

Escuela Superior Politécnica del Litoral

Facultad de Ingeniería en Electricidad y Computación

Implementación de un Sistema para Análisis y Clasificación de
Electrocardiogramas con Algoritmos de Inteligencia Artificial

INGE-2925

Proyecto Integrador

Previo la obtención del Título de:

Ingeniero en Electrónica y Automatización

Presentado por:

Adriana Paola Ponce Marcalla

Jostin Josue Zapata Chancay

Guayaquil - Ecuador

Año: 2025

Dedicatoria

El presente proyecto le dedicamos a Dios,
por ser nuestra guía y motivación para
hacer todas las cosas con excelencia, para
Él.

Agradecimientos

Agradecemos a nuestros estimados maestros, por su dedicación, conocimiento y guía. Al MSc. Miguel Yapur, por contagiarnos su pasión y entrega a la electrónica médica. Al Dr. Rafael Castilla por orientación y apoyo en el desarrollo de nuestro proyecto. Al Ph.D. Efrén Herrera por su acompañamiento y motivación.

Adriana

Doy gracias a mis padres, Isidro Ponce y Zaida Marcalla, por su esfuerzo, amor, dedicación y apoyo en cada momento. A mis abuelos, Segundo Marcalla y Rosario Sinche, por ser mi inspiración y motivo de superación. A mis hermanos, Christian, Andrés, Alejandro por siempre estar para mí, apoyar y festejar mis logros. A mi tía Pilar por enseñarme hacer las cosas con excelencia y no tener límites para soñar en grande.

Jostin

Quiero expresar mi agradecimiento a mis padres, Pedro Zapata y Mónica Chancay, por ser el ejemplo más claro de esfuerzo, dedicación y amor para mí. Gracias por brindarme siempre su apoyo incondicional, sobre todo, gracias por enseñarme a perseguir mis sueños.

A mis hermanos, Johan Zapata, Dylan Zapata y a mi hermana, Danna Zapata, a quienes también acompañó como hermano mayor. Sin duda, han sido una de las motivaciones constantes que he tenido para seguir mejorando y dar lo mejor de mí.

Este logro también es suyo, porque sin ustedes no habría sido posible.

Declaración Expresa

Nosotros Adriana Paola Ponce Marcalla y Jostin Josue Zapata Chancay acordamos y reconocemos que:

La titularidad de los derechos patrimoniales de autor (derechos de autor) del proyecto de graduación corresponderá al autor o autores, sin perjuicio de lo cual la ESPOL recibe en este acto una licencia gratuita de plazo indefinido para el uso no comercial y comercial de la obra con facultad de sublicenciar, incluyendo la autorización para su divulgación, así como para la creación y uso de obras derivadas. En el caso de usos comerciales se respetará el porcentaje de participación en beneficios que corresponda a favor del autor o autores.

La titularidad total y exclusiva sobre los derechos patrimoniales de patente de invención, modelo de utilidad, diseño industrial, secreto industrial, software o información no divulgada que corresponda o pueda corresponder respecto de cualquier investigación, desarrollo tecnológico o invención realizada por nosotros durante el desarrollo del proyecto de graduación, pertenecerán de forma total, exclusiva e indivisible a la ESPOL, sin perjuicio del porcentaje que nos corresponda de los beneficios económicos que la ESPOL reciba por la explotación de nuestra innovación, de ser el caso.

En los casos donde la Oficina de Transferencia de Resultados de Investigación (OTRI) de la ESPOL comunique los autores que existe una innovación potencialmente patentable sobre los resultados del proyecto de graduación, no se realizará publicación o divulgación alguna, sin la autorización expresa y previa de la ESPOL.

Guayaquil, 26 de agosto del 2025.



Adriana Ponce Marcalla



Jostin Zapata Chancay

Evaluadores

Ph. D. Efrén Vinicio Herrera Muentes

Profesor de Materia

M. Sc. Miguel Yapur Auad

Tutor de proyecto

Resumen

Las enfermedades cardiovasculares representan la principal causa de mortalidad a nivel mundial. En ellas, las arritmias requieren detección temprana y precisa, siendo el electrocardiograma (EKG) la herramienta estándar para su análisis. Sin embargo, la interpretación manual demanda tiempo y experiencia clínica, lo que limita su aplicabilidad. Tiene como objetivo implementar un sistema de clasificación de latidos cardíacos en señales de EKG mediante técnicas de IA, utilizando redes neuronales convolucionales (CNN) y la clasificación estandarizada de la AMMI. El proyecto se desarrolló utilizando la base de datos MIT-BIH Arrhythmia, procesada mediante técnicas de filtrado pasa eliminación de ruido, normalización y segmentación de latidos. Se aplicó balanceo de datos con SMOTE y posteriormente se entrenó y validó una CNN en Python, empleando bibliotecas como WFDB, TensorFlow y Keras. Los resultados evidenciaron un modelo con capacidad para clasificar latidos cardíacos en las cinco categorías definidas por la AAMI, alcanzando métricas de precisión y sensibilidad para un diagnóstico confiable. El sistema constituye una herramienta complementaria para el personal médico y potencial de integración en dispositivos portátiles, telemedicina, contribuyendo a la detección temprana de arritmias y optimización de la atención médica.

Palabras Clave: Arritmias, Redes Neuronales, AAMI.

Abstract

English to write the same as described before in the Resumen. Use cursive fonts in this section. Cardiovascular diseases are the leading cause of mortality worldwide. Arrhythmias require early and accurate detection, with the electrocardiogram (EKG) being the standard tool for analysis. However, manual interpretation requires time and clinical expertise, which limits its applicability. The project aims to implement a heartbeat classification system in EKG signals using artificial intelligence techniques, utilizing convolutional neural networks (CNNs) and the standardized AMMI classification. The project was developed using the MIT-BIH Arrhythmia database, processed using noise removal, normalization, and heartbeat segmentation filtering techniques. Data balancing was applied with SMOTE, and a CNN was subsequently trained and validated in a Python environment using specialized libraries such as WFDB, TensorFlow, and Keras. The results showed a model capable of classifying heartbeats into the five categories defined by the AAMI, achieving accuracy and sensitivity metrics for a reliable diagnosis. The system constitutes a complementary tool for medical personnel and has the potential for integration into portable devices and telemedicine, contributing to the early detection of arrhythmias and the optimizing medical care.

Keywords: Arrhythmias, AI, Neural Networks, AAMI.

Índice general

Resumen.....	I
<i>Abstract</i>	II
Índice general.....	III
Abreviaturas.....	V
Simbología	VI
Índice de figuras.....	VII
Índice de tablas	VII
Capítulo 1.....	1
1.1 Introducción	2
1.2 Descripción del Problema	3
1.3 Justificación del Problema	4
1.4 Objetivos	5
1.4.1 Objetivo general.....	5
1.4.2 Objetivos específicos	5
1.5 Marco teórico	5
Capítulo 2.....	10
2.1 Metodología.....	11
2.2 Carga de la base de datos	14
2.3 Preprocesamiento de la señal	15
2.4 Entrenamiento del modelo	19
2.5 Evaluación y prueba del modelo	22
Capítulo 3.....	24
3. Resultados y análisis.....	25
3.1 Precisión y pérdida del entrenamiento	25
3.2 Matriz de confusión.....	25
3.3 Métricas generales.....	27

3.4	Prueba con EKG de otros pacientes	28
3.5	Análisis de resultados.....	29
Capítulo 4.....		31
4.	Conclusiones y recomendaciones	32
4.1	Conclusiones	32
4.2	Recomendaciones.....	32
Referencias.....		33

Abreviaturas

AAMI	Association for the Advancement of Medical Instrumentation (Asociación para el Avance de la Instrumentación Médica)
ACC	Accuracy (Precisión global del modelo)
EKG	Electrocardiograma
IA	Inteligencia Artificial
CNN	Convolutional Neural Network (Red Neuronal Convolucional)
CPU	Central Processing Unit (Unidad Central de Procesamiento)
FFT	Fast Fourier Transform (Transformada Rápida de Fourier)
FPR	False Positive Rate (Tasa de Falsos Positivos)
F1-score	Métrica de equilibrio entre precisión (precisión) y sensibilidad (recall)
GPU	Graphics Processing Unit (Unidad de Procesamiento Gráfico)
MIT-BIH	Massachusetts Institute of Technology – Beth Israel Hospital Arrhythmia Database
ML	Machine Learning (Aprendizaje Automático)
QRS	Complejo QRS (conjunto de ondas en el EKG que representan la despolarización ventricular)
RMS	Root Mean Square (Valor cuadrático medio)
R-peak	Pico R (máximo de la onda R en el complejo QRS del EKG)
SNR	Signal-to-Noise Ratio (Relación Señal/Ruido)
SMOTE	Synthetic Minority Over-sampling Technique (Técnica de sobre-muestreo sintético de minorías)
TPR	True Positive Rate (Tasa de Verdaderos Positivos)
TPU	Tensor Processing Unit (Unidad de Procesamiento de Tensores)

Simbología

mV	Milivoltio
Hz	Hertz
kHz	Kilohertz
s	Segundos
ms	Milisegundos
bpm	Latidos por minuto
%	Porcentaje
Ω	Ohm

Índice de figuras

- Fig 1.1. Sistema eléctrico del corazón [5].
- Fig 2.1. GE cardiosoft cam-usb A/T.
- Fig 2.2. Procesamiento EKG
- Fig 2.3. Controlador
- Fig 2.4. Transferencia de señal a PC.
- Fig 2.5. Metodología para implementación del sistema.
- Fig 2.6. Función de guardado para EKG en formato CSV.
- Fig 2.7. Función para eliminar ruido de una señal.
- Fig 2.8. Segmentación de latidos.
- Fig 2.9. Normalización de latidos
- Fig 2.10. Gráficas de latidos por clase.
- Fig 2.11. Distribución de datos por clase
- Fig 2.12. Balance con SMOTE y ajuste de datos al número deseado.
- Fig 2.13. División de datos para entrenamiento y validación.
- Fig 2.14. Arquitectura del modelo.
- Fig 2.15. Entrenamiento del modelo.
- Fig 2.16. Guardado del modelo
- Fig 2.17. Método para obtención de precisión y pérdidas del modelo.
- Fig 2.18. Implementación de la matriz de confusión
- Fig 2.19. Obtención del reporte de métricas del modelo por clase.
- Fig 2.20. Parámetros para interpretación de resultados del modelo.
- Fig 2.21. Carga de modelo
- Fig 3.1. Gráfica de precisión y pérdida durante el entrenamiento.
- Fig 3.2. Matriz de confusión
- Fig 3.3. Matriz de confusión porcentual
- Fig 3.4. Señal EKG de paciente sano.
- Fig 3.5. Señal EKG de paciente con anomalías cardíacas.

Índice de tablas

- Tabla 3.1 Métricas de modelo

Capítulo 1

1.1 Introducción

Según la World Heart Federation, en el año 2021, las enfermedades cardiovasculares (ECV) provocaron 20.5 millones de muertes, consolidándose como la principal causa de muerte del mundo, tomando en cuenta que, cuatro de cada cinco defunciones por ECV ocurrieron en países de ingresos medio y bajo [1]. Dentro de este grupo, las arritmias cardíacas requieren una detección temprana y precisa para reducir el riesgo de complicaciones graves, por ende, el electrocardiograma (EKG) es la herramienta estándar para analizar la actividad eléctrica del corazón y detectar esta clase de anomalías.

Pues que, la interpretación de los registros EKG demanda una vasta experiencia clínica por parte del cardiólogo y tiempo considerable, así que el diagnóstico puede verse afectado por errores humanos, lo que limita su aplicabilidad en entornos con recursos médicos limitados o escasa disponibilidad de especialistas médicos.

Frente a esta problemática, el uso de algoritmos de inteligencia artificial (IA) surge como una herramienta eficaz para el personal médico frente al análisis de las señales fisiológicas del corazón. De esta manera se propone el desarrollo de un sistema para la clasificación de los latidos cardíacos en señales de EKG, empleando la base de datos pública MIT-BIH Arrhythmia y siguiendo la clasificación estandarizada de la AAMI (Association for the Advancement of Medical Instrumentation), que agrupa los latidos en cinco clases clínicas: normales (N), supraventriculares (S), ventriculares (V), de fusión (F) y no clasificados (Q).

La implementación se realizará en un entorno de programación accesible y eficiente, utilizando de Jupiter Notebook y lenguaje de programación Python junto con bibliotecas especializadas como WFDB, TensorFlow y Keras, dentro de la plataforma de Visual Studio Code. El proceso comenzará con la descarga y análisis de registros EKG reales, los cuales serán preprocesados mediante técnicas como filtrado para eliminación de ruido,

segmentación de latidos y normalización. Posteriormente, los latidos serán etiquetados conforme al estándar AAMI y utilizados como entrada para entrenar un modelo, cuyo diseño permite reconocer patrones de la señal cardíaca y clasificar los latidos con alta precisión.

La razón de este proyecto es aportar una solución tecnológica desde la perspectiva de la ingeniería electrónica y la inteligencia artificial al campo de la salud, facilitando herramientas que puedan ser integradas en dispositivos portátiles, plataformas de telemedicina o centros de salud. Se espera que el sistema propuesto contribuya a mejorar la eficiencia del diagnóstico, reducir la carga de trabajo del personal médico y extender la cobertura del monitoreo cardíaco a regiones. En conjunto, el proyecto se alinea con los objetivos de innovación tecnológica y salud preventiva, respondiendo a una necesidad clínica real mediante el aprovechamiento de los avances actuales en IA y procesamiento de señales biomédicas.

1.2 Descripción del Problema

En la actualidad, hay muchos electrocardiógrafos que, mediante software arroja una interpretación y diagnóstico automático de electrocardiogramas, que sirve como soporte para médicos. En un estudio realizado, se evaluó el rendimiento de siete programas de interpretación automática de EKG en más de 2,000 EKG digitales obtenidos de hospitales y bases de datos en Europa, EE. UU. y Australia, el cual arrojo en la detección en ritmos anormales una precisión promedio de 84.9% y una fiabilidad clínica, que es la concordancia del diagnóstico automático y realizado por profesional de salud, de 54%, descrito por De Bie et Al [2].

Ante esta problemática, se propone implementar un sistema automatizado que clasifique latidos cardíacos a partir de registros públicos de EKG, siguiendo el estándar de la Asociación para el Avance de Instrumentación Médica (AAMI). En el cual, con el uso de

redes neuronales convolucionales (CNN), se busca obtener una clasificación precisa y confiable que apoye el diagnóstico oportuno y reduzca la carga del personal médico.

Este proyecto, es desarrollado en conjunto con el cardiólogo Dr. Rafael Castilla, de la corporación médica InterHospital, que busca profundizar el alcance de los algoritmos de inteligencia artificial (IA) en detección de arritmias a partir de una clasificación de latidos de electrocardiogramas.

La restricción de nuestro proyecto es enlazar nuestro sistema asistido por IA con el software predeterminado de un equipo de EKG. Puesto que, los resultados obtenidos por el equipo de EKG deben de exportarse a formatos compatible con el sistema entrenado.

1.3 Justificación del Problema

En los últimos años la tecnología en el sector de la salud ha avanzado de manera significativa para la detección de enfermedades con un alto índice de mortalidad y las enfermedades cardíacas son unos de los problemas que afectan a una significativa proporción de la población, por lo que, se requiere métodos confiables para su temprana detección. El uso de datos reales como los del MIT-BIH Arrhythmia Database y técnicas de IA modernas, son métodos que permiten obtener un sistema capaz de detectar una afección cardíaca, mediante la identificación del latido. Un sistema de clasificación de latidos puede identificar alteraciones cardíacas que pasan desapercibidas en evaluaciones médicas. Esta tecnología puede mejorar la eficiencia diagnóstica en hospitales, centros de salud rurales y dispositivos de monitoreo personal. Además, facilita el análisis de grandes volúmenes de datos EKG sin sobrecargar al personal médico, y puede integrarse en herramientas preventivas para la reducción de eventos cardiovasculares graves[3].

1.4 Objetivos

1.4.1 Objetivo general

Implementar un sistema de clasificación de latidos cardíacos a partir de registros públicos de EKG, utilizando redes neuronales y la clasificación AAMI como referencia para el diagnóstico.

1.4.2 Objetivos específicos

1. Preprocesar señales cardíacas de la base pública para entrenamiento del sistema.
2. Aplicar la clasificación de los latidos según el estándar AAMI.
3. Entrenar una CNN para reconocer la clasificación de latidos de pacientes usando electrocardiogramas.
4. Evaluar el modelo en términos de precisión, sensibilidad y especificidad.
5. Examinar el diagnóstico que ofrece el sistema entrenado por IA.

1.5 Marco teórico

El corazón es un órgano muscular que constituye el núcleo del sistema circulatorio, impulsando la sangre a través de una red vascular. La sangre sin oxígeno se dirige a la aurícula derecha por las venas cavales hacia el ventrículo derecho. El ventrículo derecho se contrae y envía la sangre a la arteria pulmonar, que se bifurca en dos arterias, uno para el pulmón derecho e izquierdo respectivamente. Una vez que, en los pulmones se oxigena la sangre, esta se dirige a la aurícula izquierda que se contrae y pasa al ventrículo izquierdo. Al contraerse bombea la sangre a través de la aorta hacia el cuerpo [4].

El sistema de conducción cardíaca controla la frecuencia y el ritmo de latidos, por lo que cada latido es una señal eléctrica. El nodo sinoauricular (SA) genera un impulso eléctrico, que viaja hacia el marcapasos nódulo auriculoventricular (AV), haciendo que la señal se retrase por un breve instante y que las aurículas se contraigan una fracción de segundo, dándole tiempo que fluya la sangre a los ventrículos. La señal eléctrica continua

hacia el haz de His para que se contraigan los ventrículos derecho e izquierdo y bombear la sangre fuera del corazón, volviendo de nuevo el proceso [5].

El Sistema Eléctrico del Corazón

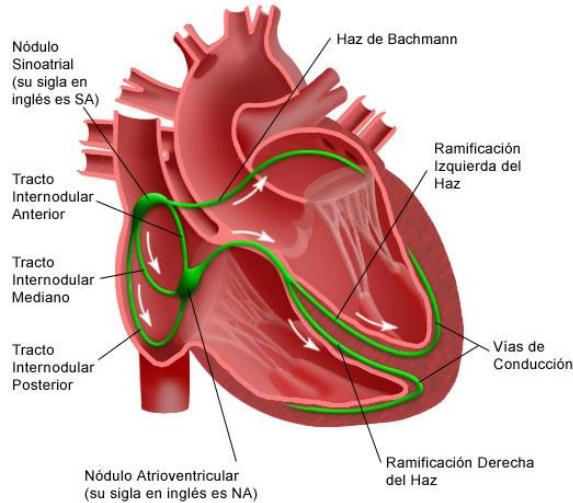


Fig 1.1. Sistema eléctrico del corazón [5].

Las alteraciones en esta dinámica eléctrica pueden desencadenar arritmias o anomalías, cuya detección rigurosa y oportuna resulta imperativa para prevenir eventos cardiovasculares, de consecuencias devastadoras.

Los electrocardiogramas (EKG) registran la actividad eléctrica cardíaca mediante electrodos cutáneos, generando una señal definida por componentes específicos: la onda P (despolarización/contracción auricular), el complejo QRS (despolarización /contracción ventricular) en el cual la onda Q indica la despolarización inicial del tabique interventricular, onda R despolarización de la masa de los ventrículos y la onda S la despolarización final de los ventrículos y la onda T (repolarización ventricular) [6].

Los intervalos y segmentos son relevantes para la interpretación de un EKG: el intervalo PR, tiempo que tarda la señal eléctrica en viajar desde la aurícula hasta los ventrículos, intervalo QT, tiempo total de la actividad eléctrica ventricular desde la despolarización hasta la repolarización, intervalo RR, tiempo entre dos complejos QRS consecutivos para calcular la frecuencia cardíaca, segmento ST, línea isoelectrica.

Desde una perspectiva técnica, un electrocardiógrafo tiene varias etapas para el procesamiento de la señal:

1. Conexión equipo al paciente: Se conecta los electrodos al paciente, generalmente están compuesto de Cloruro de Plata (AgCl).
2. La señal del EKG de baja amplitud: Se recibe a través de un amplificador buffer de alta impedancia, evitando que el contacto piel-electrodo se vea cargado por el circuito y mantiene estable la señal.
3. Amplificación diferencia: Se usa un amplificador de instrumentación, capaz de rechazar el ruido de modo común y resaltar pequeñas diferencias de voltaje entre los electrodos, amplificando la señal sin distorsiones significativas.
4. Filtrado analógico previo: Señal pasa por filtros analógicos que eliminan artefactos antes de digitalizarse. El filtro pasa alto (HPF) atenúa la deriva de la línea base (baseline wander), oscilación señal lenta de la línea isoeléctrica. El filtro pasa bajo (LPF) reduce interferencias musculares y de alta frecuencia. El filtro notch, elimina el ruido de la red eléctrica de 50/60 Hz. El filtro anti-aliasing para evitar distorsiones durante la conversión digital
5. Conversión analógico-digital (ADC): Toma muestras de la señal a frecuencias típicas de 250 a 1000Hz con resoluciones de 16 bits o más. Transforma la señal continua en datos digitales, para ser procesados por software.
6. Procesamiento digital: Elimina los artefactos digitales mediante filtros software, el cual se corrige la deriva de la línea base, aplica filtrado de 50/60Hz, reducir ruidos musculares o movimiento.
7. Detección y segmentación de latidos: El sistema identifica automáticamente los picos R, para calcular intervalos y frecuencias cardíacas.

Así, segmenta los intervalos y facilita el análisis clínico de alteraciones cardíacas [7].

La clasificación automatizada de latidos cardíacos se rige estrictamente por el estándar de la AAMI, que los categoriza en cinco clases distintivas: N (normal), S (supraventricular), V (ventricular), F (fusión) y Q (no clasificado). Este marco normativo resulta crucial para el entrenamiento supervisado de algoritmos de inteligencia artificial.

La inteligencia artificial, en particular las redes neuronales profundas como las redes convolucionales (CNN), ha transformado radicalmente el análisis de señales EKG. Estas redes extraen de manera autónoma patrones morfológicos complejos directamente desde la señal cruda, eliminando la dependencia de la ingeniería manual de características. Los modelos basados en deep learning han demostrado precisiones superiores al 98% en la clasificación de arritmias, reduciendo el tiempo de procesamiento en hasta un 70% frente a métodos tradicionales. Además, exhiben una capacidad sobresaliente para detectar anomalías sutiles, incluyendo:

- Alteraciones en el intervalo PR
- Cambios en la morfología del QRS
- Indicadores de fibrilación auricular, taquicardias o bloqueos [8].

La MIT-BIH Arrhythmia Database, un repositorio de prestigio internacional proporciona registros EKG anotados provenientes de pacientes reales, muestreados a 360 Hz y acompañados de etiquetas clínicas en formato atr [8]. Su disponibilidad abierta a través de PhysioNet ha impulsado significativamente la investigación en clasificación de arritmias mediante inteligencia artificial. El desarrollo y la implementación de modelos se benefician de herramientas computacionales de código abierto, tales como Python, TensorFlow, Keras y la librería WFDB, las cuales permiten la carga, preprocesamiento, entrenamiento y validación de redes neuronales en plataformas como Google Colab [9].

Investigaciones clínicas han revelado tasas de error de hasta el 25% en la interpretación manual de EKG, atribuidas a la variabilidad, la presencia de ruido y la fatiga

del personal médico. En contraste, los sistemas automatizados impulsados por inteligencia artificial ofrecen una mejora sustancial en precisión, reducen significativamente los falsos positivos y negativos, y aseguran resultados reproducibles, incluso en entornos clínicos de alta demanda o aplicaciones de telemedicina [10].

Capítulo 2

2.1 Metodología.

En este capítulo se describe la metodología usada para la implementación del sistema de clasificación de latidos usando algoritmos de inteligencia artificial. Tomando como referencia el equipo de EKG del cardiólogo Dr. Rafael Castilla, GE cardiosoft cam-usb A/T, con su adaptador de electrodos, el cual se encarga del procesamiento digital de señales de EKG de manera precisa y segura. Toda su electrónica se encuentra integrada en una única placa, dividida en tres bloques funcionales: procesamiento de EKG, controlador y puerto de PC.

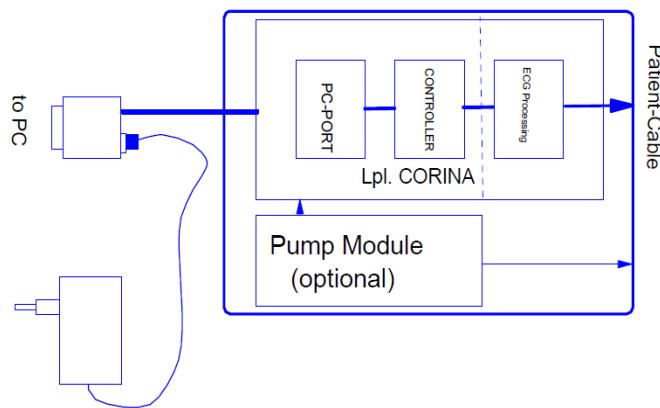


Fig 2.1. GE cardiosoft cam-usb A/T.

En la etapa de procesamiento de EKG, las señales bioeléctricas captadas por hasta 11 electrodos pasan por un circuito de protección y un preamplificador de alta impedancia, que evita cargar al paciente y mejora la sensibilidad. Luego, estas señales son acondicionadas, amplificadas y convertidas a digital mediante ASICs dedicados. Esta sección también realiza funciones como detección de marcapasos, verificación de señal y medición de voltaje de CC. Los datos digitalizados se transfieren al controlador mediante una interfaz serial QSPI.

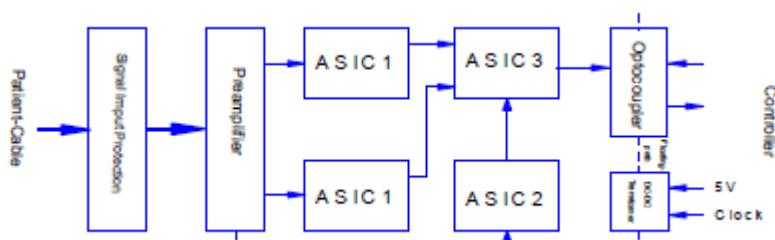


Fig 2.2. Procesamiento EKG.

El controlador, basado en el microcontrolador, gestiona toda la lógica del sistema. Controla la comunicación entre el procesador de EKG, el PC y módulos opcionales como el CORINA. Integra memoria FLASH programable, temporizadores, manejo de interrupciones y buses internos.

El módulo CORINA del sistema GE CardioSoft se encarga de procesar electrónicamente la señal de EKG captada desde el paciente. El proceso inicia con la captación de señales bioeléctricas a través de los electrodos que detectan mV generados por la actividad eléctrica del corazón, normalmente en el rango de mV.

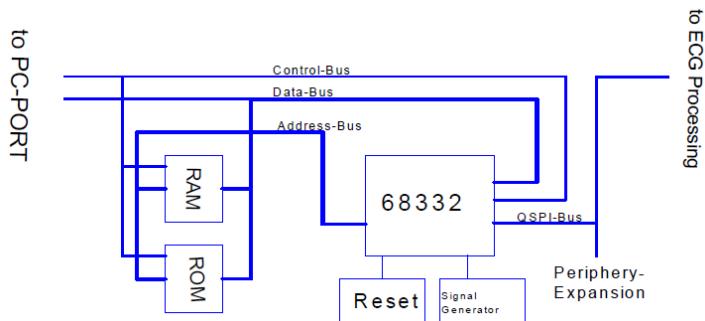


Fig 2.3. Controlador

Estas señales pasan primero por un amplificador diferencial de alta impedancia (>50 MΩ), lo que asegura que no se genere carga sobre el paciente y se mantenga una buena sensibilidad. Luego, se aplica un rechazo de modo común (CMRR) para eliminar interferencias típicas del entorno, como el ruido de 50 o 60 Hz.

Una etapa clave es la supresión de los pulsos de marcapasos, que podrían distorsionar el trazado del EKG. Esta función se realiza mediante el canal F. Por ejemplo, una señal de entrada de 1 mV en una salida de 1 V.

Posteriormente, se aplicó un filtrado pasa bandas, que elimina las frecuencias no deseadas: un filtro pasa bajo que corta a 150 Hz (evita interferencias musculares) y un pasa alto a 0,08 Hz (corrige la línea base).

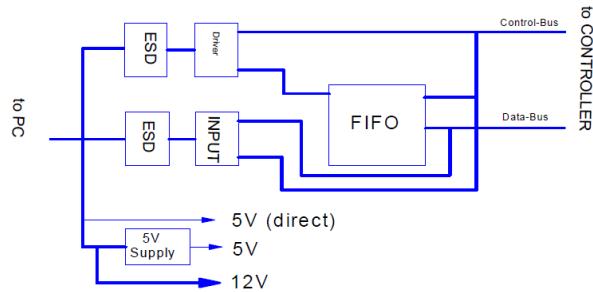


Fig 2.4. Transferencia de señal a PC.

La seguridad del paciente está garantizada con un diseño de aislamiento galvánico, corriente de fuga limitada ($<10 \mu\text{A}$). Finalmente, la señal acondicionada se envía como salida analógica aislada, con 1 V por cada 1 mV de entrada.

Finalmente, el puerto de PC permite la alimentación eléctrica del sistema a través de un transductor DC/DC y protege la comunicación mediante circuitos contra descargas electrostáticas. La transmisión de datos al software CardioSoft se realiza mediante una memoria FIFO, que garantiza una transferencia ordenada y sin pérdidas. El electrocardiógrafo genera el examen EKG en formato pdf, que posteriormente se lo imprime en una impresora convencional usando papel bond.

Siendo este, un ejemplo de un electrocardiógrafo y analizando electrónicamente el proceso de adquisición del EKG, se procedió con la carga de datos a nuestro sistema, utilizando un conjunto de 48 registros de electrocardiogramas tomadas a una frecuencia de muestreo de 360 Hz por cada canal, la escala de la amplitud de la señal está en el rango de los mV, provienen de la base de datos pública del MIT.

El desarrollo del proyecto siguió un diseño con una secuencia lógica y cronológica, este enfoque permitió establecer cada etapa del proyecto de manera ordenada, desde la carga de datos de las señales eléctricas del corazón de los respectivos pacientes hasta la evaluación del modelo entrenado con el respectivo diagnóstico de la clase de latido que posee el paciente.

A continuación, se describe en detalle los pasos realizados para la elaboración del proyecto (figura 2.5):

- Carga de base de datos MIT Arrhythmia.
- Aplicación del filtro pasa banda (eliminación de ruido).
- Normalización de la señal EKG.
- Segmentación de latidos.
- Balanceo de datos usando SMOTE.
- Entrenamiento del modelo.
- Validación con nuevos datos de EKG.
- Evaluación del modelo entrenado.
- Diagnóstico.

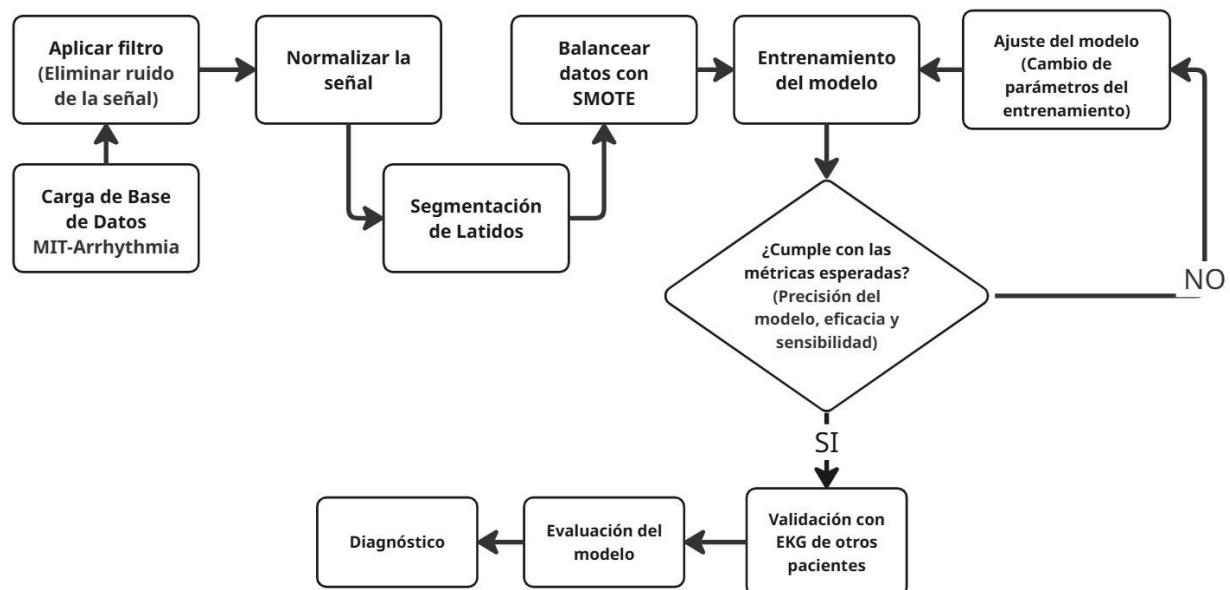


Fig 2.5. Metodología para implementación del sistema.

Para facilitar la explicación de la metodología propuesta, resumimos la estructura en carga de la base de datos, preprocesamiento de la señal, entrenamiento del modelo y evaluación del modelo.

2.2 Carga de la base de datos

En esta primera sección procedemos con la carga de la base datos MIT-Arrhythmia y existen 2 maneras de realizarlo: descargando directamente el archivo desde el sitio web PhysioNet o mediante comando en Visual Studio Code usando la librería wfdb.

Para este caso hemos importado mediante código de la librería wfdb, ya que con este método es posible obtener de manera clara y precisa los archivos EKG de los pacientes. El otro método en cambio posee directorios html, archivos xws y otros archivos que son en este caso poco relevantes para el desarrollo del modelo.

Una vez se obtuvo la carpeta con los archivos a usar se procede a convertir los archivos dat, hea y atr, en archivos csv para facilitar la visualización de los datos EKG y su respectivo procesamiento en la siguiente etapa de desarrollo, esto se lo realiza con la función presente en la figura 2.6.

```
def guardar_ecg_csv(paciente_id, carpeta_salida="mitdb_csv", ruta_lectura="mitdb"):
    try:
        registro = wfdb.rdrecord(os.path.join(ruta_lectura, paciente_id))
        derivaciones = registro.sig_name
        datos_ecg = registro.p_signal

        if not os.path.exists(carpeta_salida):
            os.makedirs(carpeta_salida)

        nombre_archivo = os.path.join(carpeta_salida, f"{paciente_id}_ecg.csv")

        with open(nombre_archivo, mode='w', newline='') as archivo:
            escritor_csv = csv.writer(archivo)
            encabezado = ['indice'] + derivaciones + ['symbol']
            escritor_csv.writerow(encabezado)

            for i, fila in enumerate(datos_ecg):
                fila_csv = [i] + list(fila) + ['']
                escritor_csv.writerow(fila_csv)

            print(f"Archivo guardado: {nombre_archivo}")
    except Exception as e:
        print(f"Error con paciente {paciente_id}: {e}")
```

Fig 2.6. Función de guardado para EKG en formato CSV.

2.3 Preprocesamiento de la señal

En esta sección después de la obtención de los archivos CSV se requiere eliminar el ruido presente en las señales EKG, en electrónica se usan los filtros pasa banda para la obtención de la señal eléctricas del corazón y los filtros notch se usan para eliminar el ruido presente en la señal por efecto de las señales electromagnéticas de otros equipos, sus propios componentes o por problemas de diseño y su efectividad dependerá del orden del filtro, generalmente para equipos EKG se usan de orden 4 en adelante.

De esta manera se empleó un equivalente de los filtros notch y esta es la transformada de wavelet, que funciona como un filtro base para el análisis de la señal y dependiendo de su configuración puede eliminar la mayoría de las variaciones que tiene la señal EKG, la figura 2.7 muestra la implementación de la transformada de wavelet en una función para eliminar el ruido de la señal cardíaca.

```
def eliminar_ruido(signal, wavelet='db4', level=4):
    coeffs = pywt.wavedec(signal, wavelet, level=level)
    sigma = (1/0.6745) * np.median(np.abs(coeffs[-level]))
    uthresh = sigma * np.sqrt(2 * np.log(len(signal)))
    coeffs_thresh = [pywt.threshold(c, value=uthresh, mode='soft') if i > 0 else c for i, c in enumerate(coeffs)]
    return pywt.waverec(coeffs_thresh, wavelet)
```

Fig 2.7. Función para eliminar ruido de una señal.

Una vez limpia la señal, se guardaron los datos en variables y se procedió con la segmentación de los latidos del EKG de cada paciente, para este paso se requiere definir una ventana de datos que es lo que define al latido como tal.

En la figura 2.8 se muestra cómo se realizó la segmentación de latidos que depende de definir una ventana_antes y una ventana_despues, este será el segmento de datos que representa el latido.

```
for i, pico in enumerate(ann_indices):
    inicio = pico - ventana_antes
    fin = pico + ventana_despues
    if inicio < 0 or fin > len(ecg_array):
        continue
    latido = ecg_array[inicio:fin, :]
```

Fig 2.8. Segmentación de latidos.

Para estar seguros de que hemos segmentado bien los latidos graficamos cada grupo de latidos, pero antes hemos normalizado los latidos para que tengan una escala estandarizada como se muestra en la figura 2.9.

```
# Normalización por canal (min-max)
latido_norm = (latido_filtrado - latido_filtrado.min(axis=0)) / (latido_filtrado.max(axis=0) - latido_filtrado.min(axis=0) + 1e-6)
```

Fig 2.9. Normalización de latidos.

Ahora observamos si los latidos fueron correctamente segmentados y normalizados, para este paso graficamos los latidos de acuerdo con el grupo perteneciente (figura 2.10), guiándonos por las anotaciones de las señales cardíacas presente en el archivo CSV o en el caso que se use la librería wfdb, en los archivos atr.

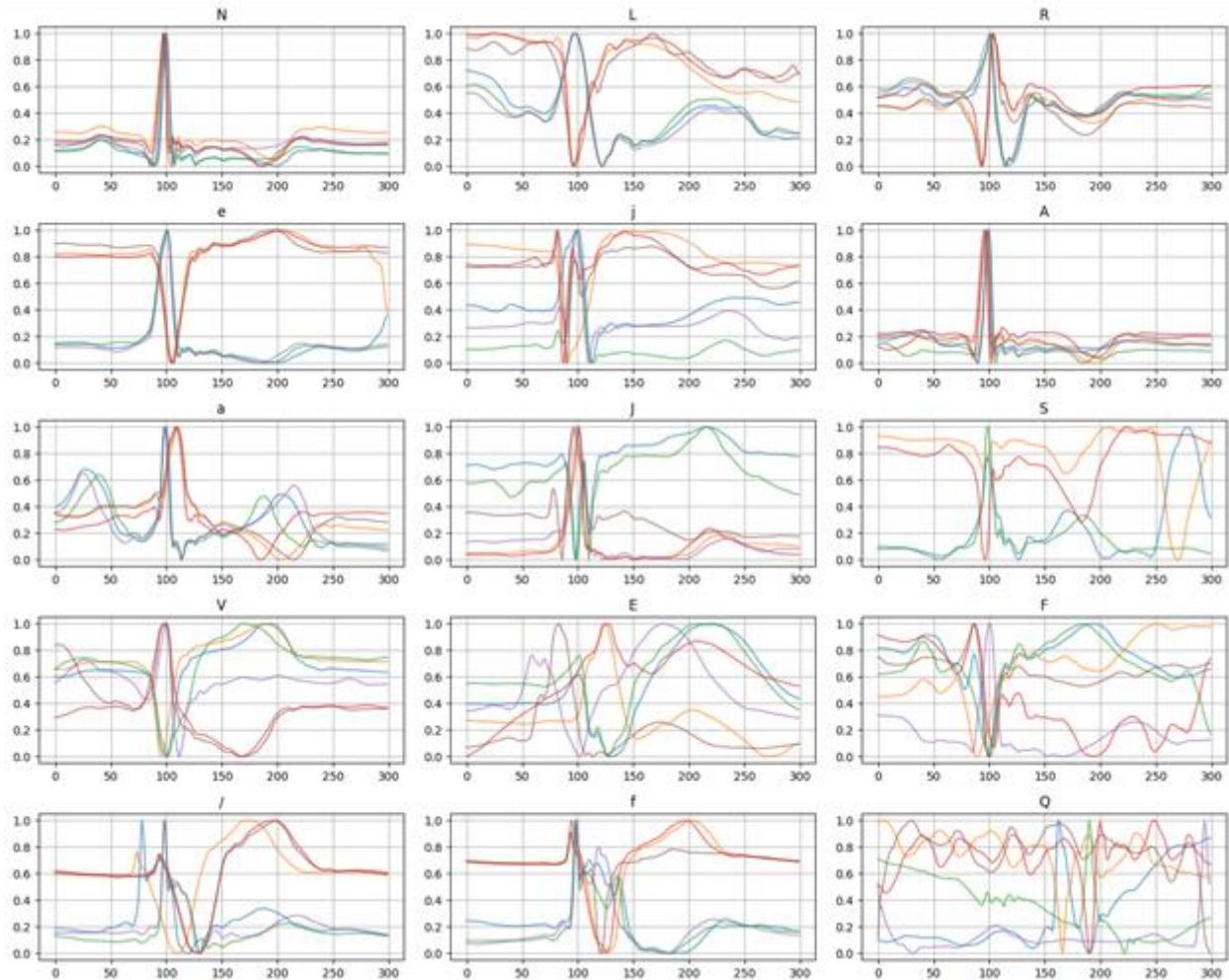


Fig 2.10. Gráficas de latidos por clase.

Nuestro siguiente paso fue clasificar los latidos en 5 tipos fundamentales: Normal (N), Supraventricular (S), Ventricular (V), Fusión (F) y Desconocidos (Q). Estas serán las clases que se usarán para el entrenamiento del modelo, la cantidad de datos por cada de estas clases se presentan en la figura 2.11.

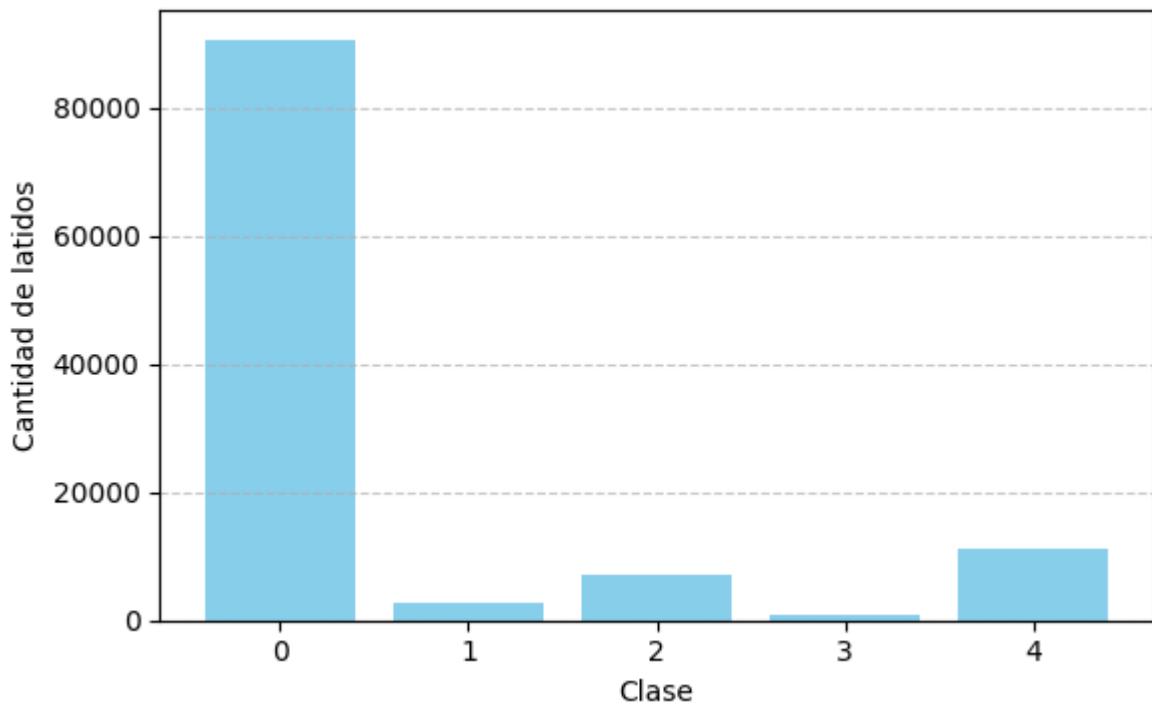


Fig 2.11. Distribución de datos por clase

Como se muestra en la figura anterior la distribución por clases fue muy desbalanceada, tal que la clase normal supera los 80000 datos, si entrenamos el modelo con esta distribución existirán sesgos durante el entrenamiento, afectando el diagnóstico por predicciones equivocadas del modelo.

Por esto se requirió de la creación de un dataframe para balancear los latidos a 20000 muestras por clase (figura 2.12) usando SMOTE, este número puede variar dependiendo de los requerimientos del modelo a entrenar, cuando los datos se guardaron procedimos con el almacenamiento de los datos balanceados en formatos csv y npz, esto facilitará su uso para el entrenamiento del modelo.

```

num_muestras, longitud, canales = X.shape
X_flat = X.reshape((num_muestras, -1))
smote = SMOTE(sampling_strategy='auto', random_state=42, k_neighbors=1)
X_smote, Y_smote = smote.fit_resample(X_flat, Y_5)

print(f"Después de SMOTE: {X_smote.shape}, {Y_smote.shape}")

# =====
# 5. Ajuste final: undersampling a 20,000 muestras por clase
# =====
target = 20000
df_smote = pd.DataFrame(X_smote)
df_smote["label"] = Y_smote

df_final = []
for clase in sorted(df_smote["label"].unique()):
    df_clase = df_smote[df_smote["label"] == clase]
    if len(df_clase) > target:
        df_bal = resample(df_clase, replace=False, n_samples=target, random_state=42)
    else:
        df_bal = df_clase # Si justo tiene 20k, no tocar
    df_final.append(df_bal)

```

Fig 2.12. Balance con SMOTE y ajuste de datos al número deseado.

Con los datos cuya distribución fue balanceada, son los óptimos que sirven para finalmente proceder con el entrenamiento del modelo.

2.4 Entrenamiento del modelo

Se procedió con la división de los datos para entrenamiento y validación, un 80% para entrenamiento y 20% para validación es un valor típico para el aprendizaje del modelo.

```

# División en entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X, y_cat, test_size=0.2, random_state=42, stratify=y_cat)

```

Fig 2.13. División de datos para entrenamiento y validación

Usando tensorflow se emplea como método de entrenamiento una red neuronal convolucionada (CNN) de una dimensión (1D) implementada en Keras, este método es ideal para el procesamiento de señales temporales 1D como lo son los EKG ya que consta de

varias capas diseñadas para extraer las características más importantes de los latidos del corazón, usando el comando Sequential hemos definido la arquitectura del modelo:

- Convid1D: Estas capas se encargan de la extracción de características de la señal EKG, se definen la cantidad de filtros a usar para cada capa, el tamaño del kernel y si usa la función ReLU para la mejora de convergencia del modelo.
- Batch Normalization: Para acelerar y estabilizar el proceso de entrenamiento se aplica la normalización de lotes después de cada capa.
- MaxPooling1D: Para reducir el tiempo de entrenamiento y el riesgo de sobreajuste se empleó un maxpooling para reducir la dimensionalidad de las características extraídas del EKG, para este caso se dejó una ventana de 2.
- Flatten: Las capas densas requieren entradas de una sola dimensión así que se requiere aplanar características 3D obtenidas de las capas convolucionadas anteriores.
- Dense: Son capas densas que se incluyen después de las capas convolucionales, para este entrenamiento se aplicó una capa de 128 neuronas, esto permite capturar patrones complejos de las señales.
- Dropout: Previene el sobreajuste del modelo al desactivar aleatoriamente un porcentaje de las neuronas de la capa densa anterior por cada iteración durante el entrenamiento, en la arquitectura se define un 0.3 (30%) de estas neuronas para su desactivación.

```

modelo = Sequential([
    Conv1D(32, 5, activation='relu', input_shape=(X.shape[1], 1)),
    BatchNormalization(),
    MaxPooling1D(2),

    Conv1D(64, 3, activation='relu'),
    BatchNormalization(),
    MaxPooling1D(2),

    Flatten(),
    Dense(128, activation='relu'),
    Dropout(0.3),
    Dense(5, activation='softmax') # 5 clases
])

modelo.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
modelo.summary()

```

Fig 2.14. Arquitectura del modelo.

Para empezar con el entrenamiento se empleó la función fit(), definimos una cantidad de 10 épocas para el entrenamiento, podemos reducir o aumentar el número épocas si verificamos que el modelo no sufra de un sobreajuste y tengamos los recursos computacionales necesarios para dicho entrenamiento, además aquí se define el argumento validation_split con un valor de 0.2 que corresponde al 20% para validación del modelo y por ende un 80% será para el entrenamiento.

```

history = modelo.fit(
    X_train, y_train,
    validation_split=0.2,
    epochs=10,
    batch_size=64,
    verbose=1
)

```

Fig 2.15. Entrenamiento del modelo.

```
modelo.save("modelo_cnn_20000.h5")
```

Fig 2.16. Guardado del modelo

2.5 Evaluación y prueba del modelo

Para evaluar la fiabilidad del modelo se visualizó los resultados de este mediante gráficos que muestran la evolución de la precisión y pérdidas a lo largo de las épocas durante el entrenamiento, en la figura 2.17 se muestra el método para visualizar la precisión y perdidas del modelo durante su entrenamiento. Se evalúa si el modelo tiene problemas de sobreajuste o si no cumple con los estándares de precisión deseados para el sistema de clasificación de latidos mediante algoritmos de inteligencia artificial.

```
# Evaluación
loss, acc = modelo.evaluate(X_test, y_test)
print(f"Precisión en test: {acc:.4f}")
```

Fig 2.17. Método para obtención de precisión y pérdidas del modelo.

Para tener más resultados del modelo se requirió emplear métodos para obtener la matriz de confusión del modelo y las métricas promedio que mostraran precisión, sensibilidad y eficacia.

```
# Matriz de confusión
cm = confusion_matrix(y_true, y_pred)
cm_percent = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis] * 100 # porcentajes
```

Fig 2.18. Implementación de la matriz de confusión

```
# Reporte de clasificación detallado (precision, recall, f1-score)
print(classification_report(y_true, y_pred, target_names=['N', 'S', 'V', 'F', 'Q']))
```

Fig 2.19. Obtención del reporte de métricas del modelo por clase.

La prueba final del modelo se realizó con datos EKG de pacientes de fuentes externas, esta señal debe pasar por el proceso de preprocesamiento que hemos realizado para el entrenamiento, ya que el modelo entrenado requiere entradas de datos cuya dimensión sea iguales a los datos con los que realizó el aprendizaje.

Definimos los parámetros para interpretar los resultados arrojados por el sistema de clasificación ya que las clases de latidos son interpretadas por el modelo como clase 0 (N), clase

1 (S), clase 2 (V), clase 3 (F) y clase 4 (Q), con el modelo cargado que ya fue guardado previamente como ekg_cnn_model.h5 hacemos uso de este, el resultado es la señal EKG del paciente con el diagnóstico del latido en su pico R.

```
# ===== Parámetros =====
ventana_antes = 99
ventana_despues = 201
mapa_5clases = {0: 'N', 1: 'S', 2: 'V', 3: 'F', 4: 'Q'}
colores_clases = {'N': 'green', 'S': 'orange', 'V': 'red', 'F': 'purple', 'Q': 'blue'}
```

Fig 2.20. Parámetros para interpretación de resultados del modelo.

```
# ===== Cargar modelo =====
modelo = load_model("modelo_cnn_20000.h5")
```

Fig 2.21. Carga de modelo

Capítulo 3

3. Resultados y análisis

3.1 Precisión y pérdida del entrenamiento

Se obtuvo el comportamiento de los datos de entrenamiento en comparación con los datos de validación durante las 10 épocas que dura el aprendizaje del modelo como se muestra en la figura 3.1.

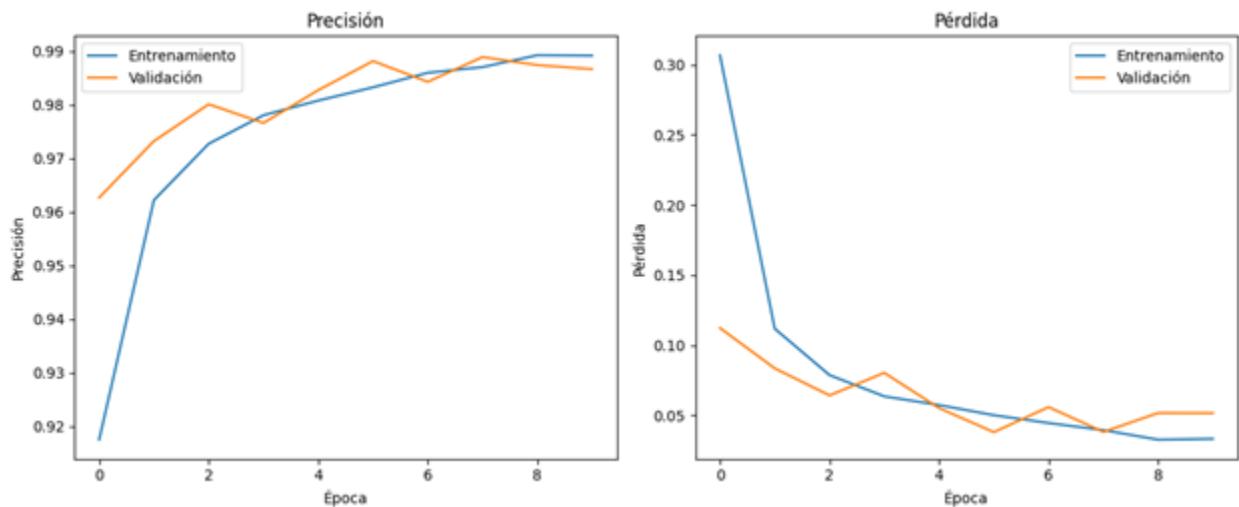


Fig 3.1. Gráfica de precisión y pérdida durante el entrenamiento.

El comportamiento de los datos se puede evaluar en términos de precisión y pérdida, la precisión muestra una proporción de predicciones correctas sobre las realizadas por el modelo, esto quiere decir que el modelo identificó el 98.64% de los datos como correctos.

La pérdida muestra el error que se obtuvieron entre los datos reales y las predicciones del modelo, en este caso la perdida fue inferior al 10%.

3.2 Matriz de confusión

La matriz de confusión es una herramienta utilizada en estadística y aprendizaje automático para evaluar el rendimiento de un modelo de clasificación. Se representa como una tabla que muestra la relación entre las predicciones realizadas por el modelo y las clases reales de los datos. Cada fila de la matriz corresponde a la clase real, mientras que cada columna representa la clase predicha, permitiendo visualizar los aciertos y errores del modelo.

El total de datos utilizados para el entrenamiento es de 16.000 por cada clase, de esta manera se empleó 4000 datos por cada clase para la validación del modelo durante el aprendizaje y pueden ser interpretadas de la siguiente manera de acuerdo con la figura 3.2:

- Para la clase 0 que corresponde a los latidos normales (N), se está prediciendo de manera correcta 3912 datos.
- Para la clase 1 que corresponde a los latidos supraventriculares (S), se está prediciendo de manera correcta 3926 datos.
- Para la clase 2 que corresponde a los latidos ventriculares (V), se está prediciendo de manera correcta 3941 datos.
- Para la clase 3 que corresponde a los latidos fusión (F), se está prediciendo de manera correcta 3983 datos.
- Para la clase 4 que corresponde a los latidos desconocidos (Q), se está prediciendo de manera correcta 3965 datos.

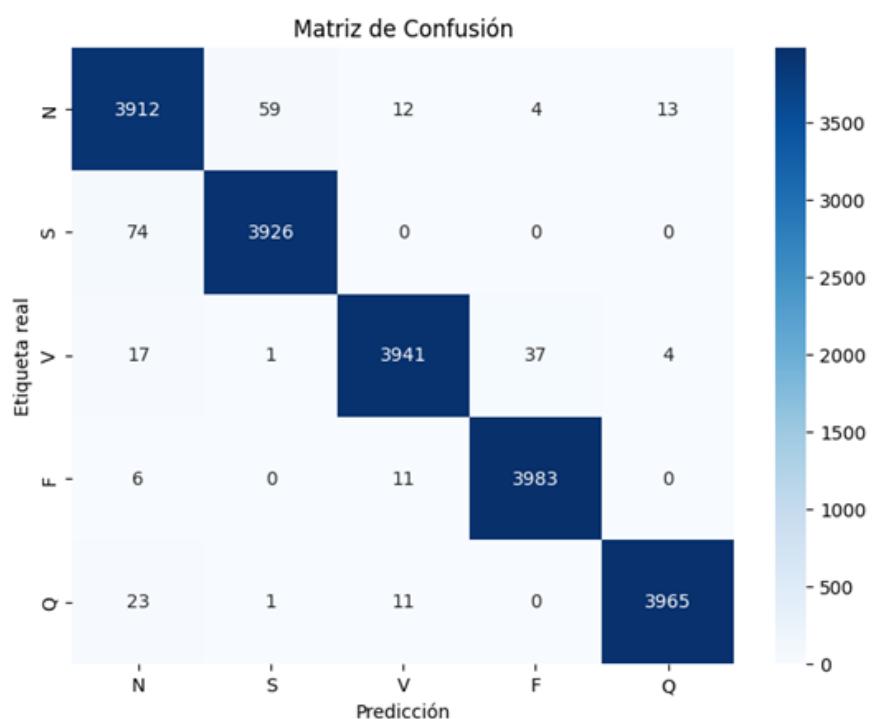


Fig 3.2. Matriz de confusión

Para una mayor comprensión de la matriz de confusión se observó en porcentajes la cantidad de aciertos del modelo durante el entrenamiento. En la figura 3.3 se visualiza dichos resultados de la siguiente manera:

- 97.8% de manera correcta está prediciendo la clase N.
- 98.2% de manera correcta está prediciendo la clase S.
- 98.5% de manera correcta está prediciendo la clase V.
- 99.6% de manera correcta está prediciendo la clase F.
- 99.1% de manera correcta está prediciendo la clase Q.

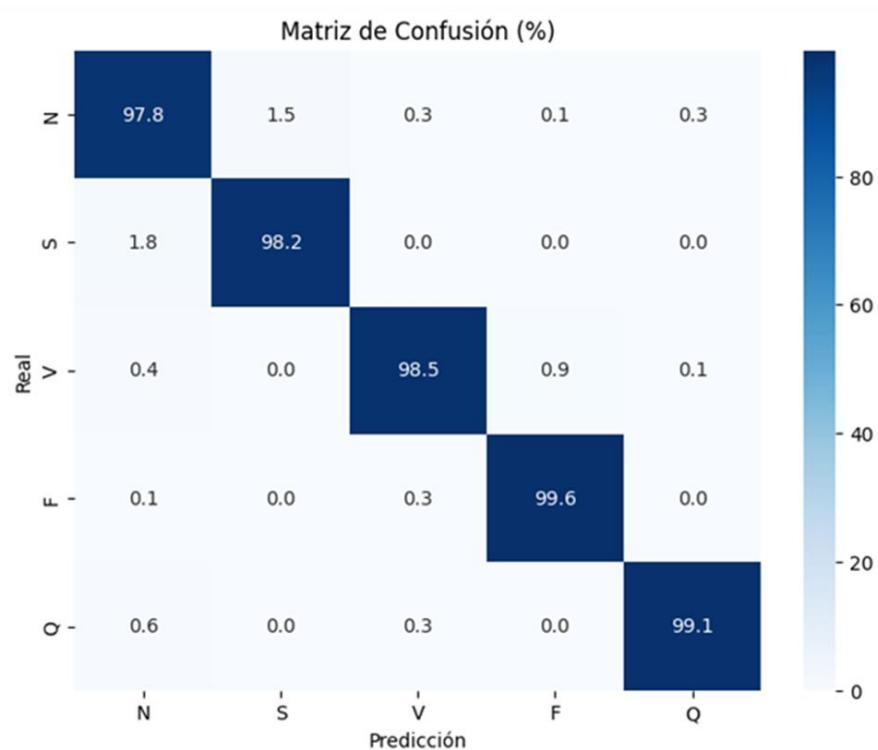


Fig 3.3. Matriz de confusión porcentual

3.3 Métricas generales

En la tabla 1, se muestra la evaluación promedio obtenida del modelo después del entrenamiento que define su rendimiento para la clasificación de latidos.

Tabla 3.1 Métricas de modelo

Métricas promedio del modelo	
Precisión	98.64 %
Recall	98.63%
F1-Score	98.64%

El modelo evaluado mostró una precisión del 98.64%, lo que indica que el 98.64% de las instancias clasificadas como verdaderos positivos fueron correctas. El recall muestra la sensibilidad del modelo para identificar casos positivos que fue del 98.63%. Por último, el F1-Score refleja la eficacia del modelo para detectar los verdaderos positivos mientras minimiza los falsos positivos, su valor fue del 98.64%.

3.4 Prueba con EKG de otros pacientes

En esta sección se muestran los resultados obtenidos durante la prueba del sistema de clasificación de EKG entrenado con la base de datos pública del MIT-BIH Arrhythmia Database. El modelo fue probado con varios EKG que no fueron usados durante el entrenamiento del modelo con el objetivo de validar la capacidad de clasificación de los diferentes latidos cardíacos.

La figura 3.4 representa una señal de EKG de un paciente con la clasificación de los 20 primeros latidos con la respectiva clasificación de cada latido del EKG de acuerdo con la clasificación AAMI como diagnóstico. Se empleó marcadores en el pico R de los latidos que señala que clase de latido tiene el paciente, de color verde para la clase normal (N) y de color naranja para la clase supraventricular (S), en este caso 19 de 20 latidos son de clase normal.

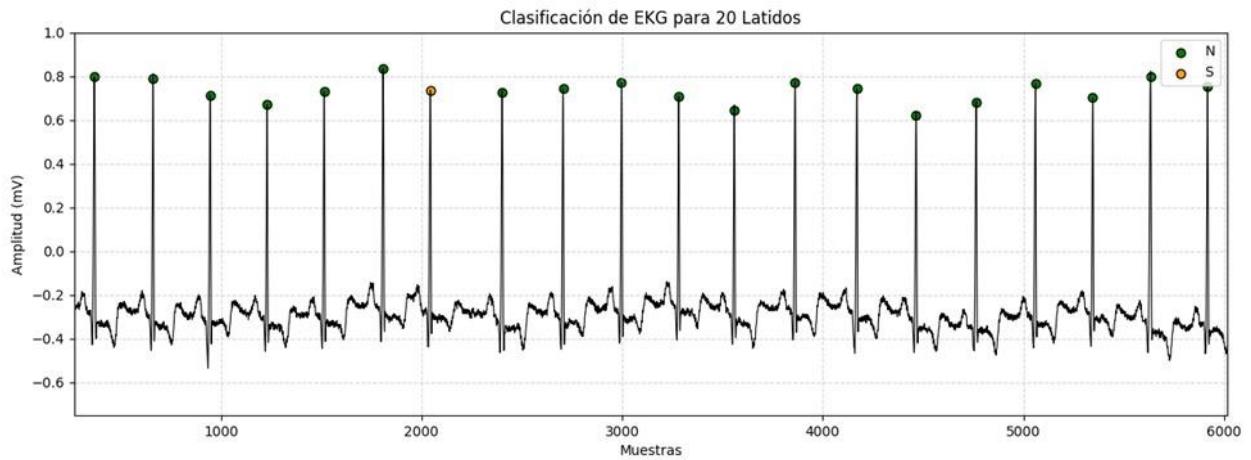


Fig 3.4. Señal EKG de paciente sano.

La figura 3.5 muestra una señal EKG con mayor variabilidad de clase como verde para normales, rojo para ventriculares y morado para fusión. En esta señal hay una mayor cantidad de anomalías que la mostrada en la figura 3.4, pero muestra la capacidad del sistema para detectar más clases de latidos.

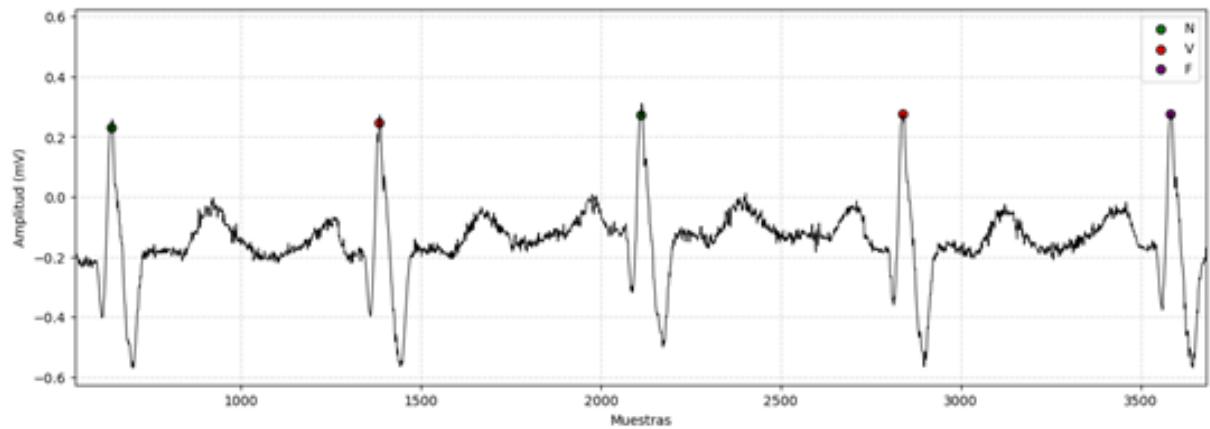


Fig 3.5. Señal EKG de paciente con anomalías cardíacas.

3.5 Análisis de resultados

En base a la matriz de confusión porcentual obtenida se puede observar un excelente desempeño del modelo de clasificación de EKG, la clase N alcanza el 97.8% con un nivel de confusión relativamente bajo, alrededor del 1.5% de confusión con la clase S y menos del 1% para las clases V, F y Q.

La clase S muestra una precisión del 98.2%, con una confusión del 1.8% para la clase N, para las otras clases no muestra confusiones y este resultado muestra que el modelo distingue con un buen índice de aciertos los latidos supraventriculares que las otras clases, reduciendo en gran medida que se pase por alto arritmias por latidos supraventriculares.

La clase V alcanzo un porcentaje del 98.5% de aciertos, con confusiones pequeñas del 0.4% con la clase N, 0.9% con la clase F y 0.1% con la clase Q, este nivel de aciertos es óptimo para la detección temprana de alteraciones ventriculares.

La clase F alcanzo un porcentaje del 99.6% de aciertos con confusiones del 0.1% con la clase N y 0.3% para la clase V, esto muestra que la clase F es la que tiene mayor nivel de precisión por lo que se sugiere que el modelo clasifica mejor esta clase de latidos.

La clase Q obtuvo una precisión del 99.1% con confusiones del 0.6% con la clase N y del 0.3% con la clase V, esto confirma la capacidad del modelo para detectar latidos atípicos que están fuera de las categorías principales.

Los resultados gráficos de los EKG clasificados refuerzan los resultados obtenidos en la matriz de confusión, en la figura 3.4 que corresponde a un paciente relativamente sano, se observó que la mayoría de los latidos fueron categorizados como normales (N), validando de esta manera que no existen falsos positivos significativos.

En la figura 3.5 corresponde a un paciente con anomalías cardíacas, pues el modelo identifico varios tipos de latidos (N, S, V, F) en la misma señal, esta clasificación demuestra que el sistema es consistente con la detección de cada latido.

Finalmente, se puede concluir que tanto la matriz de confusión como los resultados de la clasificación de EKG validan el sistema de clasificación como una herramienta de diagnóstico asistido para el personal médico capacitado, integración en entornos clínicos y de telemedicina para el monitoreo de señales cardíacas.

Capítulo 4

4. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- El sistema implementado basado en redes neuronales convoluciones (CNN) alcanzó, en la clasificación de latidos cardíacos, una precisión promedio equivalente al 98.64%, en las clases evaluadas. Confirmando la viabilidad de la IA como herramienta confiable para la detección de arritmias en electrocardiogramas.
- La baja confusión porcentual entre las clases confirma que el proceso de: filtrado, normalización, segmentación y balanceo de datos mediante SMOTE, optimizó la calidad de las señales y permitió al modelo distinguir patrones con alta precisión.
- Los resultados obtenidos evidencia que el sistema puede integrarse como una herramienta de soporte en entornos hospitalarios, de telemedicina o en dispositivos portátiles de monitoreo cardíaco.
- El sistema contribuye a disminuir el tiempo de interpretación de registros EKG y reducir el riesgo de errores humanos, lo cual aumenta eficiencia diagnóstica en centros de salud.

4.2 Recomendaciones

- Se recomienda obtener más datos de EKG externos para validación y prueba del modelo, pues es posible que valores tan altos de precisión se deban a que la etapa de validación se la realice con la misma base de datos.
- Se debe evaluar el uso del sistema de clasificación en conjunto con los programas de los electrocardiógrafos, de esta manera puede ser posible un diagnóstico en tiempo real.
- El modelo puede ser embebido en una Raspberry o una FPGA, de esta manera se puede analizar su uso comercial como una herramienta portátil para el personal médico y aplicaciones de telemedicina.

Referencias

- [1] World Heart Federation, Deaths from cardiovascular disease surged 60% globally over the last 30 years: Report, World Heart Federation. [En línea]. Disponible en: www.world-heart-federation.org/news/deaths-from-cardiovascular-disease-surged-60-globally-over-the-last-30-years-report/
- [2] J. De Bie, C. Martignani, G. Massaro, y I. Diemberger, Performance of seven ECG interpretation programs in identifying arrhythmia and acute cardiovascular syndrome, vol. 58, pp. 143-149, 2020, doi: 10.1016/j.jelectrocard.2019.11.043.
- [3] J. Bailey et al., Recommendations for standardization and specifications in automated electrocardiography: bandwidth and digital signal processing. A report for health professionals by an ad hoc writing group of the Committee on Electrocardiography and Cardiac Electrophysiology of the Council on Clinical Cardiology, American Heart Association, vol. 81, n.o 2, pp. 730-739, feb. 1990, doi: 10.1161/01.CIR.81.2.730.
- [4] National Heart, Lung, and Blood Institute, How the Heart Works, National Heart, Lung, and Blood Institute. Accedido: 17 de agosto de 2025. [En línea]. Disponible en: <https://www.nhlbi.nih.gov/es/salud/corazon/latidos-cardiacos>.
- [5] Stanford Medicine Children's Health, Anatomy and Function of the Electrical System, Stanford Medicine Children's Health. Accedido: 17 de agosto de 2025. [En línea]. Disponible en: <https://www.stanfordchildrens.org/es/topic/default?id=anatomy-and-function-of-the-electrical-system-90-P04865>
- [6] R. M. Rangayyan y S. Krishnan, Biomedical Signal Analysis. John Wiley & Sons, 2024.
- [7] GE HealthCare, A Guide to ECG Signal Filtering. [En línea]. Disponible en: www.gehealthcare.co.uk/insights/article/a-guide-to-ecg-signal-filtering
- [8] A. Darmawahyuni et al., Deep learning-based electrocardiogram rhythm and beat features for heart abnormality classification, PeerJ Comput. Sci., vol. 8, p. e825, ene. 2022, doi: 10.7717/peerj-cs.825.
- [9] M. Sermesant, H. Delingette, H. Cochet, P. Jaïs, y N. Ayache, Applications of artificial intelligence in cardiovascular imaging, Nat. Rev. Cardiol., vol. 18, n.o 8, pp. 600-609, ago. 2021, doi: 10.1038/s41569-021-00527-2.
- [10] H. Lu et al., Métodos computacionales y matemáticos en medicina, Wiley Online Library. [En línea]. Disponible en: www.onlinelibrary.wiley.com/doi/10.1155/2022/3016532