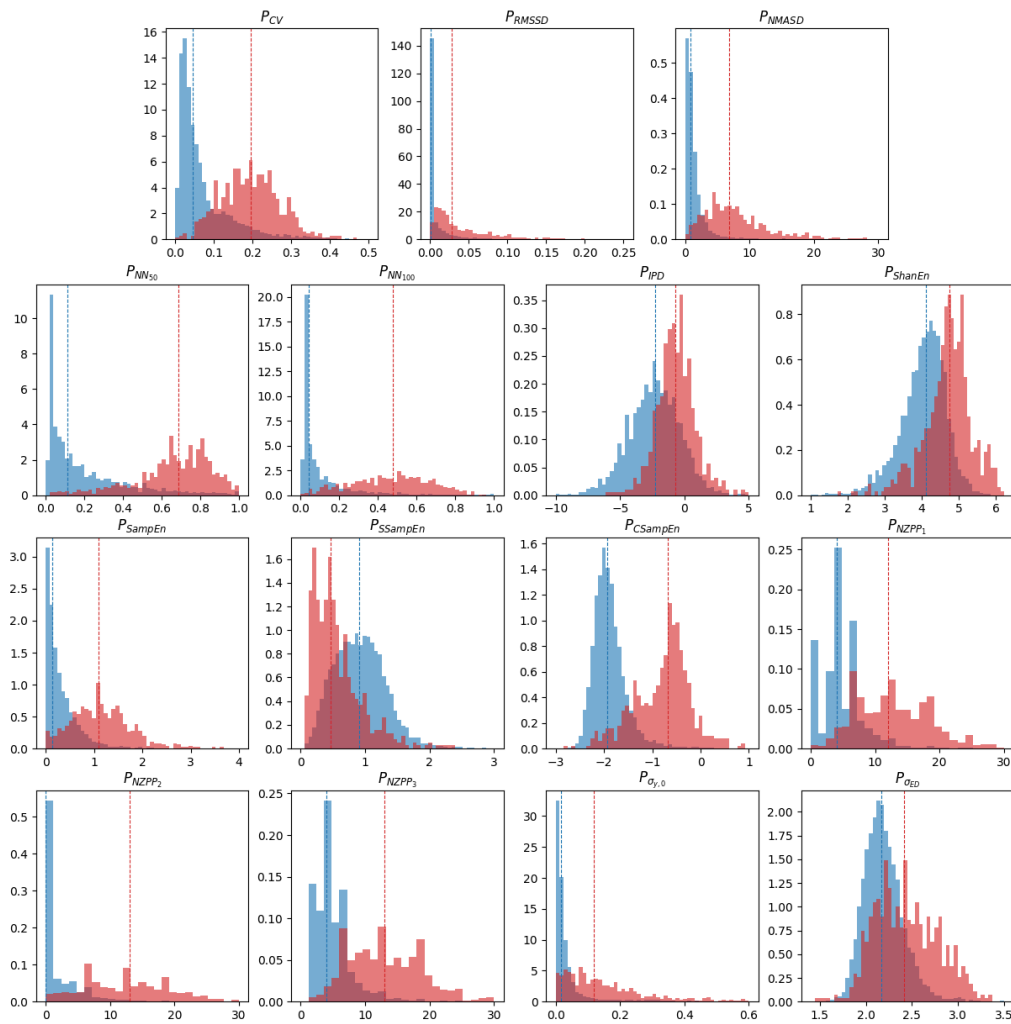


Figura 5.12. Histograma de los índices de FA de implementación propia, para las clases FA (rojo) y no FA (azul). La líneas punteadas muestran la mediana de cada clase.

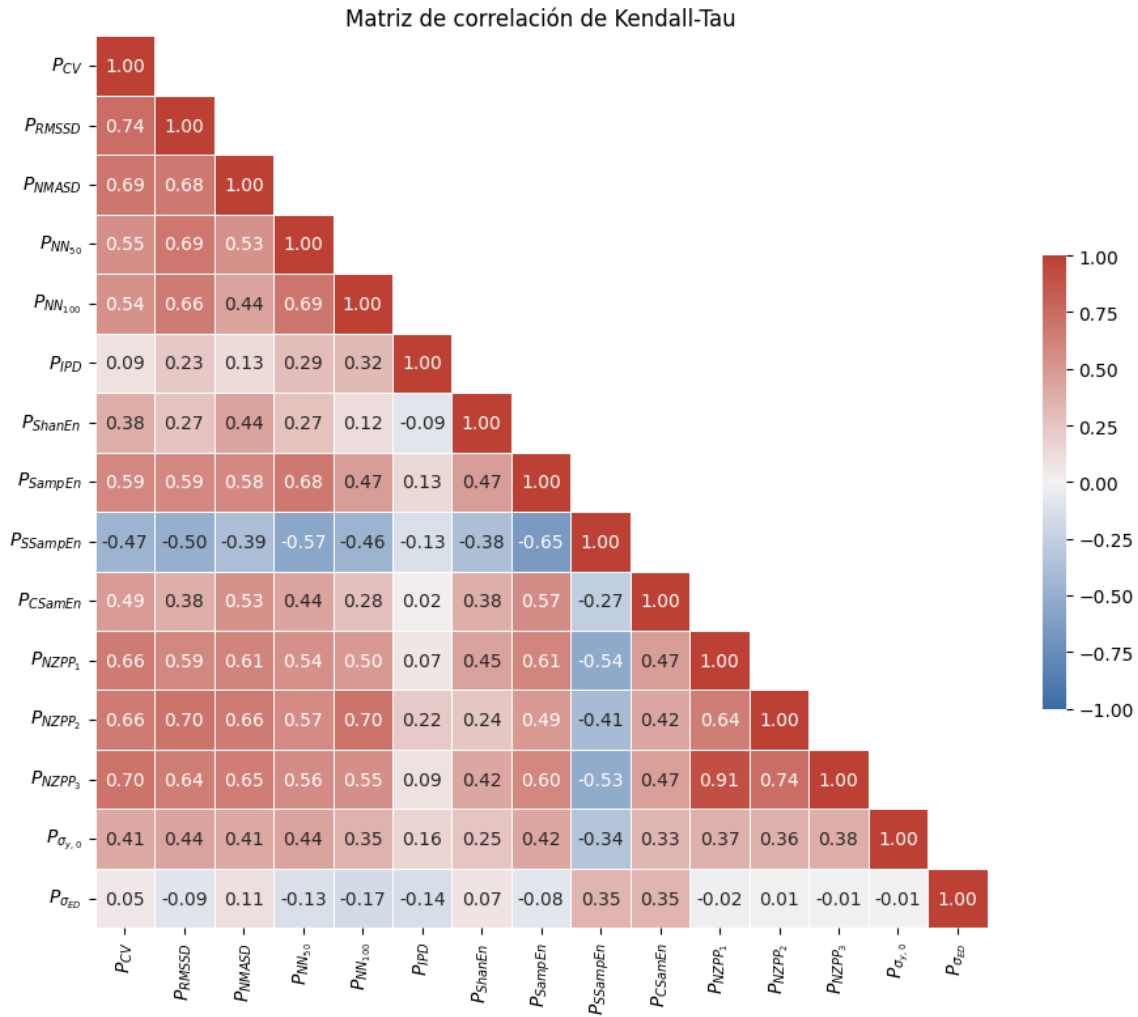


Figura 5.13. Matriz de correlación de Kendall-Tau de todos los índices de FA

P_{CV} , P_{RMSSD} y P_{NMASD} presentan una buena separación y una distribución similar por clase. Esto es coherente, ya que los tres son índices basados en dispersión estadística que, debido a su método de cálculo, analizan información muy parecida. Por la misma razón, es coherente observar los altos grados de correlación entre estos tres atributos, según se observa en la matriz de Kendall-Tau, con valores todos superiores a 0.65.

En el caso de $P_{NN_{50}}$ y $P_{NN_{100}}$, los otros dos parámetros de dispersión estadística, ambos presentan distribuciones similares, pero con desplazamientos en las medianas. En particular, para $P_{NN_{100}}$ las medianas están desplazadas hacia la izquierda (valores más bajos) en ambas clases. Esta observación es coherente, considerando que la única diferencia entre $P_{NN_{50}}$ y $P_{NN_{100}}$ es un cambio en el umbral de tolerancia usado para determinar si dos intervalos

sucesivos son distintos. Además, es esperado que para registros sin FA, el valor de ambos índices se aproxime a 0, mientras que en registros con FA, estos valores aumenten, reflejando la variabilidad típica de la arritmia con un mayor porcentaje de intervalos NN sucesivos diferentes. Dada la similitud de estos índices, no sorprende que muestren una fuerte correlación de 0.69. P_{RMSSD} , al basarse también en la diferencia entre intervalos sucesivos, naturalmente presenta correlaciones altas con los dos índices mencionados. Lo mismo aplica a P_{CV} y $P_{NMA SD}$, que aunque no lleguen al límite establecido de 0.6 para ser considerada una correlación fuerte, tampoco se alejan mucho de este.

P_{IPD} , un parámetro fundamentado en los puntos de inflexión, evidencia una notable superposición en los histogramas para cada clase, lo que podría sugerir que es menos útil como indicador de FA. No obstante, esta superposición no es total, identificándose claramente dos picos distintos en el histograma de cada clase. Además, la matriz de correlaciones indica que tiene valores bajos de correlación con la mayoría de los atributos, sugiriendo que podría seguir siendo de interés.

En cuanto a P_{ShanEn} , un atributo basado en la entropía de Shannon, su comportamiento es parecido al de P_{IPD} . También presenta una superposición significativa entre los histogramas de cada clase, pero no completa. Además, al examinar la matriz de correlaciones, se aprecia que posee valores bajos de correlación con la mayoría de los atributos. Esto indica que podría seguir siendo relevante ya que parece proporcionar información distinta a los otros índices.

Para los atributos de entropía de muestreo (P_{SampEn} , $P_{SSampEn}$ y $P_{CSampEn}$), los histogramas se perciben notablemente distintos. De estos, $P_{SSampEn}$ parece mostrar la menor separación. Al inspeccionar las correlaciones, $P_{SSampEn}$ y P_{SampEn} presentan una correlación alta (-0.65). Por otro lado, $P_{CSampEn}$ y P_{SampEn} tienen un valor de correlación de 0.57, por debajo del límite establecido para considerarla alta, lo cual es sorprendente ya que uno es una modificación del otro (ver sección 4.5.1). También se puede notar una correlación alta negativa de P_{SampEn} con $P_{NN_{50}}$.

Los parámetros basados en el gráfico de Poincaré $P_{NZPP_{1,2,3}}$ muestran distribuciones similares y con una remarcada separación de clases, donde P_{NZPP_2} parece ser el más distinto y el

que mejor separa los ritmos. Esto se condice con los resultados de las correlaciones, donde los 3 atributos muestran entre sí valores elevados pero con un valor muy alto entre P_{NZPP_1} y P_{NZPP_3} (0.91), mientras que bastante mas inferior (aunque aún alto) cuando se observa la correlación de estos dos con P_{NZPP_2} (0.64 y 0.74 respectivamente). En cuanto a correlaciones con los otros atributos, las altas son P_{NZPP_2} con $P_{NN_{50}}$ (0.70) y $P_{NZPP_{1,2,3}}$ con P_{CV} , P_{RMSSD} y $P_{NMA SD}$, con valores superiores a 0.6 para todas estas combinaciones exceptuando P_{NZPP_1} con P_{RMSSD} (0.59), quedando levemente por debajo del umbral establecido en 0.6.

Finalmente, $P_{\sigma,0}$ muestra una buena separación de clases mostrando también valores relativamente bajos de correlación con las demás variables, por lo que es un índice prometedor no solo para discriminar bien los registros con FA sino también porque pareciera agregar información única. El caso de $P_{\sigma_{ED}}$ es similar al de P_{IPD} y P_{ShanEn} , donde el histograma muestra una elevada superposición de clases pero las correlaciones con los otros atributos dan muy bajas, por lo que podría de todos modos junto a otros atributos ser de utilidad.

Dado el análisis de histogramas y correlaciones, todos los atributos parecen contener información pertinente. Por ello, se optó por incluirlos todos en el método *wrapper* de selección de atributos e hiperparámetros para los cinco clasificadores seleccionados, evitando combinaciones con atributos altamente correlacionados. Se evitaron las combinaciones con atributos altamente correlacionados (con valores absolutos de correlación mayores a 0.6), lo que lleva a entrenar modelos con un máximo de 9 atributos. Por ejemplo, un modelo no podría utilizar P_{CV} y $P_{NMA SD}$ simultáneamente, limitándose a incluir como máximo uno de estos a la vez.

5.4.2 Optimización de índices, hiperparámetros y clasificador de FA

Modelos entrenados con conjunto de datos balanceado

Los resultados de la mejor combinación de atributos e hiperparámetros para cada uno de los 5 clasificadores probados entrenando con un dataset balanceado puede verse en la Tabla 5.3. En estas tablas también se muestra el rendimiento de dichos clasificadores en la base de datos de Kaggle (K) y la sintética (S).

| Clasificador | Atributos | Hiperparámetros | Exactitud | Precisión | Sensibilidad | Especificidad | F1-Score |
|--------------|--|---|--|--|--|--|--|
| SVM | $P_{CV}, P_{NN100}, P_{IPD}, P_{ShanEn}, P_{SSamEn}, P_{CSamEn}, P_{NZPP1}, P_{\sigma_{y,0}}, P_{\sigma_{ED}}$ | C = 100 gamma = 0.001 kernel = 'sigmoid' | 91.22 ± 3.14% V: 92.28% K: 87.07% S: 84.70% | 92.94 ± 5.23% V: 52.51% K: 80.16% S: 97.09% | 89.59 ± 4.43% V: 89.84% K: 62.52% S: 71.33% | 92.84 ± 5.42% V: 92.51% K: 95.00% S: 97.89% | 91.09 ± 3.11% V: 66.28% K: 70.25% S: 82.24% |
| KNN | $P_{CV}, P_{NN100}, P_{IPD}, P_{CSamEn}, P_{NZPP1}, P_{\sigma_{ED}}$ | k = 4 weights = 'uniform' p = 1 | 90.44 ± 2.23% V: 92.68% K: 88.34% S: 86.08% | 93.75 ± 4.00% V: 54.15% K: 73.20% S: 80.26% | 86.88 ± 3.06% V: 86.72% K: 82.46% S: 95.44% | 94.00 ± 3.94% V: 93.23% K: 90.24% S: 76.85% | 90.10 ± 2.23% V: 66.67% K: 77.55% S: 87.19% |
| RL | $P_{CV}, P_{NN50}, P_{ShanEn}, P_{CSamEn}, P_{NZPP2}, P_{\sigma_{y,0}}, P_{\sigma_{ED}}$ | penalty = 'elasticnet' l1_ratio = 0.01 solver = 'saga' C = 1 | 90.93 ± 2.55% V: 92.35% K: 89.20% S: 91.95% | 92.38 ± 3.53% V: 52.73% K: 72.39% S: 87.07% | 89.40 ± 4.24% V: 90.62% K: 90.17% S: 98.40% | 92.46 ± 3.69% V: 92.51% K: 88.89% S: 85.59% | 90.77 ± 2.63% V: 66.67% K: 80.31% S: 92.39% |
| XGBoost | $P_{CV}, P_{NN50}, P_{ShanEn}, P_{SSamEn}, P_{CSamEn}, P_{NZPP2}, P_{\sigma_{y,0}}, P_{\sigma_{ED}}$ | learning_rate = 0.1 max_depth = 5 n_estimators = 200 | 90.54 ± 1.13% V: 90.57% K: 89.13% S: 86.41% | 90.82 ± 2.60% V: 46.86% K: 73.53% S: 79.97% | 90.35 ± 2.70% V: 87.50% K: 86.66% S: 96.90% | 90.73 ± 2.98% V: 90.85% K: 90.24% S: 76.85% | 90.52 ± 1.10% V: 61.04% K: 79.57% S: 87.62% |
| AdaBoost | $P_{NMASD}, P_{NN50}, P_{IPD}, P_{SSamEn}, P_{CSamEn}, P_{NZPP2}, P_{\sigma_{y,0}}, P_{\sigma_{ED}}$ | learning_rate = 1 max_depth = 5 n_estimators = 150 | 91.02 ± 2.48% V: 90.96% K: 91.20% S: 89.61% | 91.60 ± 4.64% V: 48.13% K: 80.67% S: 84.47% | 90.55 ± 3.51% V: 90.62% K: 84.11% S: 96.9% | 91.50 ± 5.10% V: 90.99% K: 93.49% S: 82.43% | 90.98 ± 2.34% V: 62.87% K: 82.36% S: 90.26% |

Tabla 5.3. Mejores resultados para cada clasificador de FA con datos de entrenamiento balanceados. El primer valor de cada métrica de evaluación corresponde al obtenido por validación cruzada en el conjunto de entrenamiento de SSLAFDB. V: Conjunto de validación, K: Base de datos de Kaggle, y S: Base de datos sintética.

La Tabla 5.3 muestra que todas las métricas obtenidas con validación cruzada en el conjunto de entrenamiento balanceado son altas, oscilando entre 86 y 94%. Aun así, al observar las métricas en el conjunto de validación, tanto la precisión como el F1-score disminuyen significativamente. Este descenso era esperado debido a que el conjunto de entrenamiento se encuentra balanceado mientras que el de validación está altamente desbalanceado.. A pesar de que el algoritmo puede presentar un bajo porcentaje de falsos positivos, el alto predominio de señales de ritmo no FA en el conjunto de validación implica que el número de falsos positivos sea elevado.

El rendimiento de cada clasificador indica que uno de los más eficientes en validación cruzada parece ser SVM, con una exactitud y F1-score cercanos al 91%. Sin embargo, estas métricas tienen las desviaciones estándar más altas (3.11 - 5.42%), lo que indica una menor

fiabilidad. Asimismo, estas métricas disminuyen considerablemente en las bases de datos de Kaggle y sintéticas, lo que sugiere que el clasificador es poco robusto.

KNN es otro clasificador que presenta un buen rendimiento, destacándose por su precisión y especificidad rondando el 94% en validación cruzada. A pesar de estos números fuertes, su sensibilidad queda en torno al 87% tanto en la validación cruzada como en el conjunto de validación. Esta es la cifra de sensibilidad más baja en comparación con otros clasificadores en estos conjuntos de datos, y disminuye aún más en la base de datos de Kaggle.

XGBoost y AdaBoost obtienen métricas altas y consistentes mediante validación cruzada, aproximadamente del 91% en ambos casos. Sin embargo, en el conjunto de validación, sus precisiones son del 46.86% y 48.13%, respectivamente, los dos valores más bajos de la tabla, lo cual resulta inaceptable. Además, el rendimiento de XGBoost disminuye notablemente en la base de datos sintética.

Por último, la regresión logística arroja métricas altas y bastante equilibradas en validación cruzada, aunque no superiores a las de los otros clasificadores. Sin embargo, se demuestra robusta en los distintos conjuntos y bases de datos, especialmente por su F1-score del 66.67% en el conjunto de validación. Su precisión en este mismo conjunto no es alta, pero representa la segunda más elevada de la tabla, con un valor del 52.73%. Dado el rendimiento robusto y comportamiento consistente, se optó por seleccionar la regresión logística como el clasificador preferido entre los modelos entrenados con datos balanceados. La matriz de confusión y la curva ROC del conjunto de validación de este clasificador de FA se muestran en la Figura 5.14.

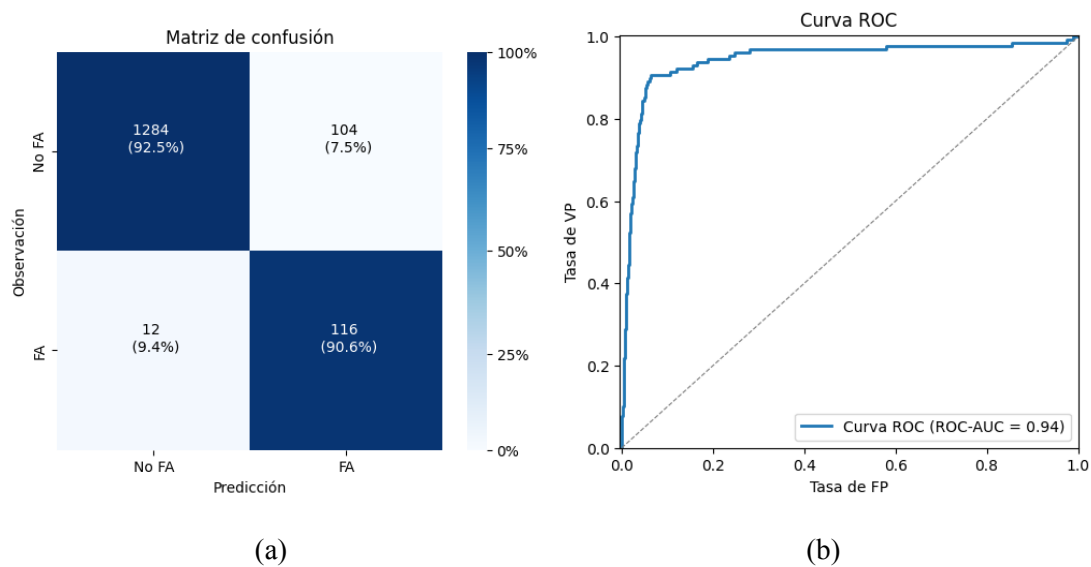


Figura 5.14. Resultados del rendimiento del clasificador balanceado de FA RL ganador sobre el conjunto de validación. a) Matriz de confusión. b) Curva ROC.

Modelos entrenados con conjunto de datos no balanceado

Los resultados de la combinación óptima de atributos e hiperparámetros para cada uno de los 5 clasificadores probados entrenando con un dataset desbalanceado se presentan en la Tabla 5.4.

| Clasificador | Atributos | Hiperparámetros | Exactitud | Precisión | Sensibilidad | Especificidad | F1-Score |
|--------------|--|--|--|--|--|--|--|
| SVM | P_{CV} , P_{NN100} , P_{ShanEn} , P_{SamEn} , P_{CSamEn} , P_{NZPP1} , $P_{\sigma_{y,0}}$, $P_{\sigma_{ED}}$ | C = 10 gamma = 0.1 kernel = 'rbf' | 96.21 ± 0.46% V: 95.84% K: 86.49% S: 88.92% | 81.80 ± 3.44% V: 77.78% K: 88.06% S: 96.71% | 71.62 ± 3.59% V: 71.09% K: 51.72% S: 80.42% | 98.50 ± 0.32% V: 98.13% K: 97.73% S: 97.30% | 76.32 ± 2.94% V: 74.29% K: 65.16% S: 87.81% |
| KNN | P_{CV} , P_{NN50} , P_{ShanEn} , P_{SSamEn} , P_{CSamEn} , P_{NZPP2} , $P_{\sigma_{ED}}$ | k = 16 weights = 'distance' p = 2 | 96.37 ± 0.54% V: 95.78% K: 85.84% S: 90.11% | 82.00 ± 3.52% V: 76.67% K: 91.45% S: 97.63% | 73.73 ± 4.86% V: 71.88% K: 46.35% S: 82.08% | 98.48 ± 0.31% V: 97.98% K: 98.60% S: 98.03% | 77.58 ± 3.74% V: 74.19% K: 61.52% S: 89.18% |
| RL | P_{RMSSD} , P_{IPD} , P_{ShanEn} , P_{CSamEn} , P_{NZPP1} , $P_{\sigma_{y,0}}$, $P_{\sigma_{ED}}$ | penalty = 'l1' solver = 'liblinear' C = 1 | 95.99 ± 0.42% V: 95.78% K: 86.24% S: 86.30% | 82.99 ± 4.01% V: 80.77% K: 91.61% S: 99.36% | 66.97 ± 3.88% V: 65.62% K: 48.07% S: 72.87% | 98.70 ± 0.38% V: 98.56% K: 98.58% S: 99.53% | 74.03 ± 2.95% V: 72.41% K: 63.06% S: 84.08% |
| XGBoost | P_{CV} , P_{NN50} , P_{ShanEn} , P_{SSamEn} , P_{CSamEn} , P_{NZPP2} | learning_rate = 0.1 max_depth = 3 n_estimators = 100 | 96.27 ± 0.18% V: 95.84% K: 87.52% S: 88.61% | 82.02 ± 2.51% V: 78.26% K: 92.07% S: 97.63% | 72.39 ± 2.82% V: 70.31% K: 53.51% S: 78.89% | 98.50 ± 0.31% V: 98.20% K: 98.51% S: 98.20% | 76.83 ± 1.19% V: 74.07% K: 67.68% S: 87.31% |
| AdaBoost | P_{CV} , P_{NN100} , P_{IPD} , P_{ShanEn} , P_{SamEn} , P_{CSamEn} , P_{NZPP1} , $P_{\sigma_{ED}}$ | learning_rate = 0.1 max_depth = 3 n_estimators = 50 | 96.24 ± 0.28% V: 95.58% K: 88.90% S: 89.51% | 83.09 ± 3.08% V: 76.07% K: 89.93% S: 97.20% | 70.46 ± 2.81% V: 69.53% K: 61.42% S: 81.22% | 98.65 ± 0.31% V: 97.98% K: 97.78% S: 97.69% | 76.19 ± 1.82% V: 72.65% K: 72.99% S: 88.49% |

Tabla 5.4. Mejores resultados para cada clasificador de FA con datos de entrenamiento no balanceados. El primer valor de cada métrica de evaluación corresponde al obtenido por validación cruzada en el conjunto de entrenamiento de SSLAFDB. V: conjunto de validación, K: base de datos de Kaggle, y S: base de datos sintética.

La Tabla 5.4 revela una tendencia general de los clasificadores entrenados con el conjunto de datos no balanceado, donde se logra una alta precisión y especificidad mediante validación cruzada, aproximándose al 96 y 98%, respectivamente. Sin embargo, la sensibilidad (y por ende, el F1-score) disminuye notablemente en comparación con la tabla anterior, ubicándose en un rango del 67 al 74% en validación cruzada. En la base de datos de Kaggle, la sensibilidad disminuye aún más, alcanzando valores entre 46 y 61%. Esto era previsible dado que en este caso, la clase FA está considerablemente menos representada que la otra, lo que conlleva a un incremento de falsos negativos.

Un efecto positivo de este desbalance de datos es la alta especificidad obtenida por todos los clasificadores en todos los conjuntos y bases de datos, fluctuando entre el 97 y 99%. Adicionalmente, se nota una alta consistencia entre las métricas logradas por validación cruzada y las del conjunto de validación, debido a que los conjuntos de entrenamiento y validación mantienen la misma desproporción de clases.

Al analizar el rendimiento de los distintos clasificadores, se observa que KNN parece ser el que mejores métricas presenta por validación cruzada, destacándose por su precisión del 96.37%, sensibilidad del 73.73% y F1-score del 77.58% en validación cruzada. No obstante, la desviación estándar de estas dos últimas métricas es alta, con valores de 4.86 y 3.74% respectivamente, lo que resta confiabilidad a estas métricas. Además, la sensibilidad de este clasificador en la base de datos de Kaggle es la métrica más baja de la tabla, con un valor de 46.35%, lo que sugiere que KNN no es robusto en este caso.

XGBoost, con la segunda mejor precisión, sensibilidad y F1-score en validación cruzada, logra valores de 96.27%, 72.39% y 76.83%, respectivamente. No obstante, la sensibilidad en la base de datos de Kaggle también cae significativamente, con un valor de 53.51%.

La regresión logística presenta métricas de sensibilidad bajas en todos los conjuntos, con valores de 65.62% en el conjunto de validación y 48.07% en la base de datos de Kaggle.

AdaBoost muestra un rendimiento bastante promedio en comparación con los otros clasificadores. Su distintivo radica en mantener métricas elevadas en la base de datos sintética y, en particular, en la de Kaggle, donde la mayoría de los clasificadores exhiben una baja sensibilidad. Este comportamiento robusto es un atributo destacable de este clasificador.

Al seleccionar un único modelo que se haya entrenado con datos no balanceados, se consideraron XGBoost y AdaBoost, ya que ambos parecen candidatos sólidos. Sin embargo, la elección final fue AdaBoost debido a su robustez en las tres bases de datos y su reconocida efectividad al manejar conjuntos de datos desbalanceados. La matriz de confusión y la curva ROC del conjunto de validación de este modelo de FA se muestran en la Figura 5.15.

Como se mencionó en la sección 4.5.2, se pueden apreciar las diferencias fundamentales en el rendimiento de los clasificadores al equilibrar o no los datos de entrenamiento al observar los resultados del conjunto de validación. Al no equilibrar, hay una importante pérdida de sensibilidad, a cambio de una mejora en la especificidad. Esta última mejora es menor en magnitud comparada con la pérdida de sensibilidad pero, al tener muchas más señales de ritmo no FA que de ritmo FA (6939 y 639 respectivamente), tiene un impacto mucho mayor en la precisión y el F1-score. Por lo tanto, al equilibrar el conjunto de entrenamiento, la

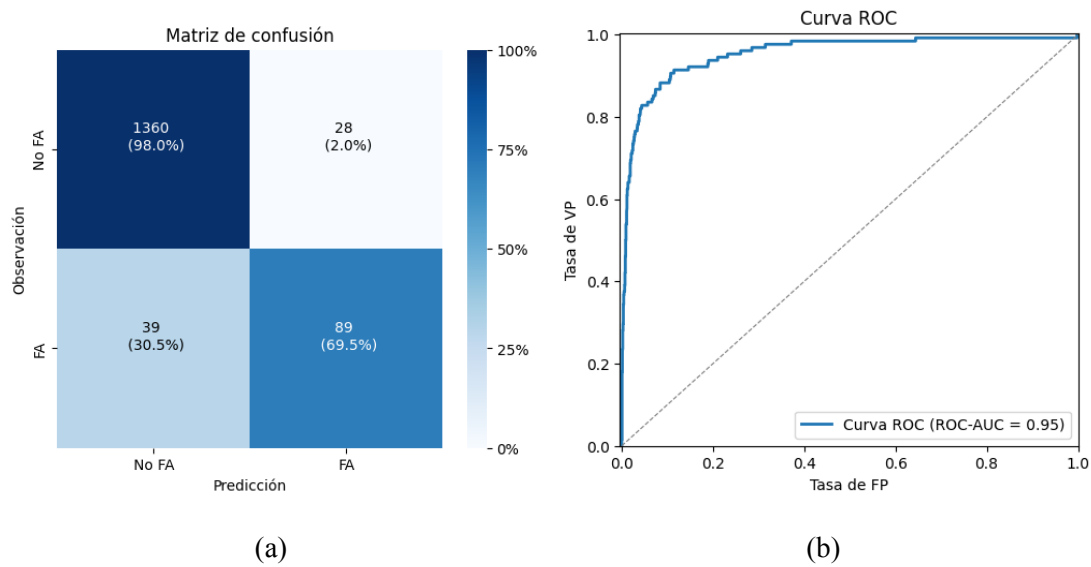


Figura 5.15. Resultados del rendimiento del clasificador no balanceado de FA AdaBoost ganador sobre el conjunto de validación. a) Matriz de confusión. b) Curva ROC.

sensibilidad aumenta significativamente a costa de la especificidad que, aunque sigue siendo elevada, provoca una baja importante en la precisión y el F1-score del conjunto de validación.

5.4.3 Tiempo de cómputo y uso de memoria

El tiempo de cómputo promedio de procesar totalmente una señal del conjunto de validación fue de 0.3203 s, mientras que el uso de memoria máximo promedio fue de 212.4 MiB (mebibyte) o 222.9 MB (megabyte).

5.4.4 Evaluación de la utilidad de la detección de ruido y ectópicos

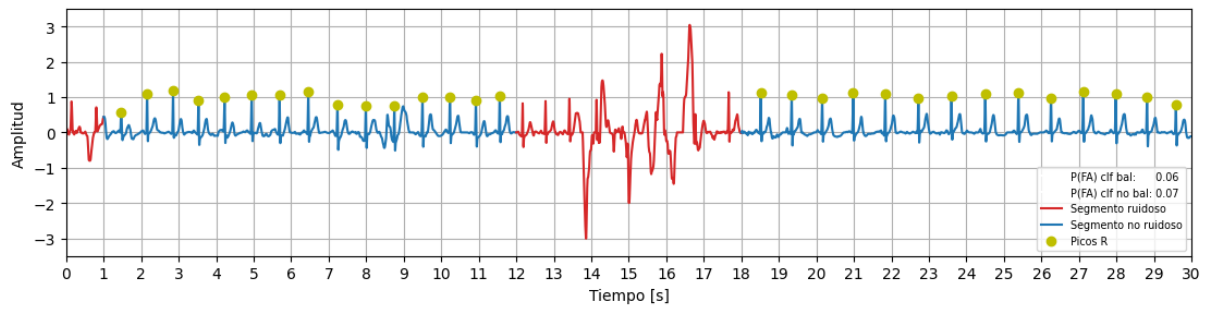
En el apéndice A.1, la Tabla A.1 muestra la comparación de los resultados de detección de FA para dos modelos, RL y AdaBoost, contrastados contra los resultados de los mismos modelos pero omitiendo la etapa de detección de ruido y latidos ectópicos.

Se observa que el rendimiento general del clasificador RL, cuando se entrena con datos balanceados, disminuye al deshabilitar las etapas de detección de ruido y latidos ectópicos. Esta disminución se evidencia a través de una caída del 7-11% en la precisión, que llega a un valor mínimo del 42% en el conjunto de validación.

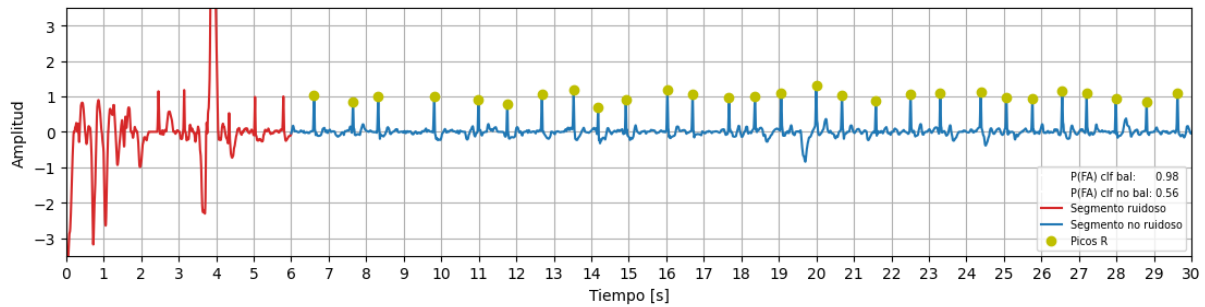
En contraposición, para el clasificador AdaBoost, que se entrena con datos desbalanceados, la precisión parece mantenerse relativamente constante. Sin embargo, la sensibilidad experimenta una notable caída, en particular en la base de datos de Kaggle, donde llega a ser solo del 26.85%, un valor considerado inaceptable.

5.4.5 Clasificación de FA de señales ejemplo

La Figura 5.16 presenta dos ejemplos de señales de SSLAFDB donde se exponen los resultados del algoritmo en su totalidad. El primer paso es el preprocesamiento de las señales para atenuar el ruido existente, resultando en las señales mostradas en la figura. A continuación, se realiza la detección de ruido que permite diferenciar los segmentos ruidosos (en rojo) de los no ruidosos (en azul). A estos últimos se les aplica una detección de latidos (puntos verdes). A partir de estos latidos, se obtiene la serie RR, que se procesa y somete a una detección de ectópicos para eliminarlos, obteniendo así las series NN. Finalmente, estas series se analizan para identificar la presencia de FA en la señal. La probabilidad de FA se muestra en la leyenda de cada gráfico y se calculó tanto con el clasificador entrenado con el conjunto balanceado (clf bal) como con el no balanceado (clf no bal).



(a)



(b)

Figura 5.16. Ejemplos de señales de SSLAFDB con procesamiento completo, donde se realiza una detección de segmentos ruidosos (rojo), detección y clasificación de latidos (puntos verdes) y se determina si tiene ritmo FA. Las probabilidades de FA, predichas por el clasificador balanceado y no balanceado, se muestran en la leyenda de cada gráfico.

a) Señal n° 66, ritmo N. b) Señal n° 52, ritmo A.

En la señal de ritmo N (Figura 5.16a), la probabilidad de FA predicha por ambos clasificadores es similar, y muy próxima a cero, lo que indica que ambos clasificadores predicen de manera acertada la ausencia de FA. En la señal de ritmo A (Figura 5.16b), la presencia de FA se predice de forma acertada. Sin embargo, la probabilidad predicha por cada clasificador difiere considerablemente, siendo mucho más elevada en el caso del clasificador balanceado que en el no balanceado. Esto es coherente con lo observado anteriormente, ya que el clasificador balanceado presenta una sensibilidad notablemente superior, reflejada en un número menor de falsos negativos.

Discusión

Este trabajo se centra en el diseño y validación de un algoritmo para detectar la fibrilación auricular a partir de señales de ECG capturadas con dispositivos móviles. Considerando la creciente importancia de la telemedicina y el monitoreo de salud a través de dispositivos personales, es crucial desarrollar métodos robustos que puedan manejar señales de diferentes calidades. El algoritmo se estructuró en tres etapas fundamentales: detección de ruido, eliminación de latidos ectópicos y determinación de FA. A lo largo de esta discusión, se compararán los resultados obtenidos con investigaciones anteriores y se discutirá la relevancia y el impacto de nuestro enfoque en la detección de fibrilación auricular.

El algoritmo de detección de ruido desarrollado demostró un rendimiento general sólido. Se han documentado numerosos algoritmos de detección de ruido y evaluación de calidad de señal previamente. Uno de los primeros trabajos donde se emplearon los SQIs utilizados en este trabajo, es el realizado por Li et al. [24]. Sin embargo, el rendimiento de su trabajo fue evaluado indirectamente a través del error de estimación de la frecuencia cardíaca, lo que hace imposible una comparación directa de los resultados.

Zhao et al. [23] también desarrollaron un sistema de detección de ruido basado en umbrales de SQIs y lógica difusa. Este sistema fue entrenado y validado utilizando la SSLAFDB, y logró una precisión, sensibilidad y especificidad del 92.67%, 97.33% y 88.67% respectivamente, resultados que se alinean estrechamente con los obtenidos en este trabajo. No obstante, las métricas reportadas en este trabajo y en el de Zhao son difíciles de comparar. En su trabajo, ellos clasificaron entre señales ruidosas y no ruidosas, mientras que en este caso el objetivo es la evaluación de calidad de segmentos ruidosos, utilizando el clasificador para estimar la probabilidad de que ventanas de 5 segundos sean ruidosas y realizando un promedio ponderado para evaluar la calidad de cada segundo individual de la señal. Además, al haber etiquetado y seleccionado manualmente las ventanas utilizadas, no es posible la comparación directa del rendimiento frente a otros algoritmos.

La eliminación de ectópicos es una etapa frecuentemente incorporada en los algoritmos de detección de FA. Entre los diversos enfoques, un método simple basado en simples reglas fue desarrollado por Dash et al. [47]. Este depende únicamente de tres relaciones entre series RR y umbrales asociados. Sin embargo, al igual que muchas técnicas de eliminación de latidos ectópicos utilizadas en los algoritmos de detección de FA, el rendimiento real de su enfoque no ha sido informado dado que no se presentaron métricas para evaluar la efectividad del mismo. García et al. [48] desarrollaron una técnica más sofisticada, creando un clasificador de ectópicos para registros Holter con un rendimiento superior al obtenido con el clasificador propuesto en este trabajo, con especificidad, sensibilidad, precisión y F1-score superiores al 97%. Sin embargo, solo se utilizaron 5800 latidos seleccionados de la base de datos MIT-BIH, eligiendo 2000 latidos de ritmo sinusal, 2000 ectópicos ventriculares y 1800 ectópicos auriculares, dando lugar a un subconjunto de datos acotado y balanceado, por lo que no hay garantías de que estos resultados sean reproducibles utilizando la base de datos entera, como fue hecho en este trabajo. Además, este método fusiona características morfológicas y de la serie RR. Dada la reducida fiabilidad del análisis de morfología en señales de baja calidad, este enfoque no sería óptimo para señales de ECG de dispositivos smartphone.

Entre otros estudios de detección de latidos ectópicos que utilizaron la misma base de datos que se usa en este trabajo (MIT-BIH), se destacan los de Martis et al. [49] y Chazal et al. [50]. En el primero, se emplearon características morfológicas y temporales extraídas del ECG y utilizaron análisis de componentes independientes para reducir la dimensionalidad. Posteriormente, se utilizaron estos atributos reducidos como entrada de una Red Neuronal Probabilística. Los resultados obtenidos fueron muy favorables, con valores de especificidad, sensibilidad, precisión y exactitud superiores al 99%. Sin embargo, no se realizó independencia de pacientes para entrenamiento y testeo, por lo que no se garantiza que el modelo sea generalizable, como afirma Pflugradt et al. [51], lo cual si se realiza en este trabajo. En el trabajo de Chazal et al., por otro lado, si se garantizó la independencia de pacientes en el conjunto de entrenamiento y testeo y se obtuvieron una precisión, sensibilidad, especificidad y F1-score 66.85%, 88.41%, 95.03% y 76.13% respectivamente. Para esto, se utilizó una combinación de 15 atributos basados en la serie RR, morfología y duración del complejo QRS, onda T y onda P. Además, se analizaron independientemente las dos derivaciones presentes por paciente en la

base de datos MIT-BIH para entrenar dos clasificadores, uno por derivación, y luego realizar una clasificación final resultado del consenso entre ambos. En contraste, el detector de latidos ectópicos desarrollado en este trabajo ha mostrado resultados más que satisfactorios al lograr valores muy superiores de precisión (89.48%), F1-score (87.01%) y sensibilidad (99.30%) a costa de una sensibilidad levemente inferior (84.70%), utilizando además una única derivación y clasificador, 12 atributos y sin usar información morfológica, de onda T o de onda P, lo que permite que siga siendo confiable para utilizarse en señales de baja calidad.

El algoritmo de detección de FA demostró un rendimiento sólido en diversas bases de datos, lo que sugiere que el modelo logró un nivel sustancial de generalización. Una ventaja clave del método presentado es su simplicidad; propone un algoritmo sencillo que solo requiere un máximo de ocho características para la clasificación, utilizando un modelo relativamente simple. En contraste, muchos otros métodos diseñados para la detección precisa de FA en señales de baja calidad proponen el uso de modelos complejos como redes neuronales convolucionales, como en el trabajo de Hong et al. [52]. Alternativamente, utilizan clasificadores basados en características similares a este estudio, pero con un conjunto de características más grande, como en el trabajo de Datta et al. [53], que utilizó más de 20 características. Estos dos estudios fueron co-ganadores del primer lugar en el desafío “Physionet Computing in Cardiology” de 2017 [54]. El algoritmo de Datta et al. logró un F1-score para detección de FA del 74% para el conjunto secreto de testeo del desafío. Comparando este resultado con el obtenido para el clasificador ganador utilizando el conjunto de entrenamiento no balanceado, puede verse que se obtuvieron resultados más que satisfactorios, con un F1-score del 76.83% en el conjunto de validación. Además, la comparación de ambos resultados es altamente confiable dado que no solo se utilizó la misma base de datos sino que además en ambos trabajos se utilizó el clasificador AdaBoost para detectar FA.

El sistema de detección de ritmo en tiempo real del Apple Watch 4 (AW4) proporciona otro punto de comparación. Según un artículo publicado en la revista de la “American Heart Association” (AHA) [55], el método de notificación en tiempo real demostró una sensibilidad y especificidad del 41% y 100% respectivamente para la detección de FA. Además, dio resultados inconclusos en el 31% de los casos. En contraste, los resultados obtenidos en este trabajo son mas

que aceptables, demostrando rendimientos muy superiores para la sensibilidad y similares para la especificidad tanto entrenando con el conjunto de datos balanceado como el no balanceado.

Como se observó, utilizar datos de entrenamiento balanceados o no balanceados trae aparejado ventajas y desventajas para cada caso. Una baja en la precisión es importante, dado que esta simboliza la cantidad de falsas alarmas que porcentualmente se generarán. Por otro lado, una baja en la sensibilidad también es trascendente, ya que significa que quedarán sin diagnosticar mas episodios reales de FA, evitando que reciban el correcto tratamiento. Elegir cual de estas dos cosas priorizar es complejo y depende del escenario en el que se planea utilizar el algoritmo.

La caída en el rendimiento del clasificador de FA sobre las distintas bases de datos al quitar la detección de ruido y de latidos ectópicos es un indicador de la importancia de estas dos etapas en la detección de FA. Al margen de esto, la evaluación de calidad de la señal tiene gran importancia por fuera de asistir a mejorar la detección de FA. Este algoritmo esta pensado para ser utilizado en contextos de señales de ECG obtenidas por dispositivos simples operados sin la presencia de profesionales, por lo que existen muchos escenarios en los cuales podrían obtenerse señales completamente o casi completamente ruidosas, como por ejemplo que el usuario esté utilizando el dispositivo de forma incorrecta o que este presente algún tipo de avería o falla que afecte la toma de la señal. De no evaluar la calidad de la señal, el usuario podría no advertir estos escenarios, y siempre otorgaría un resultado, generando pérdida de fiabilidad en el resultado del diagnóstico.

Para comparar el uso de memoria, y tomando en cuenta el objetivo de ejecutar este algoritmo en dispositivos smartphone, se puede tomar como referencia el Motorola Moto G41, el celular más vendido en Argentina según un artículo del diario El Cronista en marzo de 2022 [56], que cuenta con 4 GB de memoria RAM y un procesador MediaTek Helio G85. Teniendo esto en consideración, un uso de memoria de 222.9 MB es aceptable, ya que representa el 5.44% de la memoria total. Además, este consumo será solamente momentáneo, debido a que la aplicación se utiliza de manera ocasional y el algoritmo no está diseñado para correr en segundo plano. El algoritmo, al ejecutarse en una MacBook Pro M1, presenta un tiempo promedio de 0.3203 s, lo cual sugiere su factibilidad para un uso en tiempo real en smartphones. De acuerdo

con un estudio de *benchmarking* en hardware informático, el procesador Apple M1 supera en rendimiento al Helio G85 por un factor de cinco [57]. Si extrapolamos esta relación al tiempo de ejecución, incluso en el peor de los casos, donde el tiempo se quintuplicara, aún estaríamos considerando un tiempo inferior a 2 s. Esta cifra se mantiene dentro de los márgenes aceptables para el objetivo de este trabajo.

Conclusiones

En conclusión, en este trabajo se desarrolló un robusto algoritmo basado en aprendizaje automático para la detección de la fibrilación auricular, utilizando un enfoque de tres etapas: detección de ruido, eliminación de latidos ectópicos y detección de FA. El algoritmo muestra sólidas métricas de rendimiento para las tres etapas y es particularmente robusto, ya que ha sido entrenado con señales de baja calidad adquiridas por dispositivos móviles, tomando en cuenta y lidiando satisfactoriamente con los altos niveles de ruido presentes en estas. Además, toma en cuenta la existencia de una amplia gama de ritmos posibles, discriminando la FA no solo de ritmo sinusal, sino también de otras arritmias, logrando aún así un rendimiento en la detección de FA de niveles comparables con los métodos encontrados existentes.

El novedoso método para la detección de latidos ectópicos propuesto puede ser un punto de partida fundamental para el desarrollo de nuevas técnicas en el área, mostrando resultados muy satisfactorios utilizando un bajo número de atributos basados únicamente en la serie RR. Esto permite la detección de suprasístoles en señales de baja calidad al evitar utilizar parámetros morfológicos. Hasta donde llega nuestro conocimiento, este es el primer algoritmo de este estilo. Sin embargo, futuros estudios en bases de datos con anotaciones de latidos son necesarios para obtener una medida mas real de su eficiencia.

Con el aumento de la telemonitorización en la atención médica, el algoritmo podría desempeñar un papel crucial en mejorar el acceso a la telemedicina y la atención al paciente. Sin embargo, se requiere una validación adicional en señales de pacientes telemonitorizados a través de la plataforma Virtual Sense. Futuros estudios evaluarán su rendimiento en esta plataforma, proporcionando una valiosa perspectiva sobre su aplicabilidad en el mundo real. Los resultados también podrían guiar refinamientos adicionales en el algoritmo, incluida la incorporación de características y datos de entrenamiento, lo que mejoraría su utilidad en la detección de la FA. Finalmente, la selección final del algoritmo a utilizar debe ser establecida por Virtual Sense en base a la opinión de los médicos del ICBA que testearán el algoritmo y del ambiente de uso y objetivo que la empresa priorice.

Bibliografía

- [1] M. de Salud de Argentina, “Glosario de salud,” 2023, ultimo acceso: Abril 2023. [Online]. Available: <https://www.argentina.gob.ar/salud/glosario/fibrilacionauricular>
- [2] M. P. Turakhia, J. Shafrin, K. Bogner, J. Trocio, Y. Abdulsattar, D. Wiederkehr, and D. P. Goldman, “Estimated prevalence of undiagnosed atrial fibrillation in the united states,” *PLoS One*, vol. 13, no. 4, pp. 1–11, 04 2018. [Online]. Available: <https://doi.org/10.1371/journal.pone.0195088>
- [3] O. M. de la Salud, 2023, ultimo acceso: Junio 2023. [Online]. Available: <https://www.who.int/es/health-topics/cardiovascular-diseases>
- [4] I. C. de Buenos Aires, 2023, ultimo acceso: Junio 2023. [Online]. Available: <https://www.icba.com.ar/acerca-del-icba/prensa/2019/fibrilacion-auricular-una-de-las-enfermedades-mas-comunes-de-latinoamerica>
- [5] J. Kornej, C. Börschel, E. Benjamin, and R. Schnabel, “Epidemiology of atrial fibrillation in the 21st century: Novel methods and new insights,” *Circulation Research*, vol. 127, pp. 4–20, 06 2020.
- [6] M. Clinic, “Atrial fibrillation,” 2023, ultimo acceso: Junio 2023. [Online]. Available: <https://www.mayoclinic.org/diseases-conditions/atrial-fibrillation/symptoms-causes/syc-20350624>
- [7] M. A. Altuve, “Adaptación de un método de detección de latidos y de segmentación del electrocardiograma al neonato prematuro paracaracterizar episodios de apnea-bradicardia,” p. 8, 2010. [Online]. Available: https://www.researchgate.net/publication/246992000_Adaptacion_de_un_metodo_de_deteccion_de_latidos_y_de_segmentacion_del_electrocardiograma_al_neonato_prematuro_paracaracterizar_episodios_de_apnea-bradicardia
- [8] W. H. R. Associates, “12-lead electrocardiogram,” 2020, ultimo acceso: Mayo 2023. [Online]. Available: <https://www.washingtonhra.com/ekg-monitoring/12-lead-electrocardiogram-ekg.php>

- [9] AliveCor, “Kardiamobile,” 2022, ultimo acceso: Mayo 2023. [Online]. Available: <https://store.kardia.com/products/kardiamobile>
- [10] WeCardio, “wecardio” at <http://en.wecardio.com/> (last accessed april 2023).”
- [11] M. Expo, “Grabador de eventos de ecg inalámbrico wecardio-un,” 2023, ultimo acceso: Mayo 2023. [Online]. Available: <https://www.medicalexpo.es/prod/borsam-biomedical-instrument/product-112306-742003.html>
- [12] L. Sörnmo and P. Laguna, *The Electrocardiogram—A Brief Background*, 12 2005, pp. 411–452.
- [13] S. Cenzabelo, “Método para la detección y clasificación de latidos en la señal de ecg.” Ph.D. dissertation, 2004.
- [14] K. Farzam and J. R. Richards, “Premature ventricular contraction,” *StatPearls [Internet]*, Enero 2023. [Online]. Available: <https://www.ncbi.nlm.nih.gov/books/NBK532991/>
- [15] J. Heaton and S. Yandrapalli, “Premature atrial contractions,” *StatPearls [Internet]*, Enero 2023. [Online]. Available: <https://www.ncbi.nlm.nih.gov/books/NBK559204/>
- [16] G. Clifford, C. Liu, B. Moody, L.-w. Lehman, I. Silva, Q. Li, A. Johnson, and R. Mark, “Af classification from a short single lead ecg recording: the physionet computing in cardiology challenge 2017,” sep 2017. [Online]. Available: <https://doi.org/10.22489/2Fcinc.2017.065-469>
- [17] P. Wagner, N. Strodthoff, R.-D. Bousseljot, D. Kreiseler, F. Lunze, W. Samek, and T. Schaeffter, “Pt看xl, a large publicly available electrocardiography dataset,” *Scientific Data*, vol. 7, p. 154, 05 2020.
- [18] A. Goldberger, L. Amaral, L. Glass, S. Havlin, J. Hausdorg, P. Ivanov, R. Mark, J. Mietus, G. Moody, C.-K. Peng, H. Stanley, and P. Physiobank, “Components of a new research resource for complex physiologic signals,” *PhysioNet*, vol. 101, 01 2000.
- [19] K. Inc., “ptb-xl - atrial fibrillation detection” at <https://www.kaggle.com/datasets/arjunascagnetto/ptb-xl-atrial-fibrillation-detection?resource=download> (last accessed april 2023).”

- [20] P. A. P. B. M. V. . S. L. Sološenko, A., “Model for simulating ecg and ppg signals with arrhythmia episodes (version 1.3.1),” 2022. [Online]. Available: <https://physionet.org/content/ecg-ppg-simulator-arrhythmia/1.3.1/>
- [21] M. B. Hossain, S. K. Bashar, A. J. Walkey, D. D. McManus, and K. H. Chon, “An accurate qrs complex and p wave detection in ecg signals using complete ensemble empirical mode decomposition with adaptive noise approach,” *IEEE Access*, vol. 7, pp. 128 869–128 880, 2019.
- [22] D. Panigrahy and P. Sahu, “P and t wave detection and delineation of ecg signal using differential evolution (de) optimization strategy,” *Australasian Physical Engineering Sciences in Medicine*, vol. 41, 02 2018.
- [23] Z. Zhao and Y. Zhang, “Sqi quality evaluation mechanism of single-lead ecg signal based on simple heuristic fusion and fuzzy comprehensive evaluation,” *Frontiers in Physiology*, vol. 9, 2018. [Online]. Available: <https://www.frontiersin.org/articles/10.3389/fphys.2018.00727>
- [24] Q. Li, R. G. Mark, and G. D. Clifford, “Robust heart rate estimation from multiple asynchronous noisy sources using signal quality indices and a kalman filter,” *Physiological Measurement*, vol. 29, no. 1, p. 15, dec 2007. [Online]. Available: <https://dx.doi.org/10.1088/0967-3334/29/1/002>
- [25] B. Porr, L. Howell, and I. S. Yoav Nir, “Popular ecg r peak detectors written in python,” 2023. [Online]. Available: <https://github.com/berndporr/py-ecg-detectors>
- [26] M.-T. Puth, M. Neuhäuser, and G. Ruxton, “Effective use of spearman’s and kendall’s correlation coefficients for association between two measured traits,” *Animal Behaviour*, vol. 102, 04 2015.
- [27] H. Akoglu, “User’s guide to correlation coefficients,” *Turkish Journal of Emergency Medicine*, vol. 18, 08 2018.
- [28] xgboost developers, “Xgboost documentation,” 2022. [Online]. Available: <https://xgboost.readthedocs.io/en/stable/index.html>

- [29] N. inform, “Supraventricular tachycardia,” 2023, ultimo acceso: Julio 2023. [Online]. Available: [https://www.nhsinform.scot/illnesses-and-conditions/heart-and-blood-vessels/conditions/supraventricular-tachycardia#:~:text=About%20supraventricular%20tachycardia,-Supraventricular%20tachycardia%20\(SVT&text=They%20may%20occur%20regularly%2C%20several,beats%20per%20minute%20at%20rest\)](https://www.nhsinform.scot/illnesses-and-conditions/heart-and-blood-vessels/conditions/supraventricular-tachycardia#:~:text=About%20supraventricular%20tachycardia,-Supraventricular%20tachycardia%20(SVT&text=They%20may%20occur%20regularly%2C%20several,beats%20per%20minute%20at%20rest))
- [30] A. Khalaf and S. Mohammed, “Verification and comparison of mit-bih arrhythmia database based on number of beats,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, p. 4950, 12 2021.
- [31] L. Sörnmo, *Atrial Fibrillation from an Engineering Perspective*, 1st ed. Springer, 2018.
- [32] S. Meek and F. Morris, “Abc of clinical electrocardiography: Introduction. ii—basic terminology,” *BMJ*, vol. 324, no. 7335, pp. 470–473, 2002.
- [33] K. Tateno and L. Glass, “Automatic detection of atrial fibrillation using the coefficient of variation and density histograms of rr and Δrr intervals,” *Medical biological engineering computing*, vol. 39, pp. 664–71, 12 2001.
- [34] S. Datta, C. Puri, A. Mukherjee, R. Banerjee, A. D. Choudhury, R. Singh, A. Ukil, S. Bandyopadhyay, A. Pal, and S. Khandelwal, “Identifying normal, af and other abnormal ecg rhythms using a cascaded binary classifier,” in *2017 Computing in Cardiology (CinC)*, 2017, pp. 1–4.
- [35] F. Shaffer and J. P. Ginsberg, “An overview of heart rate variability metrics and norms,” *Frontiers in Public Health*, vol. 5, 2017. [Online]. Available: <https://www.frontiersin.org/articles/10.3389/fpubh.2017.00258>
- [36] MIT, “Course 18.443 statistics for applications” at https://dspace.mit.edu/bitstream/handle/1721.1/100851/18-443-spring-2009/contents/assignments/MIT18_443s09_assn03_res03.pdf (last accessed june 2023).”
- [37] D. Lake and J. Moorman, “Accurate estimation of entropy in very short physiological time series: The problem of atrial fibrillation detection in implanted ventricular

- devices,” *American journal of physiology. Heart and circulatory physiology*, vol. 300, pp. H319–25, 10 2010.
- [38] A. Petrénas, V. Marozas, and L. Sörnmo, “Low-complexity detection of atrial fibrillation in continuous long-term monitoring,” *Comput Biol Med*, vol. 65, 2015. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/25666902/>
- [39] J. Park, S. Lee, and M. Jeon, “Atrial fibrillation detection by heart rate variability in poincare plot,” *Biomedical engineering online*, vol. 8, p. 38, 12 2009.
- [40] A. S. University, “How to check your heart and pulse rate,” 2023, ultimo acceso: Junio 2023. [Online]. Available: <https://askabiologist.asu.edu/content/check-your-heart-rate>
- [41] M. Plus, “Palpitaciones cardíacas,” 2023, ultimo acceso: Junio 2023. [Online]. Available: <https://medlineplus.gov/spanish/ency/article/003081.htm#:~:text=Normalmente%2C%20el%20coraz%C3%B3n%20late%20entre,de%2060%20latidos%20por%20minuto.>
- [42] C. Clinic, “Heart rate variability,” 2021, ultimo acceso: Junio 2023. [Online]. Available: <https://my.clevelandclinic.org/health/symptoms/21773-heart-rate-variability-hrv>
- [43] M. Clinic, “Heart arrhythmia,” 2023, ultimo acceso: Junio 2023. [Online]. Available: <https://www.mayoclinic.org/diseases-conditions/heart-arrhythmia/symptoms-causes/syc-20350668>
- [44] A. Widmann and E. Schröger, “Filter effects and filter artifacts in the analysis of electrophysiological data,” *Frontiers in psychology*, vol. 3, p. 233, 07 2012.
- [45] Scikit-learn, “Documentación de regresión logística,” 2023, ultimo acceso: Junio 2023. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html
- [46] XGBoost, “Documentación de xgboost,” 2023, ultimo acceso: Junio 2023. [Online]. Available: <https://xgboost.readthedocs.io/en/stable/R-package/discoverYourData.html>

- [47] S. Dash, K. Chon, S. Lu, and E. Raeder, "Automatic real time detection of atrial fibrillation," *Annals of biomedical engineering*, vol. 37, pp. 1701–9, 07 2009.
- [48] A. García, H. Romano, E. Laciár, and R. Correa, "Development of an algorithm for heartbeats detection and classification in holter records based on temporal and morphological features," *Journal of Physics: Conference Series*, vol. 332, no. 1, p. 012023, dec 2011. [Online]. Available: <https://dx.doi.org/10.1088/1742-6596/332/1/012023>
- [49] R. Martis, U. R. Acharya, and C. Lim, "Ecg beat classification using pca, lda, ica and discrete wavelet transform," *Biomedical Signal Processing and Control*, vol. 8, pp. 437–448, 09 2013.
- [50] P. Chazal, M. O'Dwyer, and R. Reilly, "Automatic classification of heartbeats using ecg morphology and heartbeat interval features," *IEEE transactions on bio-medical engineering*, vol. 51, pp. 1196–206, 08 2004.
- [51] M. Pflugradt, K. Geissdoerfer, M. Goernig, and R. Orglmeister, "A fast multimodal ectopic beat detection method applied for blood pressure estimation based on pulse wave velocity measurements in wearable sensors," *Sensors*, vol. 17, no. 1, 2017. [Online]. Available: <https://www.mdpi.com/1424-8220/17/1/158>
- [52] S. Hong, M. Wu, Y. Zhou, Q. Wang, J. Shang, H. Li, and J. Xie, "Encase: an ensemble classifier for ecg classification using expert features and deep neural networks," 09 2017.
- [53] S. Datta, C. Puri, A. Mukherjee, R. Banerjee, A. Dutta Choudhury, R. Singh, A. Ukil, S. Bandyopadhyay, A. Pal, and S. Khandelwal, "Identifying normal, af and other abnormal ecg rhythms using a cascaded binary classifier," 09 2017.
- [54] Physionet, "2017 computing in cardiology challenge," 2017, ultimo acceso: Julio 2023. [Online]. Available: <https://physionet.org/content/challenge-2017/1.0.0/>
- [55] D. Seshadri, B. Bittel, D. Browsey, P. Houghtaling, C. Drummond, M. Desai, and A. Gillinov, "Accuracy of apple watch for detection of atrial fibrillation," *Circulation*, vol. 141, pp. 702–703, 02 2020.

- [56] Cronista, “Los tres celulares más vendidos en argentina y por qué valen menos de 40.000,” 2022, último acceso: Septiembre 2023. [Online]. Available: <https://www.cronista.com/infotechnology/gadgets/los-tres-celulares-mas-vendidos-en-argentina-que-tienen-y-por-que-valen-menos-de-40-000>
- [57] Gadgetversus, 2023, último acceso: Septiembre 2023. [Online]. Available: <https://gadgetversus.com/processor/apple-m1-vs-mediatek-mt6769z-helio-g85/>

Apéndice A

A.1 Rendimiento de los 2 clasificadores de FA, con y sin detección de ruido y ectópicos

| Datos balanceados | Detección de ruido y ectópicos | Exactitud | Precisión | Sensibilidad | Especificidad | F1-Score |
|-------------------|--------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Si | Activada | V: 92.35% K: 89.20% S: 91.95% | V: 52.73% K: 72.39% S: 87.07% | V: 90.62% K: 90.17% S: 98.40% | V: 92.51% K: 88.89% S: 85.59% | V: 66.67% K: 80.31% S: 92.39% |
| Si | Desactivada | V: 89.61% K: 85.64% S: 91.07% | V: 42.02% K: 65.27% S: 86.19% | V: 90.00% K: 89.61% S: 97.82% | V: 89.57% K: 84.34% S: 84.32% | V: 57.29% K: 75.53% S: 91.63% |
| No | Activada | V: 95.58% K: 88.88% S: 98.51% | V: 76.07% K: 89.76% S: 97.20% | V: 69.53% K: 61.49% S: 81.22% | V: 97.98% K: 97.73% S: 97.69% | V: 72.65% K: 72.98% S: 88.49% |
| No | Desactivada | V: 95.93% K: 81.13% S: 82.85% | V: 80.00% K: 89.45% S: 98.10% | V: 63.33% K: 26.85% S: 98.70% | V: 98.67% K: 98.96% S: 67.00% | V: 70.07% K: 41.31% S: 79.62% |

Tabla A.1. Comparación de rendimiento de clasificadores ganadores entrenados con datos balanceados y no balanceados, utilizando y sin utilizar la etapa de detección de ruido y latidos ectópicos. V: conjunto de validación, K: base de datos de Kaggle, y S: base de datos sintética.

Apéndice B

B.1 Lista de librerías y sus respectivas versiones

| Librería | Versión |
|-------------------|---------|
| matplotlib | 3.6.2 |
| matplotlib-inline | 0.1.6 |
| memory-profiler | 0.61.0 |
| numpy | 1.23.4 |
| pandas | 1.5.1 |
| pandas-summary | 0.2.0 |
| pip | 22.3 |
| py-ecg-detectors | 1.3.2 |
| scikit-learn | 1.2.2 |
| scipy | 1.9.3 |
| seaborn | 0.12.2 |
| statsmodels | 0.13.5 |
| wfdb | 4.1.0 |
| xgboost | 1.7.4 |

Tabla B.1. Librerías y sus versiones