



**UNIVERSIDAD DE GUANAJUATO**

---

---

**CAMPUS IRAPUATO - SALAMANCA  
DIVISIÓN DE INGENIERÍAS**

**“CLASIFICACIÓN DE SEÑALES DE  
ELECTROCARDIOGRAFÍA (ECG)  
UTILIZANDO REDES NEURONALES  
PULSANTES (SNN)”**

**TESIS**

QUE PARA OBTENER EL GRADO DE:  
**MAESTRO EN INGENIERÍA ELÉCTRICA**

PRESENTA:

**Ing. Javier Augusto Galvis Chacón**

DIRECTORES:

**Dr. Horacio Rostro González  
Dr. Alberto Patiño Saucedo**

Salamanca, Guanajuato

Agosto 2023

Salamanca, Gto., a 25 de julio del 2023.

**M. en I. HERIBERTO GUTIÉRREZ MARTIN  
COORDINADOR DE ASUNTOS ESCOLARES  
P R E S E N T E.-**

Por medio de la presente, se otorga autorización para proceder a los trámites de impresión, empastado de tesis y titulación al alumno(a) **Javier Augusto Galvis Chacón** del *Programa de Maestría en Ingeniería Eléctrica (Instrumentación y Sistemas Digitales)* y cuyo número de *NUA* es: **147871** del cual soy director. El título de la tesis es: **Clasificación de señales de electrocardiografía (ECG) utilizando redes neuronales pulsantes (SNN).**

Hago constar que he revisado dicho trabajo y he tenido comunicación con los sinodales asignados para la revisión de la tesis, por lo que no hay impedimento alguno para fijar la fecha de examen de titulación.

**A T E N T A M E N T E**



DR. HORACIO ROSTRO GONZÁLEZ  
DIRECTOR DE TESIS  
SECRETARIO



DR. ALBERTO PATIÑO SAUCEDO  
DIRECTOR DE TESIS



DR. RAÚL ENRIQUE SÁNCHEZ YÁÑEZ  
PRESIDENTE



DR. ARTURO GARCÍA PÉREZ  
VOCAL

*A mis padres Julio y Beatriz que lo han dado todo para el logro de mis metas.*

*A mis abuelitos, su amor y sus enseñanzas han dejado una huella eterna en mi vida. Los llevo siempre en mi corazón y su legado vivirá a través de mis logros y éxitos.*

*A mi hermano César en su apoyo incondicional en el transcurso de este gran sueño.*

*A mi sobrinito César Jr. por su tierna compañía y amor.*

*A mi cuñada Madai y a Doña Lucía, que me recibieron y me hicieron sentir como en casa.*

*A mis tíos, tías, primos y primas y a toda la familia, que siempre me motivaron a seguir adelante.*

*A mis amigos: Amanda, Duvan, Luis, Andrés, Jhonys, Camila, Jorge, Julet y a mis compañeros de Laboratorio de Sistemas Bioinspirados, porque el aprendizaje también se consigue de los propios amigos.*

*Y a todas las personas que hicieron parte de esta etapa de mi vida, que me apoyaron cuando más lo necesitaba.*

***Javier Augusto Galvis Chacón***

# Agradecimientos

Agradezco a Dios por su guía constante, su sabiduría y su presencia en cada paso de este proceso de investigación. Gracias por fortalecer mi fe, por darme la perseverancia y la determinación para superar los desafíos. Reconozco que este logro no hubiera sido posible sin su amor y bendiciones. A ti, Dios, te entrego esta tesis como una muestra de mi gratitud y reconocimiento por todo lo que has hecho en mi vida.

Agradezco a mi familia. Su amor incondicional, apoyo constante y comprensión han sido pilares fundamentales en el desarrollo de esta tesis. Gracias por creer en mí, por brindarme aliento en los momentos difíciles y por celebrar mis logros. Su compañía y respaldo han sido un motor de inspiración y motivación en este camino académico. A todos ustedes, mi familia, les dedico este trabajo con amor infinito.

Deseo expresar mi más sincero agradecimiento al Doctor Horacio Rostro González por su extraordinaria paciencia y dedicación al brindarnos su invaluable asesoría en esta tesis. Sus orientaciones y consejos han sido de gran apoyo a lo largo de estos dos años de maestría.

Agradezco al Doctor Alberto Patiño Saucedo. Por su apoyo, su paciencia y su compromiso en brindarme las herramientas necesarias para crecer y alcanzar mis metas. Su mentoría ha sido invaluable y me ha inspirado a superar los desafíos y a buscar la excelencia en mi trabajo. Estoy sinceramente agradecido por la oportunidad de aprender de su experiencia y por su constante apoyo en mi camino hacia el éxito.

Agradezco a la Doctora Patricia Batres Mendoza, al Doctor Erick Israel Guerra y a la Facultad de Sistemas Biológicos e Innovación Tecnológica (FASBIT) de la Universidad Autónoma Benito Juárez de Oaxaca (UABJO), por su apoyo durante mi estancia académica, su compromiso y dedicación han sido fundamentales para enriquecer mi experiencia y maximizar mi aprendizaje.

También quiero expresar mi agradecimiento a todo el cuerpo de profesores de la División de Ingenierías Campus Irapuato Salamanca. Me siento verdaderamente afortunado de haber contado con su valioso conocimiento y enseñanzas, las cuales han sido fundamentales para mi formación.

*Javier Augusto Galvis Chacón*

# Agradecimientos Institucionales

Este curso de Maestría en Ingeniería Eléctrica fue totalmente apoyado por el Consejo Nacional de Ciencia y Tecnología CONAHCYT, según la convocatoria “BECAS NACIONALES 2020 PRIMER PERIODO”, bajo el número de becario 1103929. Estoy muy agradecido con el CONAHCYT por el apoyo recibido durante estos 2 años.



Agradezco también a la Universidad de Guanajuato en general por contarme entre sus egresados y particularmente la División de Ingenierías Campus Irapuato Salamanca, por la calidad de la formación recibida.



# Resumen

La presente tesis tiene como objetivo investigar y analizar el uso de redes neuronales pulsantes (Spiking Neural Networks) con retardos sinápticos para la clasificación de señales electrocardiográficas, centrándose en la aplicación de diferentes esquemas de codificación de señales. Lo anterior, debido a la necesidad de mejorar la precisión y eficiencia en el análisis, el diagnóstico de señales electrocardiográficas, así como, visualizar el potencial de las SNN para abordar este desafío de manera innovadora.

En primer lugar, las señales electrocardiográficas (ECG) son un recurso invaluable en la detección y diagnóstico de enfermedades cardíacas. Sin embargo, el procesamiento y análisis de estas, plantea desafíos debido a, su naturaleza compleja junto con la presencia de ruido e interferencias. La clasificación precisa de las señales ECG es fundamental para identificar patrones y anomalías que puedan indicar condiciones médicas importantes.

Las redes neuronales pulsantes, al imitar el comportamiento de las neuronas biológicas, presentan ventajas significativas en el procesamiento de señales secuenciales y en la detección de patrones temporales. La capacidad intrínseca de las SNN para capturar la información temporal y la dinámica de las señales electrocardiográficas permite una mayor precisión en la clasificación y el diagnóstico. El modelo presentado es más eficiente desde el punto de vista energético y de hardware, lo que podría ser útil en aplicaciones donde los recursos son limitados.

Además, el uso de diferentes esquemas de codificación de señales en las SNN puede proporcionar flexibilidad y adaptabilidad en la clasificación de ECG. Al explorar y comparar distintas técnicas de codificación, como la codificación temporal y la codificación de frecuencia, es posible evaluar cuál esquema de codificación es más eficiente y preciso para la clasificación de señales electrocardiográficas en particular.

Esta investigación aporta desde el punto de vista clínico y técnico. Por un lado, una detección por medio ECG, puede favorecer a una detección temprana de enfermedades cardíacas y permitir el tratamiento adecuado. Por otro, el estudio de las SNN y su aplicación en el análisis de ECG, contribuye al avance en el campo de la inteligencia artificial aplicada a la medicina y promueve el desarrollo de sistemas de diagnóstico más efectivos.

**Palabras clave:** Codificación de bioseñales, Redes Neuronales Pulsantes (SNN).

# Abstract

The present thesis aims to investigate and analyze the use of spiking neural networks (SNN) with synaptic delays for the classification of electrocardiographic signals, focusing on the application of different signal encoding schemes. Due to the need to improve the accuracy and efficiency in the analysis and diagnosis of electrocardiographic signals, as well as to visualize the potential of SNNs to address this challenge in an innovative way.

Firstly, electrocardiographic (ECG) signals are an invaluable resource in the detection and diagnosis of cardiac disease. However, the processing and analysis of these signals poses challenges due to their complex nature along with the presence of noise and interference. Accurate classification of ECG signals is crucial to identify patterns and abnormalities that may indicate important medical conditions.

Spiking neural networks, by mimicking the behavior of biological neurons, have significant advantages in sequential signal processing and temporal pattern detection. The intrinsic ability of SNNs to capture temporal information and dynamics of electrocardiographic signals allows greater accuracy in classification and diagnosis. The present model is more energy and hardware efficient, which could be useful in applications where resources are limited.

Moreover, the use of different signal encoding schemes in SNNs can provide flexibility and adaptability in ECG classification. By exploring and comparing different encoding techniques, such as temporal encoding and frequency encoding, it is possible to evaluate which encoding scheme is more efficient and accurate for the classification of specific electrocardiographic signals.

This research contributes from clinical and technical perspectives. On one hand, detection through ECG can facilitate early detection of cardiac diseases and enable appropriate treatment. On the other hand, the study of SNNs and their application in ECG analysis contributes to progress in the field of artificial intelligence applied to medicine and promotes the development of more effective diagnostic systems.

**Keywords:** Signal processing, spike encoding, spiking neural networks (SNN).

# Índice general

Índice de figuras	XI
Índice de tablas	XIII
<b>Capítulo 1 Introducción</b>	<b>1</b>
1.1 Objetivos . . . . .	1
1.1.1 Objetivo general . . . . .	1
1.1.2 Objetivos específicos . . . . .	1
1.2 Justificación . . . . .	2
1.3 Antecedentes . . . . .	2
1.4 Organización del trabajo . . . . .	3
<b>Capítulo 2 Marco Teórico</b>	<b>4</b>
2.1 La señal de electrocardiografía (ECG) . . . . .	4
2.2 Características de la señal de electrocardiografía (ECG) . . . . .	5
2.3 Métodos de adquisición de la señal de electrocardiografía (ECG) . . . . .	6
2.4 Bases de datos de señales de electrocardiografías (ECG) . . . . .	9
2.5 Redes neuronales pulsantes (Spiking Neural Networks, SNNs). . . . .	10
2.5.1 La neurona biológica . . . . .	11
2.5.2 Redes neuronales artificiales . . . . .	13
2.5.3 Redes neuronales pulsantes (SNN) . . . . .	15
2.5.4 Modelo Leaky Integrate-and-Fire (LIF) . . . . .	17
2.5.5 Descripción del modelo de retrasos (SNN-d) . . . . .	19
2.5.6 Técnica de sobremuestreo minoritario sintético (SMOTE) . . . . .	21
2.5.7 Codificación de la señal ECG a tren de impulsos (spikes). . . . .	23
<b>Capítulo 3 Metodología del proyecto</b>	<b>30</b>
3.1 Modelo multietapa . . . . .	30
3.2 Segmentación de la señal ECG . . . . .	31
3.3 Aplicación de la técnica SMOTE en las señales segmentadas . . . . .	35
3.4 Codificación de las señales segmentadas . . . . .	36
3.5 Entrenamiento de la red neuronal pulsante (SNN) . . . . .	39
<b>Capítulo 4 Resultados</b>	<b>43</b>



---

4.1	Análisis de resultados . . . . .	43
<b>Capítulo 5 Conclusiones</b>		<b>51</b>
<b>Bibliografía</b>		<b>54</b>

# Índice de figuras

2.1	Generación de gráfica de ECG según el ciclo cardíaco [1]. . . . .	5
2.2	Esquema de un ritmo cardíaco normal [2]. . . . .	6
2.3	Diagrama de una neurona biológica en base al modelo propuesto por Ramón y Cajal [3]. . . . .	11
2.4	Ejemplo de una ANN simple [4]. . . . .	14
2.5	Comparación de modelos de neuronas de pico [5] en términos de costo de implementación y plausibilidad biológica. El modelo de Izhikevich ofrece tanto una buena plausibilidad biológica cuando usa 22 características en su modelo como un costo computacional eficiente cuando usa 13 FLOPS en su implementación. . . . .	16
2.6	Modelo de una sola neurona [6]. . . . .	17
2.7	Dinámica neuronal Leaky Integrate and Fire (LIF) [7]. . . . .	19
2.8	Proyecciones en el tiempo en un par pre/post-sináptico para a) una red neuronal recurrente (R-SNN) conectada y b) una red neuronal de retrasos (D-SNN) utilizando campos receptivos de paso 2 y profundidad 4. Los valores de peso están en color [8]. . . . .	20
2.9	Ejemplo de las salidas típicas de los esquemas de codificación seleccionados en una señal 1D. Codificación temporal A-C. Codificación de frecuencia D-F. G Codificación de población. Para cada esquema de codificación (A-G), se muestran las señales original (azul) y reconstruida (naranja) (arriba), así como la codificación de picos correspondientes (abajo). H Eficiencia de codificación de los diferentes métodos, expresada en porcentaje. Amarillo: SF. Verde: HSA. Cian: T-HSA. Azul: TBR. Púrpura: BSA. Naranja: MW [9]. . . . .	27
3.1	Estructura SNN multietapa [10]. . . . .	31
3.2	Ejemplo de anotaciones en una base de datos MIT-BIH [11]. . . . .	32
3.3	Estructura de una señal de ECG, con anotaciones para su posterior segmentación. . . . .	34
3.4	Diferentes clases de latidos del corazón después de la segmentación. . .	34
3.5	Latido del corazón, perteneciente al registro 100 de la base de datos MIT-BIH. . . . .	38
3.6	Codificación de la señal de la Figura 3.5, esquema de codificación temporal.	39

---

3.7	Codificación de la señal de la Figura 3.5, esquema de codificación de frecuencia. . . . .	40
3.8	Pesos sinápticos. . . . .	41
3.9	Pesos sinápticos con retrasos. . . . .	41
3.10	Estructura de la red SNN-d para la clasificación de (a) dos y (b) cuatro clases. . . . .	42
4.1	Matriz de confusión para dos clases. . . . .	44
4.2	Matriz de confusión para N clases. . . . .	45
4.3	Estructura de la CNN para la clasificación de dos y cuatro clases [10]. .	47

# Índice de tablas

2.1	Eficiencia promedio de pulsos de los esquemas de codificación con respecto a la duración de la señal [12]. . . . .	28
3.1	Clases de latidos del corazón y cantidad de latidos por clase después de la segmentación. . . . .	33
3.2	Clases de latidos del corazón, cantidad de latidos por clase y grupo para el conjunto de datos de entrenamiento (DS1). . . . .	35
3.3	Algoritmo SMOTE [13]. . . . .	36
3.4	Cantidad de latidos por clase y grupo para el conjunto de datos de entrenamiento (DS1), después de aplicar SMOTE. . . . .	37
3.5	Algoritmo HSA y BSA, esquema de codificación de frecuencia [9]. . . . .	38
3.6	Algoritmo SF y TBR, esquema de codificación temporal [9]. . . . .	39
3.7	Cantidad de parámetros del modelo propuesto. . . . .	42
4.1	MC de la SNN, al ingresar los datos codificados respectivamente. . . . .	45
4.2	Porcentaje de Sensibilidad, Predictividad positiva, F1 y Precisión para las clases Normal y Anormal. . . . .	46
4.3	Matriz de confusión resultante de la SNN, al ingresar los datos codificados respectivamente. . . . .	46
4.4	Porcentaje de Sensibilidad, Predictividad Positiva, F1 y Precisión para las clases N, SVEB, VEB y F. . . . .	47
4.5	Comparación de parámetros entre el modelo SNN Y CNN. . . . .	48
4.6	Comparación de la precisión general entre el modelo CNN, SNN convertida Y SNN-d. . . . .	48

# Capítulo 1

## Introducción

Este capítulo presenta una breve descripción de los objetivos generales y específicos del trabajo de investigación realizado en esta tesis. También se presenta una justificación y los antecedentes revisados en este estudio. Finalmente, se presenta la organización de la tesis, mencionando brevemente lo que se plantea en cada capítulo.

### 1.1. Objetivos

#### 1.1.1. Objetivo general

Diseño e implementación numérica de una arquitectura neuronal con neuronas pulsantes para la clasificación de señales electrocardiográficas (ECG), para ello se utilizarán diferentes esquemas de codificación de señales.

#### 1.1.2. Objetivos específicos

- Revisar el estado del arte en temas como: señales ECG, procesamiento y clasificación de señales ECG, redes neuronales pulsantes (SNN) con retardos sinápticos, codificación de señales analógicas a trenes de impulsos, SMOTE y técnicas de sobremuestreo.
- Escoger una base de datos existente, cuyas entradas correspondan a señales ECG.
- Explorar los diferentes esquemas de codificación, utilizados para la conversión de señales analógicas a trenes de impulsos, seleccionando los algoritmos que más se adecuen para la transformación de las señales ECG y posteriormente comparar cuál de ellos brinda mejores resultados.
- Explorar arquitecturas de redes neuronales pulsantes (SNN), que puedan clasificar señales ECG, a fin de visualizar las ventajas y limitaciones al trabajar con este tipo de información.

- 
- Redactar el informe de tesis, presentando un énfasis principalmente, en el uso de las redes neuronales pulsantes como clasificadores de señales ECG y la comparación de diferentes algoritmos de codificación.

## 1.2. Justificación

La importancia del desarrollo de esta tesis se puede ver en dos sentidos. Por un lado, es de suma importancia aplicar todo tipo de soluciones éticas que den solución a problemas de salud, como lo puede ser enfermedades cardiovasculares. Por otro lado, en los últimos meses nos hemos encontrado con un tema muy disruptivo, el cual es el uso masivo de la inteligencia artificial (IA). Sin embargo, es notable la gran cantidad de recursos que requieren estas inteligencias para procesar información. En este sentido, con la aplicación de técnicas más eficientes como las redes neuronales pulsantes, las cuales no se derivan de la inteligencia artificial, sino de la neurociencia computacional, se busca reducir el consumo energético requerido por las diferentes soluciones de IA. En este sentido, lo que se propone en este trabajo es poder aplicar las redes neuronales pulsantes a la clasificación de señales de cardiografía para la detección futura de enfermedades cardiovasculares, y al mismo tiempo estudiar las capacidades de estas redes en aplicaciones que tradicionalmente son realizadas por métodos de IA.

## 1.3. Antecedentes

Un electrocardiograma (ECG o EKG), es la representación gráfica de la actividad eléctrica generada por el corazón [1]. Las señales, detectadas por medio de electrodos metálicos adheridos a las extremidades y a la pared torácica, son amplificadas y registradas por el electrocardiógrafo. Las derivaciones de ECG (leads), están configuradas para mostrar las diferencias instantáneas de potencial entre pares específicos de electrodos. La utilidad del ECG se deriva de su disponibilidad inmediata como prueba no invasiva, económica y muy versátil. Además, es utilizado para detectar arritmias e isquemia miocárdica, revelando hallazgos relacionados con alteraciones metabólicas potencialmente mortales o con una mayor susceptibilidad a un paro cardíaco repentino.

Para realizar la clasificación de las señales ECG se pueden utilizar diferentes tipos de redes neuronales. Trabajos previos han demostrado la viabilidad de algoritmos de aprendizaje automático para detectar enfermedades a partir de ECG [14]; métodos como Redes convolucionales temporales (TCN) [15], Máquina de vectores de soporte (SVM) [16], Discriminante lineal (LD) [17], AdaBoost [18], Red neuronal de propagación inversa de algoritmo genético (GA-BPNN) [19], Perceptrón multicapa (MLP) [20], Memoria a largo plazo (LSTM) [21] y, especialmente, Redes neuronales convolucionales (CNN) [22, 23].

En [24] utilizan redes neuronales profundas complejas para realizar la clasificación

---

de los latidos del corazón. Aunque obtienen una alta precisión, requieren de un alto costo computacional durante su entrenamiento. Otros trabajos, enfocados al estudio de los latidos del corazón (usando aprendizaje automático), aplican diferentes tipos de preprocesamiento a la señal a fin de obtener mejores resultados, como en [25] donde se usan las wavelets, un complejo método matemático que permite analizar funciones usando escalas. Como conclusión puede decirse que estos clasificadores presentan una alta precisión, ya que tienen muchos años de estudio y recopilación de información. Sin embargo, el principal problema de estas tecnologías es el elevado consumo de recursos, el uso desmedido de energía y los altos costos al momento de su implementación. Por tal motivo, puede ser relevante considerar a las redes neuronales de impulsos (Spiking Neural Network, SNN), las cuales se catalogan como la tercera generación de redes neuronales artificiales (RNA) [26]. Se piensa que, en un futuro cercano, revolucionarán los modelos neuronales actuales al incluir información temporal. Estas son un tipo de RNA más realistas que las clásicas, es decir, procesan la información de una forma más similar a las redes neuronales biológicas.

## 1.4. Organización del trabajo

El presente documento de Tesis de Maestría se compone de 5 capítulos y 3 apéndices con información complementaria. A continuación, se describe brevemente cada uno de ellos.

El Capítulo 1 contiene los antecedentes y justificación de esta tesis, así como el objetivo general y objetivos particulares del proyecto.

El Capítulo 2 aborda los principales conceptos que dan soporte a la tesis, una breve explicación de las señales de electrocardiografía (ECG), métodos de adquisición de señales de ECG, bases de datos de ECG, redes neuronales de impulsos (SNN), codificación de señales ECG a tren de impulsos, entre otros.

En el Capítulo 3 se describe la metodología que se aplicó para el proyecto, en ella se explica detalladamente los pasos para crear un clasificador de enfermedades cardiovasculares, desde la obtención de una base de datos, segmentación, codificación y clasificación de señales de Electrocardiografía.

En el Capítulo 4 se presentan los resultados obtenidos de cada experimento llevado a cabo

Al final del documento se presentan las conclusiones generales y la bibliografía utilizada en esta investigación.

# Capítulo 2

## Marco Teórico

Este capítulo presenta los aspectos teóricos que fundamentan esta tesis, lo cuales están estructurados de forma concisa para facilitar el entendimiento y los alcances de la misma. Comenzando con una descripción de los aspectos relacionados al entendimiento de las señales de electrocardiografía, las cuales son clasificadas por una red neuronal pulsante que también es descrita en este capítulo.

### 2.1. La señal de electrocardiografía (ECG)

Una señal de electrocardiografía o electrocardiograma (ECG) es un registro de la actividad eléctrica cardíaca. La generación de la señal ECG se presenta debido a la naturaleza eléctrica de las señales que se registran en el corazón, las cuales permiten coordinar sus movimientos, con estas señales podemos estudiar su comportamiento midiendo y analizando las mismas. De manera específica, los electrocardiogramas son registros gráficos de las corrientes que circulan en el corazón. Si bien el ECG no puede diagnosticar enfermedades específicas por sí solo, puede proporcionar información importante que ayuda a los médicos a identificar diversas afecciones cardíacas [1]. A continuación, se mencionan algunas enfermedades y condiciones que pueden detectarse o evaluarse mediante un electrocardiograma:

- Enfermedad coronaria: El ECG puede mostrar signos de isquemia o falta de flujo sanguíneo en el corazón, lo que puede indicar enfermedad coronaria, como la obstrucción de las arterias coronarias.
- Infarto de miocardio: Un ECG puede mostrar cambios característicos en el patrón de ondas que sugieren un infarto de miocardio o ataque al corazón.
- Arritmias cardíacas: El ECG puede detectar y clasificar diferentes tipos de arritmias, como taquicardia, bradicardia, fibrilación auricular o extrasístoles, al mostrar alteraciones en la secuencia y la regularidad de las ondas cardíacas.



- Bloqueo cardíaco: El ECG puede revelar anomalías en la conducción eléctrica del corazón, como bloqueos cardíacos, que pueden indicar trastornos en el sistema de conducción eléctrica del corazón.
- Hipertrofia ventricular: El ECG puede mostrar cambios en las dimensiones eléctricas del corazón que sugieren un engrosamiento anormal de las paredes ventriculares, lo que puede ser indicativo de hipertrofia cardíaca.
- Miocarditis y pericarditis: El ECG puede revelar patrones de alteración en las ondas cardíacas que indican inflamación del músculo cardíaco (miocarditis) o la membrana que lo rodea (pericarditis).
- Anomalías congénitas del corazón: En algunos casos, el ECG puede proporcionar pistas sobre la presencia de anomalías cardíacas congénitas en recién nacidos y niños.

## 2.2. Características de la señal de electrocardiografía (ECG)

El paso del potencial de acción a través de las células cardíacas genera formas de onda, las cuales, sumadas entre sí, crean una electrocardiograma. En la Figura 2.1, se presenta un esquema sobre la formación de la onda ECG:

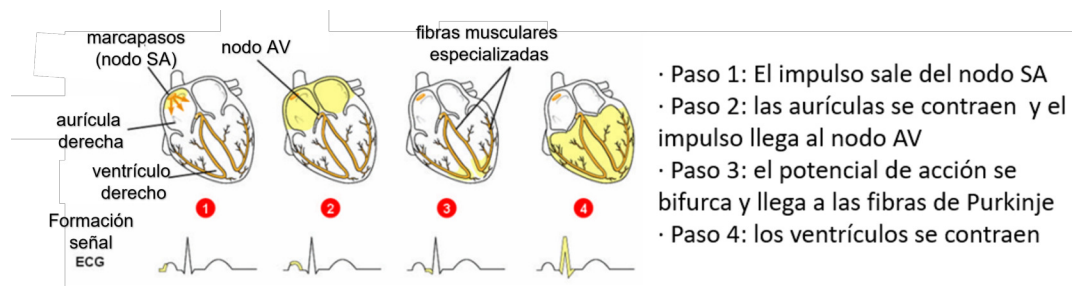


Figura 2.1: Generación de gráfica de ECG según el ciclo cardíaco [1].

Un trazado de un electrocardiograma en ritmo normal se presenta en la Figura 2.2, donde se observa una onda P, un complejo QRS y una onda T. De forma general, se considera que, en un individuo normal deben estar presentes las siguientes características en las ondas, intervalos, complejos y segmentos:

- Onda P: Inicio de un ciclo o período cardíaco. Dura entre 0.09 s y 0.11 s. Tiene una amplitud aproximada de 0.25 mV.
- Intervalo P-R: Espacio comprendido entre el fin de la onda P y el inicio del complejo QRS. Dura entre 0.11 s y 0.20 s.

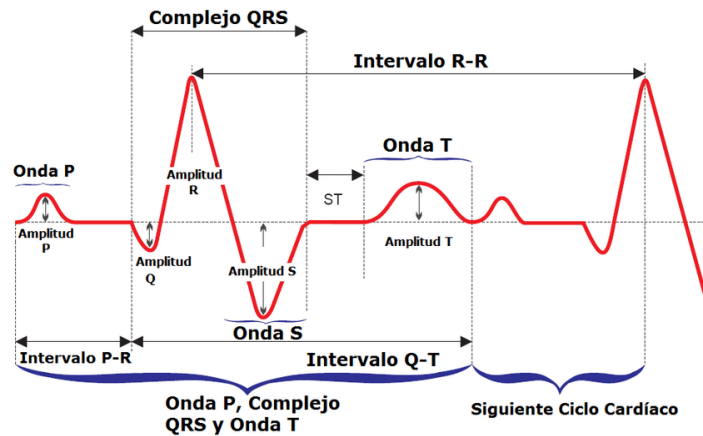


Figura 2.2: Esquema de un ritmo cardíaco normal [2].

- Complejo QRS: Compuesto por una despolarización eléctrica del músculo cardíaco, conocida como depresión u onda Q; una rápida repolarización, que constituye el pico más elevado de la señal, onda R; y otra inversión de la polaridad, dada por la depresión S. Es frecuente que el complejo QRS tenga las ondas separadas: Q, R y S, aunque no siempre ocurre así. Dura entre 0.07 s y 0.11 s. La onda R llega a alcanzar alrededor de 1.60 mV; mientras que la Q es cerca del 25 % de la amplitud de R.
- Intervalo Q-T: Posee una duración entre 0.35 s a 0.44 s.
- Segmento ST: Comprende entre el fin del complejo QRS y el inicio de la onda T. Suele ser isoelectrico en los ECG normales. Tiene un intervalo de tiempo de entre 0.05 s a 0.15 s
- Onda T: De igual polaridad al complejo QRS. Alcanza entre 0.1 a 0.5 mV.

## 2.3. Métodos de adquisición de la señal de electrocardiografía (ECG)

El perfil del ECG varía dependiendo de los puntos de referencia donde se coloquen los electrodos. Las diferentes posiciones que pueden tener, permiten apreciar mejor ciertas propiedades del corazón. Estas se describen a continuación:

### Derivación I

En la derivación I se coloca el electrodo positivo en el brazo izquierdo y el electrodo negativo en el brazo derecho. Dado que el vector medio, va desde la parte superior derecha a la parte inferior izquierda, la energía fluye hacia el positivo de la derivación

---

I, dando lugar, a un complejo QRS con morfología ascendente. Sin embargo, debido a que el vector medio no fluye directamente hacia la derivación I, sino que se aproxima a ella con un cierto ángulo, la altura del complejo QRS es de grado moderado.

### **Derivación II**

En la derivación II, el electrodo positivo se localiza en el pie izquierdo y el electrodo negativo en el brazo derecho. Dado que el vector medio fluye directamente hacia el electrodo de la derivación II, esta derivación es la que muestra generalmente los complejos QRS más altos y las ondas P más prominentes en todo el ECG de 12 derivaciones, y esta es la razón por la que la derivación II es la derivación de monitorización favorita en muchas unidades de cuidados intensivos (UCI) y unidades de telemetría.

### **Derivación III**

En la derivación III se coloca el electrodo positivo en el pie izquierdo y el electrodo negativo en el brazo izquierdo. El flujo del vector medio se aproxima a la derivación III desde la parte inferior derecha, dando lugar de nuevo a una morfología ascendente del complejo QRS. Dado que el ángulo de aproximación es más estrecho que el ángulo entre el vector medio y la derivación I, los complejos QRS que aparecen en la derivación III son más altos que los detectados por la derivación I.

### **Derivación aVR**

La derivación aVR es la única derivación de los miembros localizada en el lado derecho del cuerpo. Su electrodo de monitorización positivo se localiza en el brazo derecho y está dirigido hacia abajo y hacia la izquierda. El vector medio también fluye en direcciones inferior e izquierda, en alejamiento respecto a la derivación aVR, dando lugar a una morfología negativa de todas las ondas. En un ECG normal, la aVR es la única derivación de los miembros en la que los complejos QRS muestran una morfología descendente.

### **Derivación aVL**

La derivación aVL coloca un electrodo positivo en el brazo izquierdo y está enfrentada hacia la derecha y hacia abajo, en dirección al centro del corazón (al contrario de la derivación I, que está enfrentada estrictamente hacia la derecha). El vector medio se aproxima a la derivación aVL con un ángulo muy abierto y, por ello, los complejos QRS que aparecen en esta derivación son los más bajos en el conjunto de las derivaciones de los miembros.

### **Derivación avF**

La derivación avF tiene su electrodo positivo en el lado izquierdo y está enfrentada en línea recta al centro del tórax. El vector medio se aproxima a la derivación avF

---

con un ángulo bastante directo, aunque no tanto como en la derivación I; por ello, los complejos QRS que aparecen en la derivación aVF son muy altos y las ondas P muy prominentes. Las derivaciones II, III y aVF están dirigidas hacia arriba, hacia el vector medio que les llega, de manera que las ondas que muestran comparten numerosas características como complejos QRS fuertemente positivos y ondas P prominentes. Dado que estas derivaciones están dirigidas hacia arriba y se localizan en el fondo o en la pared ventricular inferior del corazón, se denominan derivaciones inferiores.

En [27] definen seis derivaciones torácicas o precordiales, las cuales se sitúan en la parte anterior del tórax y miden el vector medio en el plano horizontal.

- La derivación V1, se localiza en el borde esternal derecho, en el nivel del cuarto espacio intercostal, por encima del ventrículo derecho y del tabique interventricular.
- La derivación V2, se localiza en el lado izquierdo del esternón, en el nivel del cuarto espacio intercostal.
- La derivación V3, está en el punto medio de la distancia que separa las derivaciones V2 y V4.
- La derivación V4, se localiza en la línea medioclavicular, en el nivel del quinto espacio intercostal.
- La derivación V5, se localiza en la parte anterior de la línea axilar, en el nivel del quinto espacio intercostal.
- La derivación V6, se localiza en la línea medioaxilar, en el nivel del quinto espacio intercostal, por encima de la pared lateral del ventrículo izquierdo

Además, se presentan modificaciones de las derivaciones normales que resultan ser más efectivas y con mayor uso en la actualidad.

### **Derivación de extremidad modificada II (MLII)**

La derivación de extremidad modificada II (del inglés, Modified Limb Lead II (MLII)), es la modificación de los electrodos de las extremidades del sistema MLL [28]. El electrodo del brazo derecho se coloca en el tercer espacio intercostal derecho del sujeto, ligeramente a la izquierda de la línea medioclavicular. El electrodo del brazo izquierdo, se coloca en el quinto espacio intercostal derecho, ligeramente a la derecha de la línea medioclavicular, y el electrodo de la pierna izquierda, se coloca en el quinto espacio intercostal derecho, en la línea medioclavicular. El electrodo de la pierna derecha se coloca en el tobillo derecho del sujeto. La polaridad del electrodo del brazo derecho es negativa, y la polaridad del electrodo del brazo izquierdo y de la pierna

---

izquierda es positiva. Se aplica la terminología habitual; por ejemplo, la diferencia de potencial entre el electrodo del brazo derecho (RA) y el electrodo del brazo izquierdo (LA) todavía se indica como derivación I, etc. Las posiciones estándar de los electrodos precordiales V1–V6 no cambian en este estudio durante las grabaciones de MLL.

## 2.4. Bases de datos de señales de electrocardiografías (ECG)

De acuerdo con las diferentes derivaciones o métodos que existen para adquirir las señales de ECG, muchas instituciones o centros médicos han creado sus propias bases de datos, en la mayoría de los casos estas bases son de acceso público y se encuentran alojadas en páginas de internet tales como Physionet [29] y scientific data [30].

Para el presente caso de estudio se utilizó la base de datos denominada MIT-BIH Arrhythmia Database [11], desarrollada por la División de Ciencias y Tecnología de la Salud del Instituto Tecnológico de Massachusetts (MIT) y de la Universidad Harvard. A continuación, se presenta una breve descripción de los elementos que contiene esta base de datos.

La base de datos de arritmias de MIT-BIH contiene 48 extractos de media hora de registros de ECG ambulatorios de dos canales, obtenidos de 47 sujetos estudiados por el Laboratorio de Arritmias de BIH entre 1975 y 1979. Se eligieron 23 registros al azar de un conjunto de 4000 ECG de una población mixta de pacientes hospitalizados (alrededor del 60 %) y pacientes ambulatorios (alrededor del 40 %) en el Hospital Beth Israel de Boston. Las 25 grabaciones restantes se seleccionaron del mismo conjunto para incluir arritmias menos comunes, pero clínicamente significativas que no estarían bien representadas en una pequeña muestra aleatoria.

Las grabaciones se digitalizaron a 360 muestras por segundo por canal con una resolución de 11 bits en un rango de 10 mV. Dos o más cardiólogos anotaron de forma independiente cada registro; los desacuerdos se resolvieron para obtener las anotaciones de referencia legibles por computadora para cada latido (aproximadamente 110 000 anotaciones en total) incluidas en la base de datos.

Alrededor de la mitad (25 de 48 registros completos y archivos de anotaciones de referencia para los 48 registros) de esta base de datos ha estado disponible gratuitamente desde el inicio de PhysioNet en septiembre de 1999. Los 23 archivos de señales restantes, que habían estado disponibles solo en MIT-BIH Arrhythmia CD-ROM de la base de datos, se publicaron en febrero de 2005 en [11].

Para la mayoría de los registros, se utilizó la derivación MLII (Modified Limb Lead II), que se obtiene, colocando los electrodos en el tórax. La señal más baja suele ser una derivación modificada V1 (ocasionalmente V2 o V5, y en un caso V4); en cuanto a la

---

señal superior, estos también se colocan en el pecho. El laboratorio de arritmias de BIH, utiliza habitualmente esta configuración. Los complejos QRS normales suelen ser prominentes en la señal superior. Sin embargo, el eje de la derivación para la señal inferior puede ser casi ortogonal al eje eléctrico cardíaco medio (es decir, los latidos normales suelen ser bifásicos y pueden ser casi isoeletricos). Por lo tanto, los latidos normales suelen ser difíciles de discernir en la señal más baja, aunque los latidos ectópicos a menudo serán más prominentes. Una excepción notable es el registro 114, para el cual se invirtieron las señales. Dado que esto ocurre ocasionalmente en la práctica clínica, los detectores de arritmias deben estar equipados para hacer frente a esta situación. En los registros 102 y 104 no fue posible utilizar la derivación II modificada debido a los apósitos quirúrgicos de los pacientes. Se utilizó la derivación V5 modificada para la señal superior en estos registros.

La base de datos MIT-BIH Arrhythmia data base, alberga la información detallada de cada uno de los 48 registros disponibles, lo cual implica la existencia de los siguientes tres archivos por registro:

- Un archivo `.dat`, almacena los datos reales de la señal de electrocardiografía (ECG). Este archivo representa las mediciones eléctricas del corazón en forma de señales digitales.
- Un archivo `.hea`, contiene información descriptiva y metadatos sobre el registro de ECG correspondiente. Incluye detalles como el nombre del paciente, la edad, el sexo, la frecuencia de muestreo, la duración del registro y los nombres de los canales de ECG.
- Un archivo `.atr`, proporciona las anotaciones de eventos para el registro de ECG. Estas anotaciones marcan eventos específicos en el ECG, como latidos cardíacos, arritmias y otros eventos relevantes.

Para utilizar un formato válido y poder procesar estos registros en Matlab es necesario convertir los archivos `.dat` y `.hea` en un archivo `.csv`, por lo cual se utiliza la biblioteca WFDB [31] en Python; Esta biblioteca es específica para trabajar con bases de datos de señales fisiológicas, incluida la base de datos MIT-BIH Arrhythmia. En particular, se emplea la función `wfdb.rdrecord()` para leer los archivos `.dat` y `.hea` y obtener los datos del ECG en forma de matrices en Python.

## 2.5. Redes neuronales pulsantes (Spiking Neural Networks, SNNs).

Las Redes Neuronales Artificiales son un campo muy importante y ampliamente estudiado dentro de la Inteligencia Artificial. Inspirándose en el comportamiento conocido

---

del cerebro humano (principalmente el referido a las neuronas y sus conexiones), trata de crear modelos artificiales que solucionen problemas difíciles de resolver mediante técnicas algorítmicas convencionales. En años recientes, estas redes han sido estudiadas desde un punto de vista más cercano al comportamiento de las neuronas biológicas en el cerebro, para lo cual se ha desarrollado el concepto de redes neuronales pulsantes, las cuales a diferencia de los modelos de redes neuronales tradicionales, incorporan el parámetro de tiempo. De esta manera se busca que señales como las de electrocardiografía, las cuales varían con el tiempo puedan ser estudiadas de manera mas efectiva y eficiente. A continuación, se da una descripción biológica de las neuronas y cómo a partir de estas se desarrollan y estudian modelos de redes neuronales artificiales o pulsantes como en este caso.

### 2.5.1. La neurona biológica

Fue Ramón y Cajal [3] quién descubrió la estructura celular (neurona) del sistema nervioso. Defendió la teoría de que las neuronas se interconectaban entre sí de forma paralela, y no formando un circuito cerrado como el sistema sanguíneo.

Una neurona consta de un cuerpo celular (soma) de entre 10 y 80  $\mu\text{m}$ , del que surge un denso árbol de ramificaciones (dendritas) y una fibra tubular (axón) de entre 100  $\mu\text{m}$  y un metro.

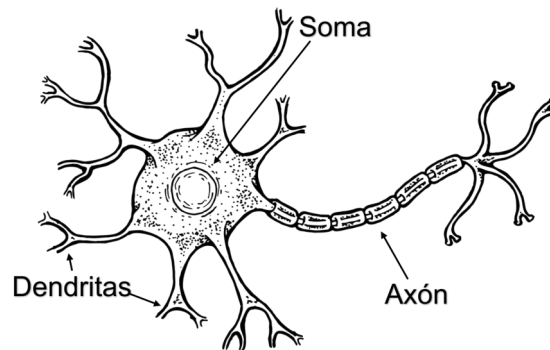


Figura 2.3: Diagrama de una neurona biológica en base al modelo propuesto por Ramón y Cajal [3].

De alguna forma, una neurona es un procesador de información muy simple:

- Canal de entrada: dendritas.
- Procesador: soma.
- Canal de salida: axón.

---

Una neurona cerebral puede recibir unas 10,000 entradas y enviar a su vez su salida a varios cientos de neuronas.

La conexión entre neuronas se llama sinapsis. No es una conexión física, existen al menos, unos 2 mm de separación. Son conexiones unidireccionales, en la que la transmisión de la información se hace de forma eléctrica en el interior de la neurona y de forma química entre neuronas; gracias a unas sustancias específicas llamadas neurotransmisores.

No todas las neuronas son iguales, existen muchos tipos diferentes según el número de ramificaciones de sus dendritas, la longitud del axón y otros detalles estructurales. Sin embargo, todas ellas operan con los mismos principios básicos.

### **El potencial de acción**

Las comunicaciones del sistema neuronal se producen a través de potenciales de acción o pulsos, de ahí el nombre de las redes neuronales estudiadas en esta tesis. Una cadena de ellos emitidos por una sola neurona, se denomina tren de pulsos, que son eventos temporales asíncronos, sin una tasa de disparo fundamental, aunque se han observado fenómenos de sincronización entre grandes poblaciones neuronales. Dado que todos los pulsos en una misma neurona se parecen, la forma del potencial de acción no contiene ninguna información. En cambio, el momento y la frecuencia de los pulsos parecen ser los responsables del cálculo neuronal.

Otra característica de las neuronas es, la presencia de un período refractario absoluto, que impide que la neurona dispare inmediatamente después de un primer potencial de acción, incluso con una entrada fuerte, proporcionando pulsos bien separados. El período refractario absoluto es seguido por un período de refractariedad relativa en el que es difícil, pero no imposible, excitar un potencial de acción [32].

### **Sinapsis y plasticidad neuronal**

Las sinapsis permiten la transmisión de señales eléctricas y químicas entre una neurona transmisora de mensajes o presináptica y una neurona postsináptica, siendo el potencial de acción presináptico el portador del mensaje. Las señales interactúan con un mecanismo que controla la conductancia de la membrana de la neurona postsináptica, dando a la neurona presináptica la capacidad de influir en la excitabilidad de las neuronas postsinápticas.

Este es un ingrediente esencial para el procesamiento de la información neuronal, ya que el flujo de información está controlado por la fuerza del efecto sináptico. Existen dos tipos de sinapsis, excitatorias e inhibitorias, dependiendo de si el efecto de la sinapsis aumenta o disminuye el potencial de membrana.

Los experimentos electrofisiológicos muestran que la amplitud del efecto sináptico sobre el potencial de membrana postsináptico causado por una neurona presináptica



---

dada no es fija, sino que puede cambiar con el tiempo. La estimulación adecuada puede inducir cambios en la respuesta postsináptica que duran horas o días. Si la estimulación conduce a un aumento persistente de la eficacia de la transmisión sináptica, el efecto se denomina potenciación a largo plazo de las sinapsis, o LTP para abreviar. Si el resultado es una disminución de la eficacia sináptica, se denomina depresión a largo plazo (LTD) [33]. Además, el tiempo relativo a corto plazo entre los pulsos presinápticos y postsinápticos influye en la eficacia de las sinapsis. Este efecto se denomina plasticidad dependiente del tiempo de pulso (STDP) [34]. Se cree que estos cambios persistentes a corto y largo plazo en la fuerza sináptica son responsables del aprendizaje y la memoria.

### 2.5.2. Redes neuronales artificiales

Existen diferentes definiciones que ayudan a construir una idea general de lo que son realmente las redes neuronales, como, por ejemplo

- Es un modelo simplificado que emula el modo en que el cerebro humano procesa la información. Funciona simulando un número elevado de unidades de procesamiento interconectadas que parecen versiones abstractas de neuronas.
- Es un sistema de computación compuesto por un gran número de elementos simples, elementos de procesos muy interconectados, los cuales procesan información por medio de su estado dinámico como respuesta a entradas externas.
- Las redes neuronales artificiales son redes interconectadas masivamente en paralelo de elementos simples (usualmente adaptativos) y con organización jerárquica, las cuales intentan interactuar con los objetos del mundo real del mismo modo que lo hace el sistema nervioso biológico.

Las unidades de procesamiento están dispuestas en capas. Una red neuronal generalmente consta de tres partes: una capa de entrada, donde las unidades representan los campos de entrada; una o más capas ocultas; y una capa de salida con una o más entidades que representan los campos objetivo. Los elementos están conectados con una fuerza de conexión variable (o peso). Los datos de entrada se presentan en la primera capa y los valores se propagan de cada neurona a cada neurona en la siguiente capa. Finalmente se envía el resultado hacia la capa de salida. En la Figura 2.4 se presenta un ejemplo de una red neuronal.

La red aprende examinando registros individuales, generando predicciones para cada registro y ajustando los pesos si se producen predicciones incorrectas. Este proceso se repite muchas veces y la red continúa mejorando sus predicciones hasta que se cumplen uno o más criterios de parada.

Inicialmente, todos los pesos son aleatorios y las respuestas generadas por la red pueden ser muy errados. La red aprende en el entrenamiento. Los ejemplos de resultados

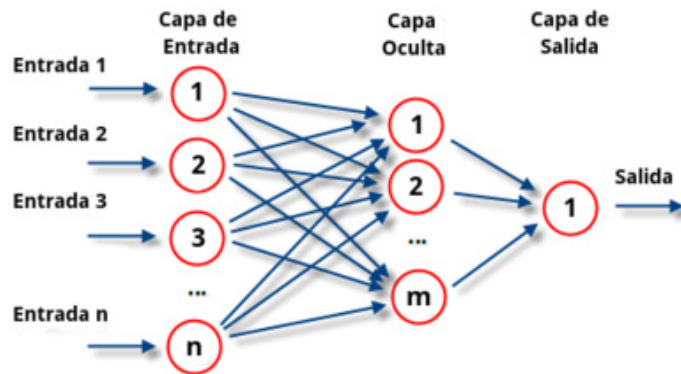


Figura 2.4: Ejemplo de una ANN simple [4].

conocidos se presentan continuamente a la red y sus respuestas se comparan con dichos resultados. La información de esta comparación se retroalimenta a la red cambiando gradualmente los pesos. A medida que avanza el entrenamiento, la red se vuelve cada vez más precisa para replicar los resultados conocidos. Después del entrenamiento, la red se puede utilizar para casos futuros en los que se desconoce el resultado.

Existen varios tipos de redes neuronales, como son las monocapa o perceptrón simple, perceptrón multicapa (MLP), convolucionales (CNN), redes neuronales recurrentes (RNN), redes neuronales pulsantes (SNN), de retroalimentación o redes de base radial (RBF).

Además, según el método que utilicen para el aprendizaje también se pueden clasificar como aprendizaje supervisado, aprendizaje por corrección de error, estocástico, aprendizaje auto supervisado, hebbian, competitivo y comparativo, o, por último, aprendizaje por refuerzo.

Algunos usos de las redes neuronales son:

- Aproximación de funciones, o el análisis de regresión, incluyendo la predicción de series temporales, funciones de amplitud y el modelado.
- Clasificación, incluyendo el reconocimiento de patrones y la secuencia de reconocimiento, detección y de la toma de decisiones secuenciales.
- Procesamiento de datos, incluyendo el filtrado, el agrupamiento, la separación ciega de las señales y compresión.
- Robótica, incluyendo la dirección de manipuladores y prótesis.
- Ingeniería de control, incluyendo control numérico por computadora.

---

### 2.5.3. Redes neuronales pulsantes (SNN)

La tesis se centra en las SNN (del inglés: spiking neural networks), la cuales son una clase especial de red neuronal artificial, donde las neuronas se comunican mediante secuencias de pulsos. En [26] consideran las SNN como la tercera generación de redes neuronales artificiales y se piensa que, en un futuro cercano, revolucionarán los modelos neuronales actuales al incluir información temporal.

En [4] se menciona que la ciencia, estudia los procesos naturales que operan, a los seres y su entorno, estos están evolucionando en dos dimensiones: espacio y tiempo. Este tipo de análisis, se presenta en las neuronas de tipo spiking, ya que utilizan modelos dinámicos que también evolucionan con el tiempo. Debido a su comportamiento similar al de las neuronas biológicas, los modelos de neuronas pulsantes, proporcionan una poderosa herramienta, para el análisis de los procesos cerebrales elementales. Al mismo tiempo, las SNN ofrecen soluciones a una amplia gama de problemas simples en ingeniería aplicada, como procesamiento rápido de señales, detección de eventos, clasificación, reconocimiento de voz, navegación espacial y control de motores, entre otros.

Los modelos SNN, son superiores a los métodos tradicionales de Machine Learning [4], las características más relevantes son:

- Modelado eficiente de procesos que involucran diferentes escalas de tiempo.
- Integración de modalidades, como sonido y visión en un solo sistema.
- Procesamiento rápido y masivo de información en paralelo.
- Procesamiento de información compacto.
- Estructuras escalables.
- Bajo consumo de energía si se implementa sobre plataformas neuromórficas.

La principal diferencia de este tipo de neuronas artificiales con los modelos tradicionales, como el Multilayer Perceptron (MLP), es la inclusión de información temporal, esto sucede cuando la función de activación (segunda generación) se cambia a un modelo Spiking (tercera generación). Los modelos tradicionales solo analizan la información de un momento específico. Se podría decir que no importa lo que pasó antes y después de que se presentara ese patrón. Lo anterior puede afectar el nivel de generalización de los modelos entrenados para resolver determinados problemas. Otra diferencia importante es el uso de trenes de pulsos codificados como entrada, de la misma manera el modelo produce un tren de pulsos como salida. En ambas generaciones los modelos utilizan pesos sinápticos.

Las neuronas biológicas, se comunican generando y propagando pulsos eléctricos, llamados potenciales de acción o en algunos casos picos. Esta característica, se convirtió en un paradigma central, para la teoría de los modelos neuronales pulsantes, los cuales, muestran tres propiedades principales: 1) Procesar información proveniente de muchas entradas y producir una única salida de pulsos; 2) Su probabilidad de activación aumenta con entradas excitatorias y disminuye con entradas inhibitorias; 3) En [35] explican la dinámica de caracterizar por lo menos una variable de estado, cuando las internas alcanzan unos valores determinados, el modelo genera uno o más picos.

Debido a estas tres propiedades, las neuronas pulsantes tienen la capacidad de realizar el procesamiento de datos, los cuales, contienen información temporal. Por lo tanto, estas evolucionan dinámicamente con el tiempo: uno o más trenes de pulsos ingresan como entradas al modelo, la dinámica interna está influenciada por estas y genera una salida codificada en forma de pulsos.

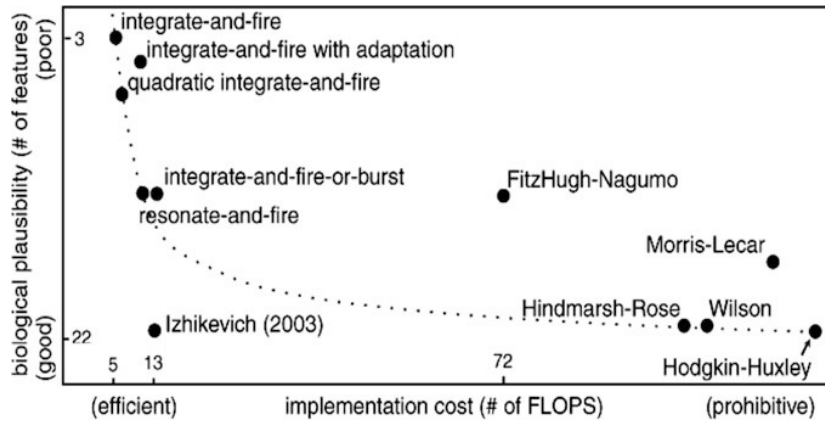


Figura 2.5: Comparación de modelos de neuronas de pico [5] en términos de costo de implementación y plausibilidad biológica. El modelo de Izhikevich ofrece tanto una buena plausibilidad biológica cuando usa 22 características en su modelo como un costo computacional eficiente cuando usa 13 FLOPS en su implementación.

Hoy en día, existen muchos modelos de redes neuronales pulsantes, algunos de ellos tienen un mayor nivel de plausibilidad biológica, mientras que otros tienen un mejor costo de implementación. En [4] se muestran algunos de ellos, desde el modelo de Hodgkin y Huxley, que es el que presenta mayor similitud biológica pero con peor rendimiento computacional, hasta los modelos más simples como el LIF (Leaky Integrate and Fire). Otro de los modelos mencionados es el de Izhikevich (IZ) que según el trabajo original presenta un alto rendimiento [5]. Las comparaciones se presentan en la Figura 2.5.

### 2.5.4. Modelo Leaky Integrate-and-Fire (LIF)

El modelo leaky integrate-and-fire (LIF) es un modelo simple de neurona aplicado en las redes neuronales pulsantes que se utiliza para describir cómo las neuronas en el cerebro responden a señales eléctricas. En este modelo, se asume que la neurona es una esfera con una membrana que contiene iones, los cuales pueden moverse a través de la membrana para crear una corriente eléctrica como se presenta en la Figura 2.6.

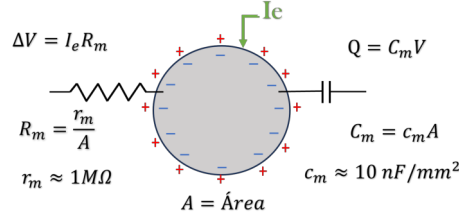


Figura 2.6: Modelo de una sola neurona [6].

Se aplica la Ley de Ohm y la ecuación del capacitor para corrientes de resistencia y capacitancia, de esta manera podemos obtener una descripción matemática del modelo.

$$I_R = \frac{u_m(t) - u_{res}}{R_m} \quad (2.1)$$

$$I_c = C_m \frac{du_m(t)}{dt} \quad (2.2)$$

donde  $u_{res}$  es el potencial de inversión o reposo y tiene en cuenta la naturaleza específica de iones de la resistencia de la membrana,  $C_m$  es la capacitancia de la membrana.

Si se estimula la neurona con una corriente de entrada  $I_e(t)$  (hacia adentro como se observa en la Figura 2.6) con el potencial de membrana  $u_m(t)$ , podemos describir la ecuación general del modelo:

$$I_e(t) = \frac{u_m(t) - u_{res}}{R_m} + C_m \frac{du_m(t)}{dt} \quad (2.3)$$

donde la constante de tiempo de la membrana  $\tau_m$  es:

$$\tau_m = R_m C_m \quad (2.4)$$

Entonces, se puede reescribir la Ecuación 2.3 como:

$$\frac{du_m(t)}{dt} = \frac{u_{res} - u_m(t)}{R_m C_m} + \frac{I_e(t)}{C_m} \quad (2.5)$$

---


$$\frac{du_m(t)}{dt} = \frac{u_{res} - u_m(t)}{\tau_m} + \frac{I_e(t)}{C_m} \quad (2.6)$$

### Modelo LIF en tiempo discreto

Una red de neuronas pulsantes está formada por al menos una neurona presináptica y una postsináptica con potenciales de membrana  $u_{pre}(t)$  y  $u_{pos}(t)$ , respectivamente. Cuando el potencial de membrana de la neurona presináptica alcanza un umbral  $u_{th}$ , la neurona se dispara, su potencial de membrana se restablece a  $u_{res}$  y el pulso estimula la neurona postsináptica a través de una corriente  $I_e(t)$ .

En las neuronas biológicas reales,  $u_{th}$  suele ser negativo, pero por simplicidad en las simulaciones, se trabaja con valores positivos y un potencial de reposo y reinicio igual a cero. Además, como la simulación tiene una resolución temporal  $dt$ , la Ecuación 2.6 se discretiza (con tiempo discreto  $k$ ) de la siguiente manera:

$$\frac{u[k] - u[k-1]}{\Delta t} = \frac{-u[k-1]}{\tau_m} + \frac{I[k-1]}{C_m} \quad (2.7)$$

Donde,  $\tau_m$  es la constante de tiempo de la membrana y  $C_m$  es la capacitancia de la membrana.

Ahora, resolviendo para la neurona postsináptica y teniendo en cuenta el restablecimiento por disparo [6], el comportamiento discretizado del potencial de membrana en una neurona postsináptica viene dado por:

$$u_{pos}[k] = \begin{cases} 0, & u_{pos}[k-1] \geq u_{th}, \\ u_{pos}[k-1](1 - \frac{\Delta t}{\tau_m}) + \frac{I[k-1]\Delta t}{C_m}, & \text{de otro modo} \end{cases} \quad (2.8)$$

### Modelo de Red

La neurona LIF recibe señales sinápticas de otras neuronas y las integra a lo largo del tiempo, es decir, acumula las corrientes entrantes. En el caso de múltiples conexiones presinápticas a la neurona, su corriente de entrada se calcula como el efecto acumulativo de pulsos presinápticos:

$$I[k] = \sum_{j=1}^M w_j \theta_j[k] + I_{bias} \quad (2.9)$$

donde  $M$  es el número de neuronas presinápticas (con potenciales de membrana  $u_j$ ),  $w_j$  es la fuerza sináptica de la  $j$ -ésima neurona presináptica (positiva si la sinapsis es excitadora y negativa si es inhibitoria),  $I_{bias}$  es una corriente de compensación y  $\theta_j[k]$  denota la aparición de un pulso en la  $j$ -ésima neurona presináptica en el paso de tiempo

actual  $k$ . La conectividad de esta red es equivalente a una ANN con conexiones hacia adelante y permite que los SNN se organicen en múltiples capas, con pesos sinápticos entrenables.

Si la corriente acumulada supera un umbral  $u_{th}$ , la neurona dispara un potencial de acción, lo que resulta en una ráfaga de actividad eléctrica que se propaga a lo largo de la neurona y se transmite a otras neuronas. El pulso es calculado por todas las neuronas postsinápticas en el siguiente paso de tiempo, y su potencial de membrana se restablece a  $u_{reset}$ . La función de disparo está dada por:

$$\theta_j[k] = \begin{cases} 1, & u_j[k-1] \geq u_{th}, \\ 0, & \text{de otro modo} \end{cases} \quad (2.10)$$

Una métrica importante del estado de la neurona de disparo es su velocidad de activación. Por definición, la tasa de disparo  $r[k]$  de una neurona dada cuya simulación comenzó en el tiempo  $k = 0$  es:

$$r[k] = \frac{\sum_t \theta[k]}{k} \quad (2.11)$$

En la Figura 2.7 se presenta un esquema propuesto [7] acerca del modelo LIF, presente en esta sección.

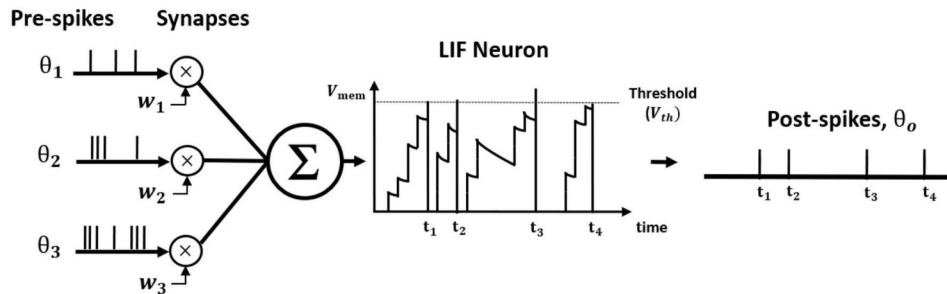


Figura 2.7: Dinámica neuronal Leaky Integrate and Fire (LIF) [7].

Los pulsos previos son modulados por el peso sináptico, para integrarse como el flujo de entrada actual en el potencial de membrana que decae exponencialmente. Cada vez que el potencial de membrana cruza el umbral de activación, la postneurona dispara un post-spike y restablece el potencial de membrana [7].

### 2.5.5. Descripción del modelo de retrasos (SNN-d)

Se utiliza el modelo conocido como Delay Model o en español modelo de retraso [8], el cual se basa en una versión multicapa del modelo Leaky Integrate-and-Fire (LIF)

visto en la Subsección anterior (2.5.4). Las neuronas LIF tienen estado y representan un compromiso entre la plausibilidad biológica y la eficiencia computacional para la implementación del hardware. Su excitación depende tanto de su entrada dependiente del tiempo  $I$  de otras neuronas como de su estado interno, conocido como el potencial de membrana  $u$  sujeto a una integración con fugas con una constante de tiempo  $\tau$ . Las ecuaciones de la actualización del potencial de membrana en una implementación en tiempo discreto de una neurona de pico LIF son:

$$u_k = u_{k-1} e^{-\frac{1}{\tau}(1 - \theta_{k-1})} + I_{k-1} \quad (2.12)$$

$$\theta_k = \begin{cases} 1 & u_k \geq u_{th}, \\ 0 & \text{de otro modo} \end{cases} \quad (2.13)$$

donde  $\theta$  denota una función para generar activaciones o picos cada vez que el potencial de membrana alcanza un umbral asociado a la neurona,  $u_{th}$ .

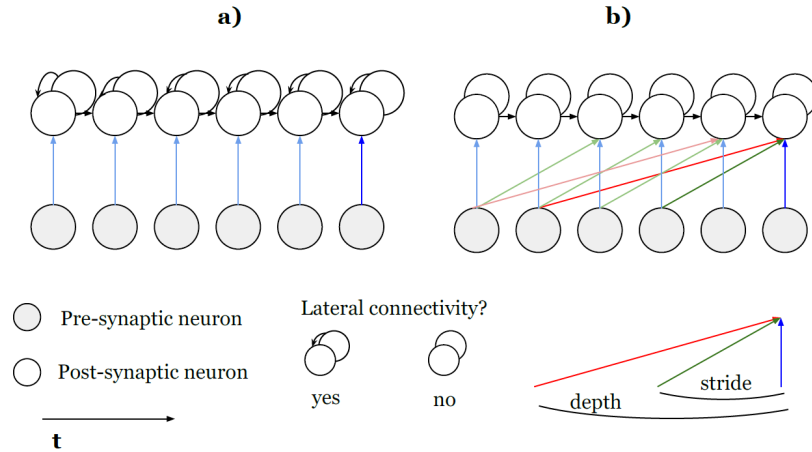


Figura 2.8: Proyecciones en el tiempo en un par pre/post-sináptico para a) una red neuronal recurrente (R-SNN) conectada y b) una red neuronal de retrasos (D-SNN) utilizando campos receptivos de paso 2 y profundidad 4. Los valores de peso están en color [8].

Las SNN multicapa se pueden organizar como redes feedforward o recurrentes. En una capa SNN recurrente, las neuronas exhiben conectividad lateral, como en la Figura 2.8(a), y su entrada se calcula sumando la contribución ponderada de las  $N$  neuronas a la capa anterior o presináptica y de las  $M$  neuronas vecinas en su propia capa, como se muestra en la ecuación 2.14:



---


$$I_{k[recurrent]} = \sum_{i=1}^N w_i \theta_{i,k} + \sum_{j=1}^M w_j \theta_{j,k} \quad (2.14)$$

Para incorporar retrasos axónicos en redes neuronales pulsantes, se crean múltiples proyecciones o sinapsis con retraso en el tiempo para cada par de neuronas presinápticas/postsinápticas. Así, la activación de una neurona en un momento dado depende tanto de su estado actual como de un subconjunto de activaciones pasadas de neuronas en la capa presináptica, con proyecciones directas. La entrada de una neurona que incorpora el modelo propuesto para retrasos axonales es:

$$I_{k[delays]} = \sum_{d \in D} \sum_{i=1}^N w_{i,d} \theta_{i,k-d} \quad (2.15)$$

donde  $D \in [0, T]$  es el conjunto de retrasos elegidos para una tarea determinada. El control sobre el paso temporal de los retrasos y la profundidad del campo receptivo temporal está incluido en el modelo (ver Figura 2.8 (b) para una visualización del concepto). Esto aumenta la flexibilidad en el número total de parámetros.

Este modelo es el que se utiliza en nuestra tesis como tema de investigación.

### 2.5.6. Técnica de sobremuestreo minoritario sintético (SMOTE)

Ya que se ha hablado de las señales de electrocardiografía y de las redes neuronales pulsantes que se utilizarán para la clasificación de estas. Ahora se hablará de una técnica para el balanceo de las clases de las señales contenidas en la base de datos que ha sido utilizada para la experimentación.

SMOTE (Synthetic Minority Over-sampling Technique) es una técnica utilizada en aprendizaje automático para tratar el problema del desbalance de clases [13]. El desbalance de clases se refiere a una situación en la que las clases en un conjunto de datos están representadas de manera desproporcionada, con una clase siendo significativamente menos frecuente que la otra.

El problema del desequilibrio de clases en conjuntos de datos puede tener consecuencias graves en el rendimiento de los modelos de aprendizaje automático, especialmente en aquellos que se utilizan para la clasificación binaria. Si una clase está infrarrepresentada en el conjunto de datos, el modelo puede tener dificultades para aprender y detectar patrones en esa clase, lo que resulta en una menor precisión en la clasificación de instancias de esa clase.

---

SMOTE aborda este problema generando ejemplos sintéticos de la clase minoritaria para aumentar su presencia en el conjunto de datos. La técnica de sobremuestreo sintético utiliza una combinación de interpolación lineal y aleatoria para generar nuevas observaciones sintéticas en el vecindario cercano de las observaciones existentes de la clase minoritaria.

En [36] se propone un enfoque de sobremuestreo en el que la clase minoritaria se sobremuestra creando ejemplos “sintéticos” en lugar de sobremuestrear con reemplazo. Se generan entonces ejemplos sintéticos de una manera menos específica de la aplicación, operando en el “espacio de funciones” en lugar del “espacio de datos”. La clase minoritaria se sobremuestra tomando cada muestra de clase minoritaria e introduciendo ejemplos sintéticos a lo largo de los segmentos de línea que unen los  $k$ -vecinos más cercanos de la clase minoritaria. Dependiendo de la cantidad de sobremuestreo requerida, los  $k$ -vecinos más cercanos se eligen aleatoriamente, donde realizan la implementación utilizando solo cinco vecinos más cercanos. Por ejemplo, si la cantidad de sobremuestreo necesaria es del 200 %, solo se eligen dos vecinos de los cinco más cercanos y se genera una muestra en la dirección de cada uno. Las muestras sintéticas se generan de la siguiente manera: se toma la diferencia entre el vector de características (muestra) bajo consideración y su vecino más cercano. Después se multiplica esta diferencia por un número aleatorio entre 0 y 1, y se agrega al vector de características en consideración. Esto provoca la selección de un punto aleatorio a lo largo del segmento de línea entre dos características específicas. Este enfoque obliga efectivamente a que la región de decisión de la clase minoritaria se vuelva más general.

En términos más específicos, la técnica SMOTE se puede resumir en los siguientes pasos:

- Selección aleatoria de una instancia de la clase minoritaria: SMOTE selecciona aleatoriamente una observación de la clase minoritaria para generar ejemplos sintéticos.
- Identificación de los  $k$ -vecinos más cercanos: SMOTE identifica los  $k$ -vecinos más cercanos a la instancia seleccionada de la clase minoritaria.  $k$  es un parámetro que se puede establecer previamente y se utiliza para controlar el nivel de sobremuestreo que se desea.
- Generación de nuevas instancias sintéticas: Para cada vecino cercano identificado en el paso anterior, SMOTE genera una nueva instancia sintética que está en algún lugar a lo largo de la línea que conecta la instancia original de la clase minoritaria y su vecino. La ubicación de la nueva instancia se determina mediante una combinación de interpolación lineal y aleatoria. El número de nuevas instancias generadas se controla mediante el parámetro tasa de sobremuestreo.
- Agregación de instancias sintéticas al conjunto de datos original: Las nuevas instancias sintéticas generadas se agregan al conjunto de datos original, lo que resulta

---

en un conjunto de datos más equilibrado con respecto a las clases minoritarias y mayor precisión en la clasificación de las instancias de esta clase.

En general, SMOTE aborda el problema del desequilibrio de clases al aumentar el número de ejemplos de la clase minoritaria, con el fin de mejorar la capacidad del modelo para detectar y clasificar las instancias de la clase minoritaria en el conjunto de datos, lo que puede conducir a una mejor precisión general del modelo. En esta investigación el desbalance de clases se presenta en la Tabla 3.2 “Clases de latidos del corazón”, del Capítulo 3, donde la clase mayoritaria son las señales normales (N) con un total de 45801 elementos y las clases minoritarias son las arritmias referentes a latidos ectópicos supraventricular (SVEB) y ventricular (VEB), a latidos de fusión (F) y a los latidos desconocidos (Q) con un total de 5179 elementos, concluyendo que las clases anormales contienen menos latidos.

Sin embargo, es importante tener en cuenta que SMOTE puede generar ejemplos sintéticos que pueden no ser realistas o relevantes para el dominio del problema. Además, la generación excesiva de ejemplos sintéticos puede conducir a un sobreajuste del modelo y disminuir su capacidad para generalizar a nuevos conjuntos de datos. Por lo tanto, es importante ajustar el nivel de sobremuestreo según las características específicas del conjunto de datos y el modelo que se está entrenando. También se recomienda utilizar técnicas de validación cruzada para evaluar el rendimiento del modelo con diferentes niveles de sobremuestreo.

### **2.5.7. Codificación de la señal ECG a tren de impulsos (spikes).**

Una vez que los datos son balanceados, las señales de electrocardiografía son codificadas en pulsos a fin de que puedan ser procesadas por la red neuronal pulsante. Este proceso se describe a continuación:

En [37] se da una idea general de la importancia de la codificación de los datos. En una SNN, la información viaja entre las unidades de procesamiento, en forma de eventos de pulsos binarios. Por lo tanto, los sistemas SNN están inspirados en las soluciones de procesamiento de información del cerebro biológico. Las mediciones del mundo real, proporcionan señales temporales de valor real analógicas (continuas o discretas); por lo tanto, es necesario implementar un método de codificación, para convertir los valores analógicos en eventos y brindar una entrada a dichos sistemas. Esta codificación de analógico a impulsos, puede comprimir considerablemente el tamaño de los datos de la señal, ya que el tren de impulsos es una serie de valores binarios. A su vez, esta señal binaria, proporciona un procesamiento rápido, especialmente si se utiliza hardware diseñado para ello, por ejemplo, SpiNNaker [38]. En última instancia, una codificación de pulsos correcta puede conducir a una mejor conservación de la información, junto con la reducción y compresión de los datos de entrada.

---

Es extremadamente importante, generar la entrada del tren de pulsos al SNN, de manera que, se conserve el contenido de información relevante para la tarea de la señal. El problema principal recae en, cuál información se pierde y cuál se conserva, una forma de visualizar, lo cual nos dirá qué tan efectiva fue la codificación.

Un enfoque consiste en no evaluar la efectividad de la codificación por separado, sino, para toda la aplicación SNN, por ejemplo, la elección dada se considera efectiva, si la salida de todo el sistema SNN produce buenos resultados. Este es el enfoque elegido para la tesis, con el cual se evaluarán las codificaciones.

Otro enfoque, es intentar optimizar el paso de codificación por sí mismo. Sin embargo, la comparación de señales originales y codificadas no es trivial. En el mejor de los casos, se puede aplicar el algoritmo de decodificación correspondiente y comparar la señal reconstruida con la original.

Cada método de codificación, tiene una forma diferente de extraer información de la señal de entrada. La selección del método específico, depende de las características de la señal, como, la presencia de información relevante en el dominio del tiempo o de la frecuencia, el ruido en los datos, si estos se pueden cambiar o escalar. También es necesario comprender, cómo cambia la codificación las características de la señal, por ejemplo, si corta el espectro de frecuencia y si puede suprimir el ruido. Además, también se debe considerar el tipo de SNN que se utilizará, es decir, qué tipo de entrada puede aceptar.

En [9] se especifica que los algoritmos de codificación de pulsos se pueden dividir en tres categorías.

## **Esquema de codificación poblacional (Population coding schemes)**

Se ha demostrado que la codificación poblacional se usa ampliamente para aplicaciones que utilizan como eje principal el cerebro (sensor, motor). Como, por ejemplo, la actividad de las neuronas (o poblaciones de neuronas) está determinada por una distribución sobre la señal de entrada, otra de ellas, son las células ganglionares de la retina, conocidas por ser las únicas neuronas involucradas en la transmisión de la información visual desde la retina al cerebro. Los estudios en la corteza somatosensorial, responsable del procesamiento de las sensaciones somáticas, también revelaron la existencia de codificación poblacional.

Algunos de estos modelos simples de codificación son: la codificación de posición y los campos receptivos gaussianos, más conocidos como GRF.

- (I) La codificación de posición (Position coding) es la forma más simple de codificación poblacional. En este método, a un grupo de neuronas se le asigna una pequeña parte de la distribución subyacente. Esta se puede dividir linealmente,

---

pero también, son posibles distribuciones no lineales. Si bien tener una distribución lineal permite codificar una señal de manera homogénea, se puede considerar implementar una distribución gaussiana en su lugar. Por lo tanto, la codificación sería más adecuada para el control neuromórfico, siendo la entrada el error en un punto de ajuste objetivo.

- (II) El modelo GRF (Gaussian Receptive Fields), codifica la señal por medio de un conjunto de neuronas de distribuciones de actividad se definen como ondas gaussianas, determinadas por un centro  $\mu$  y una varianza  $\sigma^2$ . En el modelo propuesto, todos los centros  $\mu_i$  de las neuronas  $i$  están espaciados regularmente para cubrir la máxima amplitud de la señal de entrada, mientras que las varianzas se igualan. La secuencia entre los intervalos de tiempo regulares, ahora se divide en una serie de sub-pasos de tiempo, que permiten una combinación de codificación de posición y codificación temporal. Las neuronas con una activación alta, se dispararán al principio, y las neuronas con una activación más baja después (o no se dispararán si la respuesta está por debajo de cierto umbral). Si un componente temporal, no es deseable o posible para la aplicación, también se puede elegir para asignar el valor de activación de las neuronas a una probabilidad de pulso, lo que significa que las neuronas más cercanas al valor codificado tienen una mayor probabilidad de pulso.

### Esquema de codificación de frecuencia (Rate coding schemes)

De acuerdo con el paradigma de codificación de frecuencia, la información se agrupa en la tasa de pulso en lugar del tiempo de pulso específico. Como ejemplo, se presentan los siguientes algoritmos:

- (I) El algoritmo de pulsos de Hough (HSA, Hough Spike Algorithm): equivalente a un modo basado en umbral, el HSA utiliza una convolución inversa entre la señal de entrada almacenada en búfer, con un filtro de respuesta de impulso finito (FIR) para determinar los tiempos de los pulsos.
- (II) El algoritmo umbral de pulsos de Hough con Umbral (T-HSA, Threshold Hough Spike Algorithm): similar al HSA, el T-HSA introduce un umbral para comparar con el error entre la señal y el filtro. Cada vez que el error supera este umbral, se emite un pulso y la señal de entrada se actualiza restando la respuesta del filtro. En THSA, el umbral depende de la señal y debe determinarse antes de la codificación.
- (III) El algoritmo pulsos de Ben (BSA, Ben's Spike Algorithm): al igual que en los esquemas de codificación anteriores, el BSA aplica una convolución inversa de la señal con un filtro FIR. Para determinar cuándo debe generarse un pulso, el algoritmo utiliza dos errores, el primero es la suma de las diferencias entre la señal y el filtro, y el segundo, es la suma de los valores de la señal. Luego,

---

el algoritmo genera pulsos, al comparar el primer error con una fracción de la señal acumulada, definida como el producto entre el segundo error y un umbral predefinido. A diferencia de T-HSA, el umbral depende del filtro, lo que permite mantener el mismo valor para diferentes señales.

Si bien la codificación de frecuencia se considera la forma universal en que las neuronas codifican la información, muchos estudios destacaron el desempeño deficiente de los esquemas de codificación de frecuencia en comparación con los algoritmos de codificación temporal.

### **Esquema de codificación temporal (Temporal coding schemes)**

Las representaciones temporales, también denominadas codificación de pulsos, proporcionan una codificación basada en el tiempo en la que la información se codifica en la sincronización exacta de los pulsos. A diferencia de la codificación de frecuencia, la codificación temporal brinda más capacidad de información y está respaldada por estudios neurofisiológicos que muestran que la información auditiva y visual se procesa con mayor precisión en el cerebro [12]. Se han propuesto varios modelos. Asumiendo que la información más significativa es transportada por los primeros pulsos, la codificación de orden de rango (ROC) organiza los pulsos con respecto a su hora de llegada. Inspirándose en las células ganglionares, se introdujo la codificación de fase de latencia (LPC) para combinar el pulso de tiempo exacto proporcionado por la codificación temporal con la información de fase (la codificación de información espacial en las células ganglionares). La codificación de fase también se ha implementado en la investigación de las neuronas en la corteza auditiva humana. Como ejemplo, se presentan los siguientes algoritmos:

- (I) La representación basada en el tiempo (TBR, Temporal-Based Representation): también llamado contraste temporal, el algoritmo TBR genera pulsos cada vez que la variación de la señal, entre dos marcas de tiempo consecutivas, supera un umbral fijo. Las cámaras basadas en eventos, como el sensor de visión dinámica (DVS), implementan este esquema de codificación para generar un flujo de eventos en velocidad extremadamente rápida.
- (II) El algoritmo de paso adelante (SF, Step Forward Algorithm): basado en el esquema de codificación TBR, el SF utiliza una señal de referencia para calcular la variación de la señal de entrada. En cuanto al modelo TBR, un pulso (+1 o -1) se emite siempre que la variación supere el umbral. Simultáneamente, la línea base se actualiza por el umbral dependiendo de la polaridad del pulso.
- (III) El algoritmo de ventana móvil (MW, Moving Window Algorithm): el modelo MW es similar al modelo SF, pero aquí la señal de referencia se define como la media de las intensidades de señal anteriores durante una ventana de tiempo. Dado que los modelos SF y MW presentan un componente adaptativo en el cálculo de las variaciones de la señal mediante la línea de base, se sabe que resultan en una mejor reconstrucción de la señal codificada después de la decodificación.

En [9] realizan una comparación de las diferentes codificaciones, para ello inicialmente grafican cada una de las codificaciones y decodificaciones de una misma señal original, identificando que tanto la calidad de la reconstrucción de la señal como la eficiencia en la codificación de pulsos varían de un modelo a otro, como se expone en la Figura 2.9.

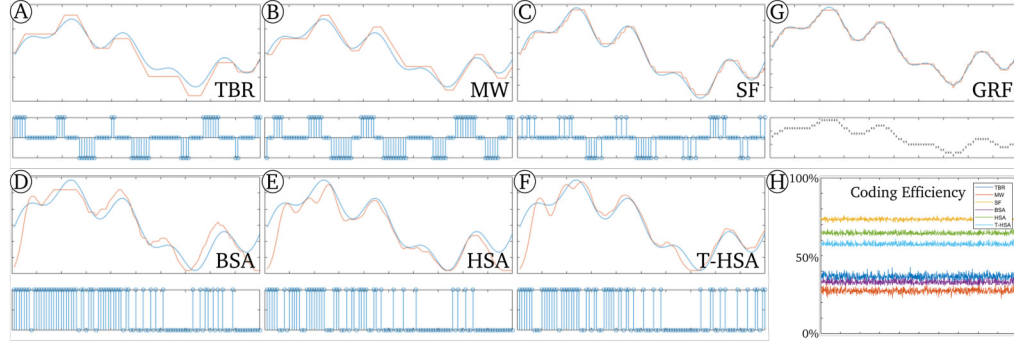


Figura 2.9: Ejemplo de las salidas típicas de los esquemas de codificación seleccionados en una señal 1D. Codificación temporal A-C. Codificación de frecuencia D-F. G Codificación de población. Para cada esquema de codificación (A-G), se muestran las señales original (azul) y reconstruida (naranja) (arriba), así como la codificación de picos correspondientes (abajo). H Eficiencia de codificación de los diferentes métodos, expresada en porcentaje. Amarillo: SF. Verde: HSA. Cian: T-HSA. Azul: TBR. Púrpura: BSA. Naranja: MW [9].

Estas observaciones se investigan más a fondo evaluando la eficiencia de aumento de cada algoritmo. Para ello se utiliza la siguiente ecuación, tal como se propone en [9]:

$$spiking \ efficiency = \left(1 - \frac{spike \ count}{length \ of \ the \ signal}\right) * 100 \quad (2.16)$$

La deriva a largo plazo, debida a errores acumulativos en la reconstrucción de la señal, también se considera extendiendo la duración de la señal de 5 segundos a 100 segundos. Los resultados estadísticos se dan en la Tabla 2.1 (eficiencia de adición).

De la Tabla 2.1, se observa que los algoritmos SF, HSA y T-HSA dan como resultado una alta eficiencia de impulso ( $>50\%$ ). En particular, observamos que el modelo SF no se ve afectado por la duración de la señal (estable en  $73\%$ ), mientras que la eficiencia de pulsos de la codificación de frecuencia HSA y T-HSA cae 13 (HSA) y 7 (T-HSA) puntos, respectivamente. Por el contrario, el esquema MW mantiene una eficiencia de picos del  $27\%$ , independientemente de la duración de la señal. Por último, es interesante notar que la desviación estándar de la eficiencia de pulsos tiende a 0 cuando la duración de la señal aumenta. Esto sugiere que, para cada algoritmo, y ciertamente para cada tipo de señal, una duración óptima de la señal asegura una eficiencia de pulso estable.

	TBR	MW	SF	BSA	HSA	T-HSA
Case #1 – T_max = 1 sec (1 period), nb_tests = 1000						
MEAN	36.7	29.2	74.6	45.8	74.0	62.4
SD	4.1	3.4	2.0	1.8	1.3	1.1
Case #2 – T_max = 5 sec (5 periods), nb_tests = 1000						
MEAN	36.7	27.5	73.4	33.1	64.5	57.5
SD	1.7	1.5	0.9	1.3	1.0	1.1
Case #3 – T_max = 15 sec (15 periods), nb_tests = 1000						
MEAN	36.6	27.2	73.2	30.0	62.1	55.8
SD	1.0	0.9	0.5	0.8	0.6	0.6
Case #4 – T_max = 50 sec (50 periods), nb_tests = 1000						
MEAN	36.7	27.1	73.2	29.3	61.6	55.6
SD	0.6	0.5	0.3	0.4	0.3	0.3
Case #5 – T_max = 100 sec (100 periods), nb_tests = 1000						
MEAN	36.6	27.1	73.2	29.0	61.4	55.5
SD	0.4	0.3	0.2	0.2	0.2	0.2

Tabla 2.1: Eficiencia promedio de pulsos de los esquemas de codificación con respecto a la duración de la señal [12].

En [39] se proporciona una base teórica acerca de la codificación de señales pulsantes, ofreciendo una visión general de diferentes implementaciones orientadas a aplicaciones que utilizan estos esquemas. En él, se concluye que los esquemas de codificación más utilizados son: los esquemas de frecuencia, los cuales convencen por su robustez frente a fluctuaciones y ruido, los esquemas de codificación temporal, los cuales logran densidades de información y eficiencias más altas. Por tal motivo, esta investigación se enfoca en comparar estos dos esquemas de codificación, específicamente se realizan pruebas simuladas utilizando los algoritmos de codificación BSA, HSA, SF y TBR.

## Principales contribuciones del proyecto

Los anteriores ejes temáticos contribuyen al desarrollo del modelo como aporte de las herramientas matemáticas para brindar solución al clasificador de señales ECG, aplicando las redes neuronales pulsantes con las bioseñales. El modelo propuesto de red neuronal pulsante SNN-d dentro de sus características destacan: bajo consumo de recursos, eficiencia en el uso de energía y considerables costos de implementación, convirtiéndose en las primeras pruebas simuladas para la clasificación de bioseñales, específicamente señales de electrocardiografía. El diseño del modelo del clasificador utiliza técnicas propias del tratamiento de señales, en este caso, la técnica SMOTE que permite el balanceo de clases debido a que la base de datos utilizada cuenta con la problemática de desequilibrio de clases. Además, se plantea las pruebas para los



---

esquemas de codificación como contribución a la investigación mediante la comparación de los esquemas de codificación de frecuencia y esquemas de codificación temporal, empleando los algoritmos BSA, HSA, SF y TBR.

# Capítulo 3

## Metodología del proyecto

En este capítulo se explica el desarrollo del proyecto, el cual, contiene la aplicación e integración de los conceptos vistos anteriormente. Primero se explica en detalle el preprocesamiento de las señales, para extraer características morfológicas y clasificar los latidos cardíacos anormales.

### 3.1. Modelo multietapa

El modelo consiste en usar primero una red neuronal pulsante simplificada para clasificar las señales como normal y anormal, las señales anormales se refieren a las arritmias correspondientes a latidos ectópicos supraventricular (SVEB) y ventricular (VEB), a latidos de fusión (F) y a los latidos desconocidos (Q). Luego, clasificar solo los latidos detectados como anormales en la primera etapa en las clases detalladas, es decir, en la segunda etapa se clasifican las cuatro clases que incluyen N, SVEB, VEB y F pero ahora se ha reducido la tasa de falsos negativos de latidos cardíacos normales, en la Figura 3.1 se ilustra el modelo multietapa.

Para mejorar el desempeño del clasificador se utiliza el modelo propuesto en [10], el cual presenta una arquitectura de dos etapas para mejorar la precisión y reducir el consumo de energía mediante el método de parada anticipada, esto aplicado a la red neuronal pulsante (SNN).

Esto en parte se hace debido a que, en muchas de las aplicaciones, como dispositivos portátiles, los latidos que son registrados pertenecen al grupo de latidos normales, entonces, utilizando el modelo propuesto estas señales se “detienen” en la primera etapa, reduciendo el consumo promedio de energía, haciendo más eficiente el clasificador portátil de señales de ECG.

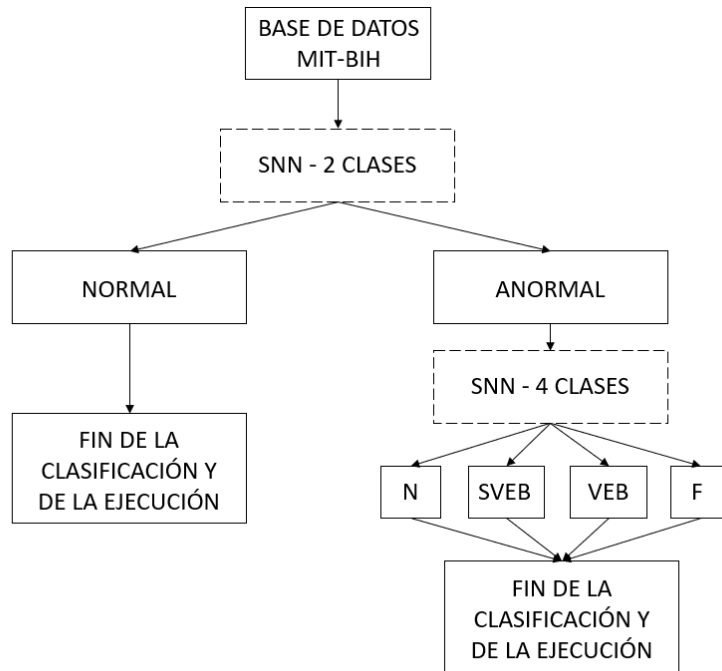


Figura 3.1: Estructura SNN multietapa [10].

### 3.2. Segmentación de la señal ECG

En la Sección 2.4 se presenta la base de datos del MIT-BIH Arrhythmia Database, disponible para descargarse y visualizarse de forma gratuita en [11]. Se eligió esta base de datos, al ser una de las más utilizadas y recomendadas por ANSI/AAMI EC57:1998/(R)2008 [40] (AAMI, Asociación para el avance de la instrumentación médica), encargada de especificar y definir el protocolo, para realizar las evaluaciones a fin de asegurar que los experimentos sean reproducibles y comparables. También, es de las primeras bases de datos disponibles para este tipo de análisis y se ha perfeccionado constantemente a lo largo de los años [41].

La mayoría de los latidos registrados en las bases de datos, tienen anotaciones asociadas con el tipo de latido o eventos. Tanto para la clase, como para los puntos de referencia (p. ej., punto R, máxima amplitud del latido) estas observaciones son fundamentales para el desarrollo y evaluación de métodos automáticos de clasificación de arritmias. El estándar ANSI/AAMI EC57:1998/(R) 2008, también especifica cómo se deben registrar estas anotaciones en las bases de datos. Un ejemplo se puede ver en la Figura 3.2, en la que se encuentra la derivación MLII en la parte superior de la imagen, la derivación V1 en la inferior y algunas anotaciones en el centro.

Aunque existen varios tipos de arritmias cardíacas, la AAMI recomienda que solo



Figura 3.2: Ejemplo de anotaciones en una base de datos MIT-BIH [11].

algunos tipos se detecten mediante equipos/métodos. Hay 15 clases recomendadas para la arritmia que se clasifican en 5 superclases o grupos: normal (N), latido ectópico supraventricular (SVEB), latido ectópico ventricular (VEB), latido de fusión (F) y latido desconocido (Q). La Tabla 3.1, ilustra las 15 clases y sus símbolos, así como la jerarquía de los 5 grupos (superclases) y la cantidad de latidos por clase.

El estándar AAMI especifica un protocolo para pruebas y evaluación de métodos en la clasificación de arritmias, también estipula las bases de datos que deben utilizarse. Sin embargo, no determina qué tipo de pacientes/latidos cardíacos son necesarios para construir el modelo a clasificar (fase de entrenamiento) y qué pacientes/latidos cardíacos deben tomarse en cuenta para la evaluación, es decir, la fase de prueba, lo que puede generar resultados sesgados. Por ejemplo, Chazal et al. [42] demostraron que el uso de los latidos del corazón del mismo paciente, tanto para el entrenamiento como para la evaluación, sesga el proceso de evaluación. Esto se debe a que, los modelos tienden a aprender las particularidades del latido del corazón del paciente durante el entrenamiento, obteniendo números expresivos durante la prueba (muy cercanos al 100%). Sin embargo, en un entorno clínico, un algoritmo/método completamente automático, encontrará latidos cardíacos de diferentes personas a los que solían aprender en la fase de entrenamiento. Con la intención de especificar un protocolo, en el trabajo de Chazal et al. [42] se propuso una división en dos conjuntos, de los latidos del corazón de la base de datos MIT-BIH, para que esta sea más coherente con la realidad. El primer conjunto se compone de todos los latidos de los registros: 101, 106, 108, 109, 112, 114, 115, 116, 118, 119, 122, 124, 201, 203, 205, 207, 208, 209, 215, 220, 223 y 230, denominado Conjunto de datos 1 (DS1). Mientras que el segundo está compuesto por todos los latidos de los registros: 100, 103, 105, 111, 113, 117, 121, 123, 200, 202, 210, 212, 213, 214, 219, 221, 222, 228, 231, 232, 233 y 234, denominado Conjunto de datos 2 (DS2).

Clases de latidos cardíacos	Anotación latidos	Tipos de latidos cardíacos	Cantidad de latidos cardíacos
N Latido normal	N	Latido normal	74482
	L	Latido con bloqueo de rama izquierda	8075
	R	Latido con bloqueo de rama derecha	7259
SVEB Latido ectópico supraventricular	e	Latido de escape auricular	16
	j	Latido de escape nodal	229
	A	Latido auricular prematuro	2546
	a	Latido auricular prematuro aberrante	150
	J	Latido prematuro nodal	83
VEB Latido ectópico ventricular	S	Latido prematuro supraventricular	2
	V	Contracción ventricular prematura	6903
E	E	Latido de escape ventricular	106
	F	Fusión de latido ventricular y latido normal	803
F Latido de fusión	F		
Q	/	Latido artificial	0
Latido desconocido	f	Fusión de latido artificial y normal	0
	Q	Latido no clasificable	15

Tabla 3.1: Clases de latidos del corazón y cantidad de latidos por clase después de la segmentación.

Los autores solo utilizaron DS1 para construir el modelo de clasificación, mientras que DS2 se reservó para la evaluación. De esta forma, garantizaron que el modelo creado no tuviera contacto con los latidos del corazón de DS2, es decir, los latidos del corazón de DS1 y DS2 provienen de personas diferentes. Este proceso es el que se conoce como esquema o paradigma interpaciente [43].

En Chazal et al. [42], los registros se dividieron principalmente en pares e impares. La selección final se logró intercambiando algunos entre las partes para equilibrar las clases. Vale la pena mencionar que, los registros 201 y 202 son del mismo paciente, pero están en conjuntos diferentes. Los demás registros pertenecen a un solo paciente.

Es importante explicar que no se tuvo en cuenta 4 de los registros de la base de datos MIT-BIH, 102, 104, 107 y 217 según lo recomendado por AAMI [40], debido al alto desequilibrio de los tipos de latidos en estos (90 % de los latidos son N latidos, 3 % y 6 % son SVEB y VEB y solo 1 % son F), y de los registros de pacientes que utilizan marcapasos.

En consecuencia, para adquirir el complejo QRS y detectar el pico R, es necesario realizar un código en Matlab. En segundo lugar, se eliminan los primeros 3 y el último latido consecutivo para asegurar, que los datos analizados sean del estado estable.

Finalmente, las señales se segmentan utilizando la ubicación de anotación pico R proporcionada en el conjunto de datos MIT-BIH, simplemente se toma la ventana de 251 puntos centrada alrededor de su pico R, tomando como punto inicial 100 muestras hacia atrás y como punto final 150 muestras hacia adelante del punto de referencia para que los vectores con las características extraídas sean de la misma longitud y que contengan la información completa de las ondas P, QRS y T, tal como se presenta en la Figura 3.3, en conjunto con la Figura 3.4, que ilustra diferentes clases de latidos después de la segmentación descrita anteriormente. Este enfoque también se adopta en otros trabajos existentes [44] y [45].

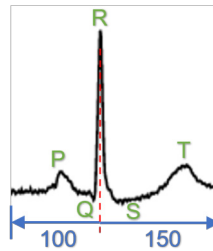


Figura 3.3: Estructura de una señal de ECG, con anotaciones para su posterior segmentación.

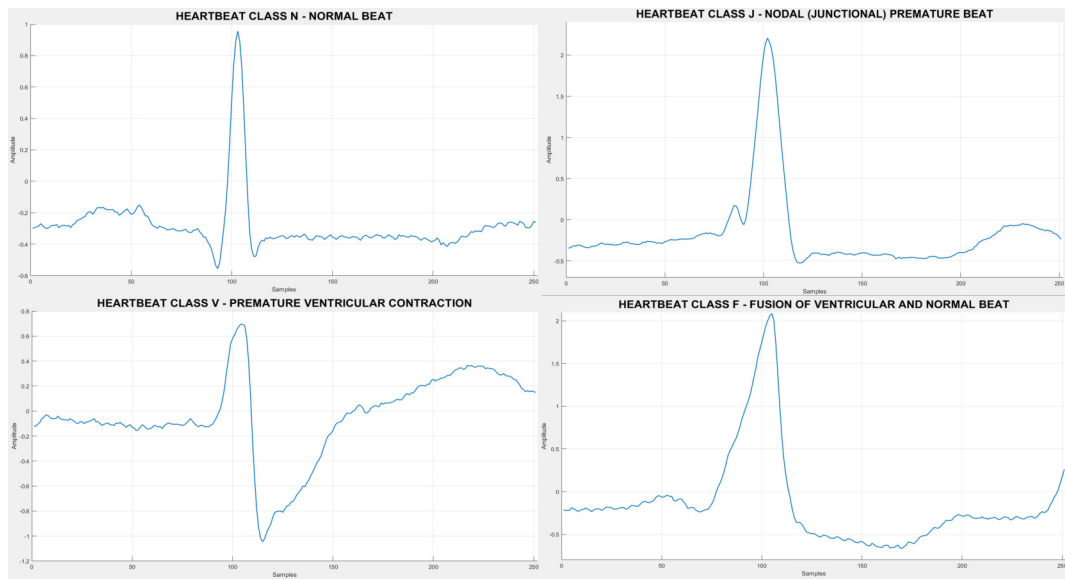


Figura 3.4: Diferentes clases de latidos del corazón después de la segmentación.

Grupo de latido cardíaco	Clases de latido cardíaco	Anotación latidos	Cantidad de latidos cardíacos	
Normal Total: 45801	N	N	38069	
	Total:	L	3949	
	45801	R	3783	
Anormal Total: 5179	SVEB Total: 976	e	16	
		j	16	
		A	810	
		a	100	
		J	32	
	VEB Total: 3788	S	2	
		V	3683	
		E	105	
	F Total: 415	F	F	415

Tabla 3.2: Clases de latidos del corazón, cantidad de latidos por clase y grupo para el conjunto de datos de entrenamiento (DS1).

### 3.3. Aplicación de la técnica SMOTE en las señales segmentadas

En la Tabla 3.2 se presenta la cantidad de latidos del conjunto de datos de entrenamiento (DS1), además se tiene en cuenta la cantidad de datos que pertenecen al tipo de señal ya sea normal o anormal, como se describe en la Sección 3.1.

De la Tabla 3.2 se observa el desbalance en la cantidad de latidos de las clases y en general de los grupos de la base de datos MIT-BIH, por tal motivo, se opta por aplicar la técnica SMOTE vista en la Subsección 2.5.6.

El algoritmo SMOTE lleva a cabo un enfoque de sobremuestreo para reequilibrar el conjunto de entrenamiento original. En lugar de aplicar una simple réplica de las instancias de la clase minoritaria, la idea clave de SMOTE es introducir ejemplos sintéticos. Estos nuevos datos se crean por interpolación entre varias instancias de clases minoritarias que se encuentran dentro de un vecindario.

El algoritmo se presenta en la Tabla 3.3, el cual funciona de la siguiente manera. Primero, se configura la cantidad total de sobremuestreo  $N$  (un valor entero), que puede configurarse para obtener una distribución de clases aproximada de 1:1 o descubrirse a través de un proceso de envoltura. Luego, se lleva a cabo un proceso iterativo, compuesto por varios pasos. Donde se selecciona aleatoriamente una instancia de clase minoritaria

Algoritmo SMOTE	
1	function SMOTE(T,N,k)
	<b>Input:</b> T; N; k #ejemplos de la clase minoritaria, cantidad de sobremuestreo, #vecinos más cercanos
	<b>Output:</b> (N/100)*T muestras sintéticas de la clase minoritaria
	<b>Variables:</b> Sample[][]: matriz para las muestras originales de la clase minoritaria;
	newindex: mantiene un conteo del número de muestras sintéticas generadas, inicializado en 0;
	Synthetic[][]: matriz para las muestras sintéticas
2	<b>if</b> N < 100 <b>then</b>
3	#Aleatorizar las T muestras de la clase minoritaria.
4	T = (N/100)*T
5	N = 100
6	<b>end if</b>
7	N = (int)N/100 #Se asume que la cantidad de SMOTE es en múltiplos enteros de 100.
8	<b>for</b> i = 1 <b>to</b> T <b>do</b>
9	#Calcular los k vecinos más cercanos para i y guardar los índices en el arreglo nnarray
10	POPULATE(N, i, nnarray)
11	<b>end for</b>
12	<b>end function</b>

Tabla 3.3: Algoritmo SMOTE [13].

del conjunto de entrenamiento. A continuación, se obtienen sus K-vecinos más cercanos (5 por defecto). Finalmente, N de estas K instancias se eligen aleatoriamente para calcular las nuevas instancias por interpolación [13].

Entonces, para generar las nuevas muestras a partir de los latidos existentes de cada clase, se utiliza un código en Matlab siguiendo como ejemplo el algoritmo presentado en la Tabla 3.3. Se ajustan los parámetros del algoritmo y se generan así nuevas muestras, la nueva cantidad de datos por clase y grupo se muestra en la Tabla 3.4.

Como se observa en la Tabla 3.4 se ha elegido la misma cantidad de latidos para cada tipo, es decir, la misma cantidad de datos normales y anormales, esto con la finalidad de obtener una mejor respuesta de la red neuronal en la primera etapa tal como se explicó en la Sección 3.1.

### 3.4. Codificación de las señales segmentadas

Del proceso anterior, segmentación, en el cual, se han separado los latidos individualmente, almacenados y agrupados por clases, se procede a realizar la respectiva codificación de las señales analógicas a trenes de impulsos o spikes. De esta forma, se obtienen diferentes archivos con los latidos convertidos a spikes de cada clase, organi-



Grupo de latido cardíaco	Clases de latido cardíaco	Anotación latidos	Cantidad de latidos cardíacos
Normal	N	N	50000
Total: 140000	Total: 140000	L R	45000 45000
Anormal Total: 140000	SVEB	e	10000
	Total: 60000	j A a	10000 10000 10000
	VEB	J	10000
	Total: 50000	S V E	10000 25000 25000
	F	F	30000
	Total: 30000		

Tabla 3.4: Cantidad de latidos por clase y grupo para el conjunto de datos de entrenamiento (DS1), después de aplicar SMOTE.

zando mejor los datos, finalmente, se crea un solo archivo, el cual contiene todos los datos de las spikes separados por clase.

Como se presenta en la Subsección 2.5.7, existen tres esquemas de codificación para señales analógicas, codificación de población, de frecuencia y temporal. En la Tabla 2.1 se presenta que, los algoritmos con mayor efectividad pertenecen al esquema de codificación temporal, como Step Forward (SF) y Temporal-Based Representation (TBR); en la escala le siguen, los esquemas de codificación de frecuencia Hough Spike Algorithm (HSA) y Ben's Spike Algorithm (BSA), al final, se encuentra el esquema de codificación de población.

Por lo anterior, se seleccionan algoritmos de codificación, pertenecientes a los dos esquemas con mayor efectividad, es decir, al temporal y al de frecuencia, esto brinda la opción de realizar una comparación entre los resultados obtenidos por el clasificador desarrollado en esta tesis (la red SNN), de acuerdo, con cada algoritmo de codificación elegido

Se desarrolla un programa para cada algoritmo de codificación mencionado anteriormente, estos se basan en los pseudocódigos presentados en [9] y que se muestran en las Tabla 3.5 y 3.6.

A modo de ejemplo se presentan una señal y sus codificaciones en las Figuras 3.5 a 3.7. En la Figura 3.5, se muestra un latido de corazón de la señal ECG, específicamente

Esquema de codificación de frecuencia (Rate coding schemes)	
Algoritmo de pulsos de Hough (Hough Spike Algorithm)	Algoritmo de pulsos de Ben (Ben's Spike Algorithm)
<pre> Data: input, filter Result: spikes, shift 1 L ← Length(input) 2 F ← Length(filter) 3 spikes ← zeros(1,L) 4 shift ← min(input) 5 input ← input - shift 6 for i = 1:L do 7   count ← 0 8   for j = 1:F do 9     if i+j+1 ≤ L &amp; input(i+j-1) ≥ filter(j) then 10      count += 1 11    end if 12  end for 13  if count == F then 14    spikes(i) = 1 15    for j = 1:F do 16      if i+j+1 ≤ L then 17        input(i+j-1) = input(i+j-1) - filter(j) 18      end if 19    end for 20  end if 21 end for </pre>	<pre> Data: input, filter, threshold Result: spikes, shift 1 L ← Length(input) 2 F ← Length(filter) 3 spikes ← zeros(1, L) 4 shift ← min(input) 5 input ← input - shift 6 for i = 1:(L-F) do 7   err1, err2 ← 0 8   for j = 1:F do 9     err1 = err1 + abs(input(i+j) - filter(j)) 10    err2 = err1 + abs(input(i+j-1)) 11  end for 12  if err1 ≤ err2 * threshold then 13    spikes(i) = 1 14    for j = 1:F do 15      if i+j+1 ≤ L then 16        input(i+j+1) = input(i+j+1) - filter(j) 17      end if 18    end for 19  end if 20 end for </pre>

Tabla 3.5: Algoritmo HSA y BSA, esquema de codificación de frecuencia [9].

del registro 100 de la base de datos MIT-BIH [11], en la Figura 3.6, se presenta la señal codificada utilizando el esquema de codificación temporal (SF Y TBR), en la Figura 3.7, se ilustra el resultado de un esquema de codificación de frecuencia (HSA y BSA).

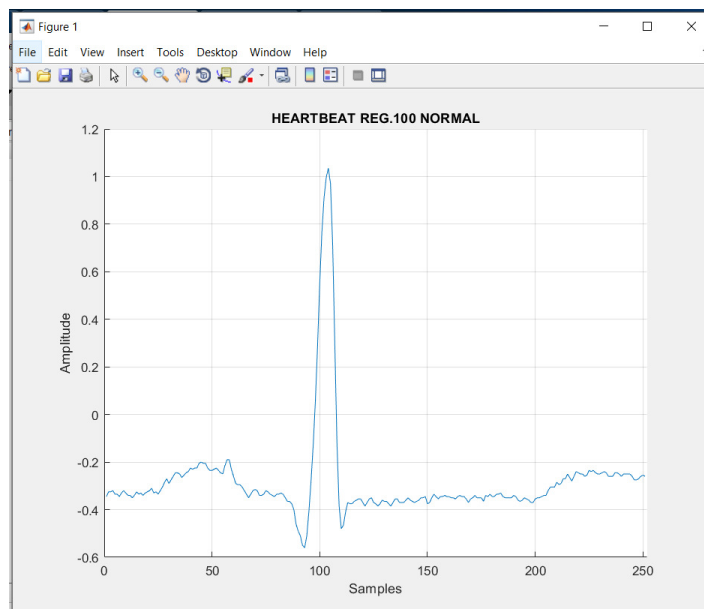


Figura 3.5: Latido del corazón, perteneciente al registro 100 de la base de datos MIT-BIH.

Esquema de codificación temporal (Temporal coding schemes)	
Algoritmo de Paso Adelante (Step Forward Algorithm)	Representación Basada en el Tiempo (Temporal-Based Representation)
<p><b>Data:</b> <math>input, threshold</math>  <b>Result:</b> <math>spikes, init</math></p> <pre> 1 <math>L \leftarrow \text{Length}(input)</math> 2 <math>spikes \leftarrow \text{zeros}(1, L)</math> 3 <math>init, base \leftarrow input(1)</math> 4 <b>for</b> <math>i = 2:L</math> <b>do</b> 5   <b>if</b> <math>input(i) &gt; base + threshold</math> <b>then</b> 6     <math>spikes(i) = 1</math> 7     <math>base = base + threshold</math> 8   <b>else if</b> <math>input(i) &lt; base - threshold</math> <b>then</b> 9     <math>spikes(i) = -1</math> 10    <math>base = base - threshold</math> 11  <b>end if</b> 12 <b>end for</b> </pre>	<p><b>Data:</b> <math>input, a</math>  <b>Result:</b> <math>spikes, threshold</math></p> <pre> 1 <math>L \leftarrow \text{Length}(input)</math> 2 <math>spikes \leftarrow \text{zeros}(1, L)</math> 3 <math>diff \leftarrow \text{zeros}(1, L-1)</math> 4 <b>for</b> <math>i = 1:L-1</math> <b>do</b> 5   <math>diff(i) = input(i+1) - input(i)</math> 6 <b>end for</b> 6 <math>threshold \leftarrow \text{mean}(diff) + a \cdot \text{std}(diff)</math> 7 <math>diff \leftarrow [diff(1), diff]</math> 8 <b>for</b> <math>i = 1:L</math> <b>do</b> 5   <b>if</b> <math>diff(i) &gt; threshold</math> <b>then</b> 6     <math>spikes(i) = 1</math> 8   <b>else if</b> <math>diff(i) &lt; -threshold</math> <b>then</b> 9     <math>spikes(i) = -1</math> 11  <b>end if</b> 12 <b>end for</b> </pre>

Tabla 3.6: Algoritmo SF y TBR, esquema de codificación temporal [9].

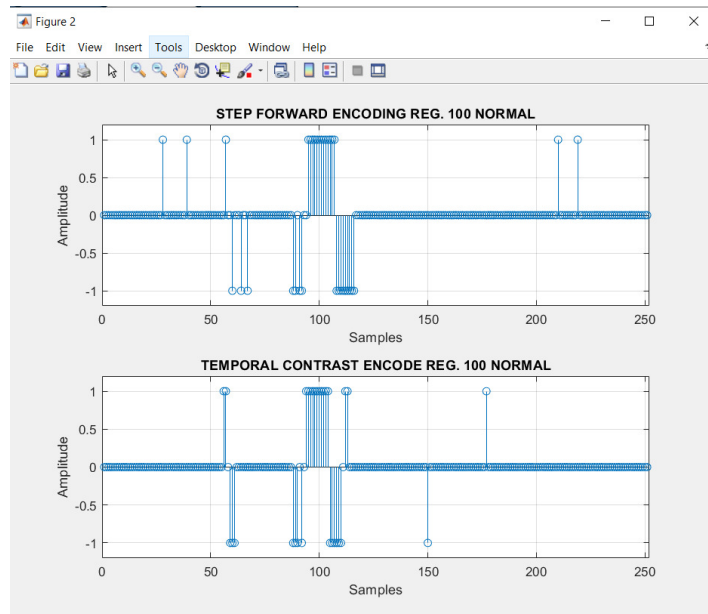


Figura 3.6: Codificación de la señal de la Figura 3.5, esquema de codificación temporal.

### 3.5. Entrenamiento de la red neuronal pulsante (SNN)

Para el entrenamiento de la red neuronal pulsante (SNN-d), se utiliza el modelo propuesto en la Subsección 2.5.5, en él, se incorporan retrasos sinápticos (synaptic delays) utilizando el siguiente enfoque, que es compatible con los marcos de retropropagación clásicos utilizados hoy en día para entrenar redes neuronales pulsantes (SNNs) y redes neuronales recurrentes (RNNs) (es decir, sin extensiones especiales al marco). La idea

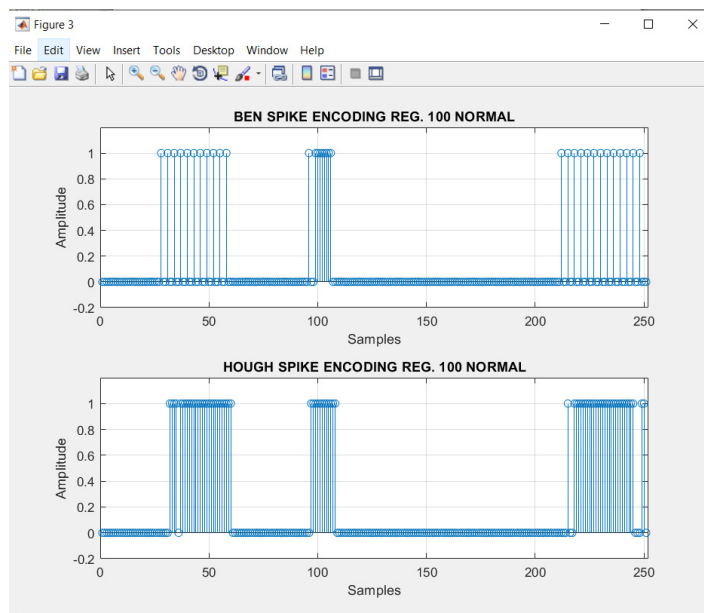


Figura 3.7: Codificación de la señal de la Figura 3.5, esquema de codificación de frecuencia.

es expresar la parametrización (temporal) de los retrasos como una parametrización espacial de los pesos sinápticos, de modo que el entrenamiento de los retrasos se logra simplemente optimizando los pesos sinápticos. Comenzamos con un conjunto de sinapsis paralelas por par de neuronas pre y post-sinápticas, cada una asociada con una salida retrasada de la neurona pre-sináptica (utilizando un rango de retrasos y un paso predeterminados). Optimizamos el modelo como de costumbre y eliminamos todas las sinapsis de retraso que acaban teniendo pesos pequeños. Luego se ajusta el modelo con sólo las sinapsis restantes.

Podemos introducir nuevas sinapsis para reemplazar las eliminadas, con una resolución de retraso incrementalmente mayor en subregiones localizadas del rango inicial de retraso, y repetir el proceso. Como resultado, diferentes neuronas terminan con diferentes entradas de retraso de fan-in (número de conexiones entrantes). Los modelos resultantes son estructuras topológicamente similares a una red neuronal de propagación hacia adelante, consistentemente más simples con menos parámetros que sus contrapartes de conectividad recurrente, y muestran un rendimiento de vanguardia. Su estructura más simple las hace llamativas para una implementación eficiente en términos de recursos en aceleradores neuromórficos.

Se entrenan los modelos de SNN con retropropagación (específicamente STBP [46]). Este método tiene en cuenta la influencia pasada en los estados de las neuronas actuales desenrollando la red en el tiempo, y los errores se calculan a lo largo de las rutas inversas de la red desenrollada. Para tener en cuenta la discontinuidad del potencial de membrana, se utiliza una función de gradiente sustituto [47] con una función sigmoide

rápida como en [48]. Durante el entrenamiento, además de los pesos sinápticos, también se optimiza las constantes de tiempo de la membrana, como en [49]. Por último, no se considera retardos adicionales para la entrada en la primera capa, ya que la capa de entrada generalmente tiene más neuronas y es responsable de una gran parte de los parámetros sinápticos.

En las Figuras 3.8 y 3.9, se interpreta gráficamente la conexión entre neuronas para el entrenamiento (Training Delays).

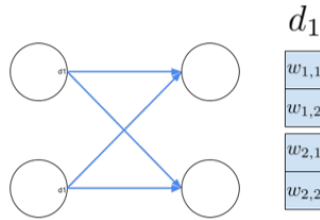


Figura 3.8: Pesos sinápticos.

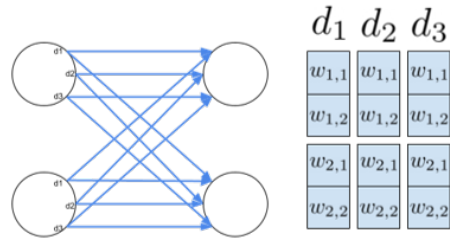


Figura 3.9: Pesos sinápticos con retrasos.

### Parámetros de la red

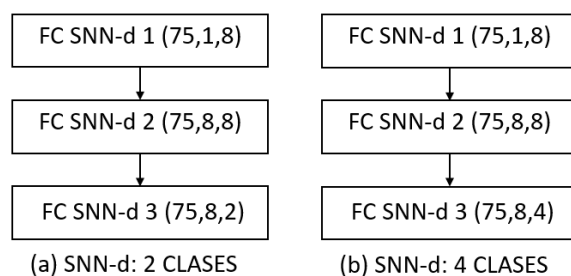
El modelo SNN-d de dos clases presentado en la Figura 3.10(a) está entrenado con 2 capas de 8 neuronas cada una y una capa de salida. Además, se cuenta con 75 retrasos por capa, un paso de 2 y una profundidad de 150. El tamaño de entrada de la primer capa SNN es la longitud del latido del corazón, que es de 251 puntos de muestreo. Con la finalidad de clasificar las señales de ECG en dos clases (normal y anormal) para cada codificación determinada. La tasa de aprendizaje del modelo SNN de 2 clases es 0.0001, un tamaño de lote de 1024 y es entrenado con 100 épocas.

El modelo SNN-d de cuatro clases presentado en la Figura 3.10(b) está diseñado para clasificar en especial las señales de ECG anormales de la red precedente de dos clases en cuatro clases detalladas (N, SVEB, VEB, F). Aquí todavía se incluye la clase normal para corregir la clasificación errónea de la red anterior. Al igual que la red de dos clases, esta red cuenta con 2 capas cada una con 8 neuronas y una capa de salida,

Etapas del modelo	Cantidad de parámetros
Primera etapa: 2 clases	6600
Segunda etapa: 4 clases	7800
Total	14400

Tabla 3.7: Cantidad de parámetros del modelo propuesto.

con 75 retrasos por capa, un paso de 2 y una profundidad de 150. La tasa de aprendizaje del modelo SNN de 4 clases es 0.0001, un tamaño de lote de 1024 y es entrenado con 100 épocas.



FC SNN-d(x1,x2,y2) donde x1 son los retardos o delays, x2 es la entrada del canal y y2 es la salida del canal.

Figura 3.10: Estructura de la red SNN-d para la clasificación de (a) dos y (b) cuatro clases.

En resumen, en la Tabla 3.7 se presenta la cantidad de parámetros de cada etapa y la cantidad de parámetros total de todo el modelo.

# Capítulo 4

## Resultados

En este capítulo, se presentan los resultados obtenidos por la red neuronal pulsante con retrasos (Subsección 2.5.5), al ingresar las señales de la base de datos de MIT-BIH ECG, previamente segmentadas y codificadas, como se vio en el Capítulo 3.

Para la evaluación del método, se utilizan las métricas recomendadas por AAMI [40], tales como: Sensibilidad (Sensitivity (Se)), predictividad positiva (Positive predictivity (+P)), tasa de falsos positivos (False positive rate (FPR)) y precisión general (Overall accuracy (Acc)).

La sensibilidad y la predictividad positiva, también se conocen en la literatura como recall y precision, respectivamente.

### 4.1. Análisis de resultados

Como se explicó en la Sección 3.1, se desarrolló un método de clasificación con dos etapas, la primera etapa utilizando una red SNN, permite seleccionar los latidos como normales o anormales, en la segunda etapa, también con una red SNN, se clasifican las 4 clases, pertenecientes al grupo de anormales.

En las presentes secciones, se da a conocer el desempeño de la red neuronal de impulsos, como objetivo principal, se realiza una comparación de las 4 codificaciones (SF, TBR, HSA y BSA), vistas en la Subsección 2.5.7, de esta manera, se conoce cual ofrece mejores resultados al momento de trabajar con redes SNN.

Siguiendo las recomendaciones de la AAMI [40], las métricas de evaluación para el rendimiento del clasificador de señales ECG, presentado en esta tesis son: Precisión (Acc, Accuracy), Sensibilidad (Se, Sensitivity o Recall), Predictividad positiva (P+, Positive Predictivity o Precision) y Puntuación F1 (F1 score). Se utilizan los términos en inglés ya que son de uso común en los ambientes técnicos. Estas métricas se definen de la siguiente manera [10]:

$$Sensitivity(Se) = \frac{T_P}{T_P + F_N} \quad (4.1)$$

$$PositivePredictivity(P+) = \frac{T_P}{T_P + F_P} \quad (4.2)$$

$$Accuracy(Acc) = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \quad (4.3)$$

$$F1score(F1) = \frac{2 * Precision * Recall}{Precision + Recall} \quad (4.4)$$

donde, TP son verdaderos positivos, TN verdaderos negativos, FP falsos positivos y FN falsos negativos.

El puntaje F1 considera de manera integral, la precisión y la recuperación del modelo de clasificación, con frecuencia, se toma como la métrica de evaluación final.

Una de las formas de conocer TP, TN, FP y FN, es a partir de la matriz de confusión, la cual es proporcionada como uno de los resultados de la red neuronal.

En la Figura 4.1, se presenta un ejemplo de cómo obtener el valor de cada parámetro con la matriz de confusión para dos clases y en la Figura 4.2 se presenta el ejemplo cuando se tiene una matriz con N clases.

	Positive	Negative
Positive	True Positive (TP)	False Negative (FN)
Negative	False Positive (FP)	True Negative (TN)

Figura 4.1: Matriz de confusión para dos clases.

### Primera etapa: 2 clases

Con lo anterior es posible, encontrar de forma correcta, las métricas de evaluación para los resultados presentados por la red neuronal, como lo es la matriz de confusión (MC). En la Tabla 4.1, se ilustra la MC obtenida para cada una de las 4 codificaciones. Los resultados de las métricas de evaluación se exponen en la Tabla 4.2.



Matriz de confusión - MC(i,j) de NXN

		Clase I					Clase N-I					
REAL 1		TP	FN	FN	FN	FN	REAL 1	TN	TN	TN	FP	TN
REAL 2		FP	TN	TN	TN	TN	REAL 2	TN	TN	TN	FP	TN
REAL ...		FP	TN	...	TN	TN	REAL ...	TN	TN	...	FP	TN
REAL N-1		FP	TN	TN	TN	TN	REAL N-1	FN	FN	FN	TP	FN
REAL N		FP	TN	TN	TN	TN	REAL N	TN	TN	TN	FP	TN
		ESTIMADO 1	ESTIMADO 2	ESTIMADO ...	ESTIMADO N-1	ESTIMADO N		ESTIMADO 1	ESTIMADO 2	ESTIMADO ...	ESTIMADO N-1	ESTIMADO N

Figura 4.2: Matriz de confusión para N clases.

CODIFICACIÓN	MATRIZ DE CONFUSIÓN
Step Forward Algorithm (SF)	$\begin{pmatrix} 32411 & 11604 \\ 1600 & 4059 \end{pmatrix}$
Temporal-Based Representation (TBR)	$\begin{pmatrix} 28678 & 15337 \\ 2224 & 3435 \end{pmatrix}$
Ben's Spike Algorithm (BSA)	$\begin{pmatrix} 25585 & 18430 \\ 2447 & 3212 \end{pmatrix}$
Hough Spike Algorithm (HSA)	$\begin{pmatrix} 25721 & 18294 \\ 2480 & 3179 \end{pmatrix}$

Tabla 4.1: MC de la SNN, al ingresar los datos codificados respectivamente.

Codificación	Normal		Anormal		F1 Score	Overall Accuracy
	Se	P+	Se	P+		
SF	73.6 %	95.3 %	71.7 %	25.9 %	60.6 %	73.2 %
TBR	65.2 %	92.8 %	60.7 %	18.3 %	52.3 %	64.7 %
BSA	58.1 %	91.3 %	56.8 %	14.8 %	47.3 %	58.0 %
HSA	58.4 %	91.2 %	56.2 %	14.8 %	47.3 %	58.2 %

Tabla 4.2: Porcentaje de Sensibilidad, Predictividad positiva, F1 y Precisión para las clases Normal y Anormal.

### Segunda etapa: 4 clases

Después que se han clasificado los datos como normales o anormales, se realiza una segunda etapa de clasificación; ésta consta de 4 clases (N, SVEB, VEB y F). En la Tabla 4.3 se presenta la matriz de confusión de cada codificación, siguiendo el proceso descrito en la Figura 4.2, se encuentran los valores de las métricas, los datos se presentan en la Tabla 4.4.

CODIFICACIÓN	MATRIZ DE CONFUSIÓN
Step Forward Algorithm (SF)	$\begin{pmatrix} 5730 & 4547 & 261 & 1066 \\ 508 & 353 & 73 & 12 \\ 335 & 43 & 2216 & 290 \\ 87 & 2 & 122 & 18 \end{pmatrix}$
Temporal-Based Representation (TBR)	$\begin{pmatrix} 6090 & 5319 & 514 & 3414 \\ 198 & 140 & 32 & 55 \\ 199 & 66 & 2210 & 233 \\ 77 & 64 & 133 & 28 \end{pmatrix}$
Ben's Spike Algorithm (BSA)	$\begin{pmatrix} 7530 & 5486 & 2812 & 2602 \\ 439 & 503 & 48 & 11 \\ 331 & 61 & 1264 & 230 \\ 145 & 6 & 169 & 5 \end{pmatrix}$
Hough Spike Algorithm (HSA)	$\begin{pmatrix} 4614 & 6837 & 5164 & 1679 \\ 748 & 540 & 257 & 16 \\ 221 & 32 & 961 & 204 \\ 16 & 4 & 44 & 136 \end{pmatrix}$

Tabla 4.3: Matriz de confusión resultante de la SNN, al ingresar los datos codificados respectivamente.

Codificación	N		SVEB		VEB		F		F1	Overall
	Se	P+	Se	P+	Se	P+	Se	P+	Score	Acc.
SF	49.4 %	86.0 %	37.3 %	7.1 %	76.8 %	82.9 %	7.9 %	1.3 %	39.2 %	82.0 %
TBR	39.7 %	92.8 %	32.9 %	2.5 %	81.6 %	76.5 %	9.3 %	0.8 %	35.2 %	74.8 %
BSA	40.9 %	89.2 %	50.3 %	8.3 %	67.0 %	29.4 %	1.5 %	0.2 %	27.9 %	70.2 %
HSA	25.2 %	82.4 %	34.6 %	7.3 %	67.8 %	15.0 %	68.0 %	6.7 %	21.8 %	64.4 %

Tabla 4.4: Porcentaje de Sensibilidad, Predictividad Positiva, F1 y Precisión para las clases N, SVEB, VEB y F.

### Comparativa entre modelo y antecedentes

En la Tabla 3.7 de la Sección 3.5 se presenta la cantidad de parámetros del modelo propuesto. Si se compara, con respecto a otras redes neuronales que se han utilizado también para la clasificación de señales ECG, como por ejemplo, redes neuronales convolucionales (CNN), se encuentra que estas ultimas utilizan una mayor cantidad de parámetros.

Esta información se puede corroborar en [10], donde utilizan un modelo similar al que se aplicó en esta tesis, en la Figura 4.3 se observa la estructura de la CNN para la clasificación de dos y cuatro clases.

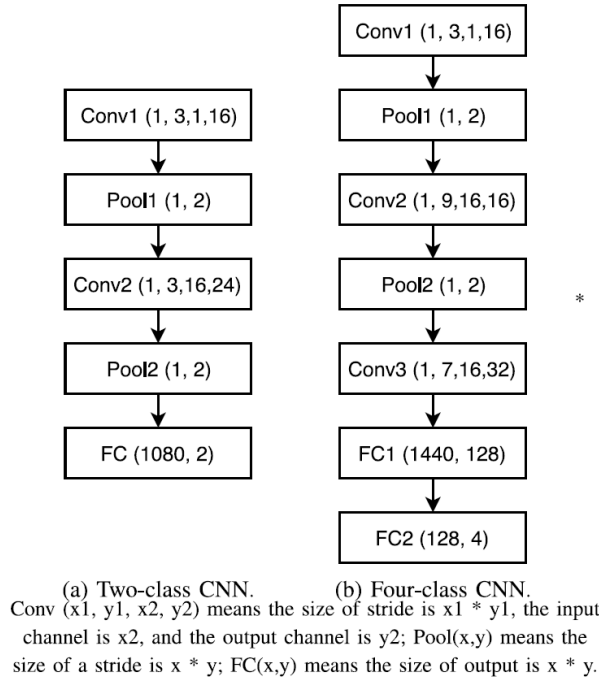


Figura 4.3: Estructura de la CNN para la clasificación de dos y cuatro clases [10].

Etapas del modelo	Cantidad de parámetros	
	Modelo SNN-d	Modelo CNN [10]
Primera etapa: 2 clases	6600	3360
Segunda etapa: 4 clases	7800	190768
Total	14400	194128

Tabla 4.5: Comparación de parámetros entre el modelo SNN Y CNN.

Modelo	Precisión general (Overall Accuracy)	
	Primera etapa: 2 clases	Segunda etapa: 4 clases
CNN [10]	81 %	74.76 %
SNN convertida [10]	79 %	73.78 %
SNN-d (Nuestra)	73 %	82 %

Tabla 4.6: Comparación de la precisión general entre el modelo CNN, SNN convertida Y SNN-d.

En la Tabla 4.5, se presenta la comparación entre la cantidad de parámetros tanto del modelo propuesto en la tesis como del modelo propuesto en [10]. Como se observa, existe una diferencia de 179728 parámetros del modelo CNN por encima de lo que utiliza el modelo SNN.

Entonces, la SNN utiliza un menor número de parámetros que una CNN y aún así presenta resultados similares, esto sugiere una mayor eficiencia en la representación de datos y un aprendizaje más efectivo, lo que puede permitir un mejor rendimiento con menos recursos computacionales.

A partir de [10], se puede realizar la comparativa de la precisión general entre los modelos expuestos allí (CNN y SNN) y el modelo de red SNN-d; Cabe aclarar que, el modelo SNN que se expone en [10], se obtiene de un proceso de convertir la red CNN entrenada a una neuronal pulsante convolucional, las diferencias de precisión para las dos etapas se presentan en la Tabla 4.6.

La precisión general en la primera etapa de la SNN convertida es del 79 %, mientras que en la red CNN es del 81 %. En la segunda etapa, la precisión de la SNN convertida es del 73.78 % mientras que es del 74.76 % para CNN.

La red SNN-d expuesta en esta investigación, presenta una precisión general del 73 % utilizando la codificación SF para la primera etapa, tal como se observa en la Tabla 4.2 y una precisión general del 82 % para la segunda etapa, como se presenta en la Tabla 4.4. La diferencia entre el resultado de la red SNN [10] con la actual es apenas del 6 % por debajo para la primera etapa y del 8 % por encima en la segunda etapa.

---

Donde se corrobora que, este nuevo modelo de red neuronal pulsante tiene resultados muy cercanos a los modelos de redes neuronales convolucionales y a modelos de redes SNN convertidos, los cuales han sido estudiados y fuertemente consolidados. Cabe recordar que el principal punto a destacar es que, el modelo actual utiliza una menor cantidad de parámetros como se presentó en la Tabla 4.5.

Aunque la red SNN-d tenga una arquitectura más básica, su enfoque en la temporalidad y los patrones de disparo puede permitir una extracción de características más discriminativas. En lugar de depender exclusivamente de capas convolucionales como en una CNN, las SNN-d pueden aprovechar el procesamiento temporal y la comunicación entre neuronas para capturar información relevante. Esto puede permitir que la red capture características importantes incluso con una arquitectura más sencilla, es decir, menos cantidad de parámetros.

### **Conclusiones de la sección**

En la Tabla 4.1 se presenta la matriz de confusión para cada codificación, resultado del clasificador, cuando la entrada es la base de datos MIT-BIH dividida en dos conjuntos, Normal y Anormal, correspondientes a la primera etapa del modelo. A partir de la matriz de confusión para cada codificación, se hallan las métricas de evaluación recomendadas por la AAMI [40], dadas en la Tabla 4.2.

Los resultados para la primera etapa confirman lo expuesto en la documentación [9], donde el esquema de codificación temporal es el más eficiente con respecto a los demás. Del cual, se encuentra el algoritmo denominado Step Forward (SF), cuyo porcentaje de precisión general es alto y cercano al 73.2%, seguido se encuentra el algoritmo Temporal-Based Representation (TBR) con una diferencia de 8.5%, es decir, un porcentaje de precisión del 64.7%. Respecto el esquema de codificación de frecuencia, se presenta el algoritmo Ben's Spike Algorithm (BSA), el cual, presenta un porcentaje de precisión cercano al 58%, en esta misma escala se encuentra el algoritmo Hough Spike Algorithm (HSA), con tan solo una diferencia de 0.2% y contrario a como se había estipulado por la documentación los algoritmos BSA y HSA presentan resultados muy similares, demostrando que no existe un rango de diferencia como se había estipulado en [9], para bioseñales, específicamente para señales de ECG.

De igual manera se analiza la Tabla 4.4, en la cual, se presentan los resultados para la segunda etapa, diseñada para clasificar las señales de ECG anormales de la red de dos clases, éstas se dividen en cuatro clases detalladas (N, SVEB, VEB, F), todavía se incluye la clase normal para corregir la clasificación errónea de la red precedente. Para esta última etapa, se encuentra al igual que en la primera, el esquema de codificación temporal es el que brinda mejores resultados, el algoritmo SF presenta una precisión general de 82%, con una diferencia de 7.2% del siguiente algoritmo TBR con una precisión general de 74.8%, por último, se presenta los algoritmos pertenecientes al esquema de codificación de frecuencia, BSA con 70.2% y HSA con 64.4%.

---

En resumen, se concluye que elegir el algoritmo de codificación adecuado al momento de trabajar con bioseñales, es de suma importancia, tal como se presenta en los resultados de esta tesis, ya que la diferencia de precisión general entre un algoritmo de codificación a otro puede ser superior del 20 % (comparando el algoritmo SF con respecto al algoritmo HSA). También se concluye que los esquemas de codificación temporal, hoy en día, son los más utilizados por su eficiencia y accesibilidad.

# Capítulo 5

## Conclusiones

En esta tesis se presentó una aplicación de las redes neuronales pulsantes en la clasificación de señales de electrocardiografía. El uso de este tipo de redes en aplicaciones reales, es aún muy escaso por lo que esta tesis puede servir para el desarrollo de futuros trabajos en el campo de la biomedicina.

De manera particular, se presentó una red de reciente estudio, denominada red neuronal pulsante con retrasos axonales. En ella se aplicó a una estructura multietapa que permite realizar la clasificación de señales de ECG. Esta topología se diseñó con el fin de disminuir de forma efectiva el consumo de energía en el sistema.

Desde el punto de vista energético, se puede identificar que, a diferencia de las redes neuronales tradicionales, que realizan operaciones de cálculo continuas y usan energía incluso en períodos de inactividad, las SNN se basan en la propagación de impulsos discretos y solo consumen cuando se produce una actividad neuronal. Esto permite un procesamiento basado en eventos y reduce la cantidad de cálculos requeridos, lo que a su vez reduce la necesidad de energía.

Además, uno de los objetivos principales de la tesis era el dar relevancia a la codificación de las señales, ya que las SNN codifican información en el tiempo mediante la emisión de pulsos. En lugar de utilizar representaciones numéricas continuas, como en las redes neuronales tradicionales, las SNN transmiten información en forma de eventos discretos. Esta codificación temporal puede ser más eficiente en términos de energía, ya que solo se activan las neuronas relevantes en momentos específicos, lo que reduce la cantidad total de operaciones y el consumo de energía asociado.

En cuanto a la codificación, existen diferentes estudios de los tipos de codificación de señales analógicas a trenes de impulsos. Como ejemplo se expuso la Tabla 2.1, en la cual, se presentan los resultados de pruebas que hicieron en [12] para medir la eficiencia de cada esquema de codificación, sin embargo, pocos estudios han realizado y documentado la eficiencia de esquemas, aplicados a bioseñales (caso actual, señales de Electrocardiografía), al momento de usarlos en clasificadores donde se hace necesario

la codificación de la entrada, tal como lo son las redes neuronales de impulsos SNN.

Con esta motivación, se diseñó una estructura multietapa, la cual, cuenta con una red neuronal de impulsos con retrasos para cada etapa como se observó en la Subsección 2.5.5, y tiene presente como única entrada la base de datos con registros de electrocardiografía (ECG) denominada MIT-BIH [11] previamente codificada a trenes de impulsos como se observó en la Sección 3.4, recordando que se presentan 2 esquemas de codificación y por cada uno, se eligieron dos algoritmos, siendo los más recomendados para este proceso [9], en total son 4 métodos de codificación. Para poder realizar la comparación con las diferentes codificaciones, se establecen los parámetros de la red neuronal, los cuales no serán modificados posteriormente, una vez concluido el modelo, se introduce la entrada codificada al clasificador, y al cabo de 100 épocas de procesamiento, se almacena la respuesta dada por éste, especialmente la matriz de confusión, siendo la que permite obtener métricas de evaluación (Capítulo 4), se repite este proceso para cada una de las otras 3 codificaciones.

De forma general el modelo de red neuronal que se propuso presentó resultados positivos para la investigación, los cuales se acercan a modelos de redes neuronales ya consolidados pero que presentan una estructura demasiado elaborada. La red con retrasos cuenta con apenas un pequeño conjunto de pausas sinápticas, junto con constantes de tiempo entrenables y que reemplazan la necesidad de conectividad lateral compleja, reducen la cantidad de capas y la cantidad total de parámetros necesarios para un buen rendimiento, es decir, con un modelo de apenas dos capas y 8 neuronas por capas, se obtienen resultados cercanos a modelos con estructuras más complejas, con muchas capas, muchas neuronas y demasiados parámetros, como se demostró en la Sección 3.5 y 4.1.

## Trabajo Futuro

La gran mayoría de los trabajos en la literatura no se preocupan por seguir la división definida en [40] o cualquier otro protocolo inter-paciente que imponga la no utilización de los latidos del corazón del mismo paciente en el entrenamiento y evaluación.

Esto hace que el modelo presentado en esta tesis sea adecuado para la implementación. Entonces como trabajo futuro se evaluará diseñar e instrumentar un dispositivo portátil capaz de registrar y clasificar las señales ECG, aprovechando el desarrollo del hardware neuromórfico, creando así un sistema eficiente para el monitoreo “activo siempre” y la detección de anomalías, lo que podría ser una herramienta práctica para el sector de la salud.

El mejoramiento de la red neuronal utilizando más capas o más neuronas, en cuyo caso se necesitaría de un procesamiento mayor por lo tanto se necesitaría un computador con mejores prestaciones.

Probar el modelo con otros tipos de bioseñales, como por ejemplo señales de electro-



encefalografía (EEG), para tener más confianza en el esquema de codificación temporal y demostrar a su vez la eficiencia de esta red neuronal con retrasos.

También como trabajo futuro sería interesante explorar la implementación de este modelo en hardware neuromórfico para evaluar el rendimiento real de este tipo de implementaciones.

# Bibliografía

- [1] Loscalzo J., Fauci A., Kasper D., Hauser S., Longo D., and Jameson J. Harrison's Principles of Internal Medicine, volume 21e. McGraw Hill, 2022.
- [2] Bistel E. R. and Fajardo M. A. Diseño de un sistema de adquisición y procesamiento de la señal de ecg basado en instrumentación virtual. Ingeniería Electrónica, Automática y Comunicaciones, 36(1):17–30, 06 2015.
- [3] S. Ramón. Textura del sistema nervioso del hombre y de los vertebrados: estudios sobre el plan estructural y composición histológica de los centros nerviosos adicionados de consideraciones fisiológicas fundadas en los nuevos descubrimientos. Moya, 1899.
- [4] Kasabov N. K. Time-Space, Spiking Neural Networks and Brain-Inspired Artificial Intelligence. Springer-Verlag GmbH Germany, part of Springer Nature 2019, 2019.
- [5] Izhikevich E. M. Which model to use for cortical spiking neurons?. IEEE transactions on neural networks, 15(5):1063–1070, 2004.
- [6] Página principal Bernstein Network Computational Neuroscience. <https://bernstein-network.de/forschung-und-lehre/studienangebote/online-learning/theoretical-neuroscience-i/>, Accedido el 01-02-2023.
- [7] Lee C., Sarwar S. S., Panda P., Srinivasan G., and K. Roy. Enabling spike-based backpropagation for training deep neural network architectures. Frontiers in neuroscience, 119:1, 2020.
- [8] Patiño A., Yousefzadeh A., Tang G., Corradi F., Linares B., and Sifalakis M. Empirical study on the efficiency of spiking neural networks with axonal delays, and algorithm-hardware benchmarking. ISCAS, 1:1, 2023.
- [9] Dupeyroux J., Stroobants S., and De Croon G. C. A toolbox for neuromorphic perception in robotics. In 2022 8th International Conference on Event-Based Control, Communication, and Signal Processing (EBC CSP), 1:1–7, 2022.
- [10] Yan Z., Zhou J., and Wong W. F. Energy efficient ecg classification with spiking neural network. Biomedical Signal Processing and Control, 63:102170, 2021.

- [11] Página principal de la base de datos MIT-BIH. <https://physionet.org/content/mitdb/1.0.0/>, Accedido el 01-02-2023.
- [12] Carr C. E. Processing of temporal information in the brain. Annual review of neuroscience, 16(1):223–243, 1993.
- [13] Fernández A., Garcia S., Herrera F., and Chawla N. V. Smote for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. Journal of artificial intelligence research, 61:863–905, 2018.
- [14] Murat F., Yildirim O., Talo M., Baloglu U. B., Demir Y., and Acharya U. R. Application of deep learning techniques for heartbeats detection using ecg signals-analysis and review. Computers in biology and medicine, 120:103726, 2020.
- [15] He Y. and Zhao J. Temporal convolutional networks for anomaly detection in time series. Journal of Physics: Conference Series, 1213(4):042050, 6 2019.
- [16] Faziludeen S. and Sabiq P. V. Ecg beat classification using wavelets and svm. IEEE Conference on Information and Communication Technologies, pages 815–818, 4 2013.
- [17] Willems J. L. and Lesaffre E. Comparison of multigroup logistic and linear discriminant ecg and veg classification. Journal of electrocardiology, 20(2):83–92, 1987.
- [18] Rajesh K. N. and Dhuli R. Classification of imbalanced ecg beats using re-sampling techniques and adaboost ensemble classifier. Biomedical Signal Processing and Control, 41:242–254, 2018.
- [19] Li H., Yuan D., Ma X., Cui D., and Cao L. Genetic algorithm for the optimization of features and neural networks in ecg signals classification. Scientific reports, 7(1):41011, 2017.
- [20] Ramkumar M., Babu C. G., Kumar K. V., Hepsiba D., Manjunathan A., and Kumar R. S. Ecg cardiac arrhythmias classification using dwt, ica and mlp neural networks. Journal of Physics: Conference Series, 1831:012015, 3 2021.
- [21] Saadatnejad S., Oveisi M., and Hashemi M. Lstm-based ecg classification for continuous monitoring on personal wearable devices. IEEE journal of biomedical and health informatics, 24(2):515–523, 2019.
- [22] Baloglu U. B., Talo M., Yildirim O., San Tan R., and Acharya U. R. Classification of myocardial infarction with multi-lead ecg signals and deep cnn. Pattern recognition letters, 122:23–30, 2019.
- [23] Li Y., Pang Y., Wang J., and Li X. Patient-specific ecg classification by deeper cnn from generic to dedicated. Neurocomputing, 314:336–346, 2018.

- [24] Mondéjar-Guerra V., Novo J., Rouco J., Penedo M. G., and Ortega M. Heartbeat classification fusing temporal and morphological information of egs via ensemble of classifiers. Biomedical Signal Processing and Control, 47:41–48, 2019.
- [25] Rai H. M., Trivedi A., and Shukla S. Ecg signal processing for abnormalities detection using multi-resolution wavelet transform and artificial neural network classifier. Measurement, 46(9):3238–3246, 2013.
- [26] Maass W. Networks of spiking neurons: the third generation of neural network models. Neural networks, 10(9):1659–1671, 10 1997.
- [27] Goldich G. El electrocardiograma de 12 derivaciones: Parte i: reconocimiento de los hallazgos normales. Nursing (Ed. Española), 32(2):28–34, 2015.
- [28] Jayaraman S., Sangareddi V., Periyasamy R., Joseph J., and Shanmugam RM. Modified limb lead ecg system effects on electrocardiographic wave amplitudes and frontal plane axis in sinus rhythm subjects. Anatol J Cardiol, 17(1):46–54, 01 2017.
- [29] Página principal de Physionet. <https://physionet.org/>, Accedido el 01-02-2023.
- [30] Página principal de Nature. [www.nature.com](http://www.nature.com), Accedido el 01-02-2023.
- [31] Página principal del módulo WFDB. <https://www.physionet.org/content/wfdb/10.7.0/>, Accedido el 01-02-2023.
- [32] Gerstner W and Kistler W. M. Spiking neuron models: Single neurons, populations, plasticity. Cambridge University Press, 2002.
- [33] Mead C. Analog VLSI and neural systems. Addison-Wesley Longman Publishing Co., 1989.
- [34] Gerstner W., Ritz R., and Van Hemmen J. L. Why spikes? hebbian learning and retrieval of time-resolved excitation patterns. Biological Cybernetics, 69 (5):503–515, 1993.
- [35] Ponulak F. and Kasinski A. Introduction to spiking neural networks: Information processing, learning and applications. Acta neurobiologiae experimentalis, 71(4):409–433, 2011.
- [36] Chawla N. V., Bowyer K. W., Hall L. O., and Kegelmeyer W. P. Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16:321–357, 2002.
- [37] Petro B., Kasabov N., and Kiss R. M. Selection and optimization of temporal spike encoding methods for spiking neural networks. IEEE transactions on neural networks and learning systems, 31(2):358–370, 2019.

- [38] Furber S. To build a brain. IEEE spectrum, 49(8):44–49, 2012.
- [39] Auge D., Hille J., Mueller E., and Knoll A. A survey of encoding techniques for signal processing in spiking neural networks. Neural Processing Letters, 53(6):4693–4710, 2021.
- [40] ANSI/AAMI. Testing and reporting performance results of cardiac rhythm and st segment measurement algorithms, american national standards institute, inc. (ansi). Association for the Advancement of Medical Instrumentation (AAMI), ANSI/AAMI/ISO EC57, 1998-(R)2008.
- [41] G. B. Moody and R. G. Mark. The impact of the mit-bih arrhythmia database. IEEE Engineering in Medicine and Biology Magazine, 20 (3):45–50, 2001.
- [42] P. de Chazal, M. O’Dwyer, and R. B. Reilly. Automatic classification of heartbeats using ecg morphology and heartbeat interval features. IEEE Transactions on Biomedical Engineering, 51 (7):1196–1206, 2004.
- [43] G. de Lannoy, D. François, J. Delbeke, and M. Verleysen. Weighted conditional random fields for supervised interpatient heartbeat classification. IEEE Transactions on Biomedical Engineering, 59 (1):241–247, 2012.
- [44] V. Mondéjar-Guerra, J. Novo, M.G. Penedo J. Rouco, and M. Ortega. Heartbeat classification fusing temporal and morphological information of ecgs via ensemble of classifiers. Biomed. Signal Process, Control 47:41–48, 2019.
- [45] Xing Y., Zhang L., Hou Z., Li X., Shi Y. and Yuan Y., and Yan L. Accurate ecg classification based on spiking neural network and attentional mechanism for real-time implementation on personal portable devices. Electronics, 11(12):1889, 2022.
- [46] Wu Y., Deng L., Li G., Zhu J., and Shi L. Spatio-temporal backpropagation for training high-performance spiking neural networks. Frontiers in neuroscience, 12:1, 2018.
- [47] Neftci E. O., Mostafa H., and Zenke F. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. IEEE Signal Processing Magazine, 36(6):51–63, 2019.
- [48] Zenke F. and Vogels T. P. The remarkable robustness of surrogate gradient learning for instilling complex function in spiking neural networks. Neural computation, 33(4):899–925, 2021.
- [49] Yin B., Corradi F., and Boht ´e S. M. Accurate and efficient time-domain classification with adaptive spiking recurrent neural networks. Nature Machine Intelligence, 3(10):905–913, 2021.

- [50] Virgilio G. C. D., Sossa A. J. H., Antelis J. M., and L. E. Falcón. Spiking neural networks applied to the classification of motor tasks in eeg signals. Neural networks, 122:130–143, 09 2019.
- [51] Frackowiak R., Friston K., Frith Ch., Dolan R., Price C., Zeki S., Ashburner J., and Penny W. Human brain function. Academic Press, second edition edition, 2003.