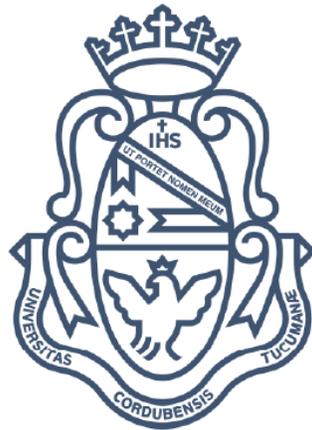


Universidad Nacional de Córdoba
Facultad de Ciencias Exactas, Físicas y Naturales
Facultad de Ciencias Médicas
Ingeniería Biomédica

Proyecto Integrador

“Procesamiento de señales de Electrocardiografía en Equipos Portables mediante Estrategias de Inteligencia Artificial”



Alumno:

Álvarez, Marcos Matías

Matrícula:

36.351.579

Director:

Riva, Guillermo Gastón

Co-director:

Díaz Dávila, Laura Cecilia



Córdoba Octubre de 2016

Agradecimientos

A Dios, en primer lugar.

A mi familia, por brindarme su apoyo durante la carrera.

A mis directores de tesis, por su tiempo y ayuda durante el desarrollo de este trabajo.

Abstract

This report presents the development of a system that enables portable electrocardiograms recording devices (Holter) move from a totally acquisition and storage/transmission scheme, to a more efficient and intelligent acquisition scheme and local processing, to determine whether to perform the transmission or storage of data.

The processing algorithm designed, performs filtering and segmentation of the signal based on simple transformations and peak detection. Then the beats are decomposed by Independent Component Analysis, resulting in a set of 5 features. Finally, the beats are classified through an artificial intelligence strategy called One Class Support Vector Machine.

The classification obtained is binary, ie the beats are classified as normal (healthy or pathological depending on the patient) and abnormal. This does not allow specifying the arrhythmia, but provides the important advantage of not requiring large databases.

The proposed algorithm was implemented in Matlab and tested with approximately 27000 beats (20500 normal and 6500 abnormal) extracted from the MIT-BIH Arrhythmia Database, obtaining 64% on average saving storage / data transmission, with 85% and 48% as upper and lower limits respectively, and 99% accuracy on classifying abnormal beats.

A series of tests were performed on the system, in order to evaluate the influence of noise on it. The system is then characterized, demonstrating a low incidence of noise on beat detection, and a slightly greater impact on beat classification.

Finally, a proof of concept was done by implementing the system in Simulink® which allowed to prove that the proposed algorithm could be used in real-time applications.

Resumen

En el presente informe de proyecto integrador, se presenta el desarrollo de un sistema que permite a dispositivos portátiles de registro de electrocardiogramas (holter) pasar de un esquema totalmente de adquisición y almacenamiento/transmisión, a un esquema más eficiente e inteligente de adquisición y procesamiento local, para determinar si es necesario realizar la transmisión o almacenamiento de los datos.

El algoritmo de procesamiento diseñado, realiza el filtrado y la segmentación de la señal basado en transformaciones sencillas y la detección de picos. Luego, los latidos son descompuestos mediante el Análisis de Componentes Independientes, resultando en un conjunto de 5 características. Finalmente, los latidos son clasificados mediante una estrategia de inteligencia artificial denominada Máquina de Vectores de Soportes de una clase.

La clasificación obtenida es binaria, es decir que los latidos se clasifican en normales (sanos o patológicos dependiendo del paciente) y anormales. Esto no permite especificar las patologías, pero brinda la importante ventaja de no requerir grandes bases de datos.

El algoritmo propuesto fue implementado en Matlab® y probado con aproximadamente 27000 latidos (20500 normales y 6500 anormales) extraídos de la base de datos de arritmias del MIT-BIH, obteniendo un ahorro medio de almacenamiento/transmisión de datos del 64%, con límites superior e inferior del 85% y 48%, respectivamente, y una precisión frente a los latidos anormales de un 99%.

Una serie de pruebas fueron ejecutadas sobre el sistema, a los fines de evaluar la influencia del ruido sobre éste. Se caracterizó entonces el sistema, demostrándose una baja incidencia del ruido sobre la detección de los latidos, y una incidencia un poco mayor sobre la clasificación de los latidos analizados.

Finalmente, se realizó una prueba de concepto implementando el sistema en Simulink® lo que permitió probar que el algoritmo propuesto podría ser utilizado en tiempo real.

Contenido

Agradecimientos	I
Abstract	II
Resumen	III
Contenido	1
Índice de Ilustraciones	3
Índice de Tablas	5
Introducción	6
Objetivos	7
Objetivos Generales	7
Objetivos Específicos	7
Capítulo 1: Marco Teórico	8
El Corazón	8
Actividad eléctrica del corazón	11
El electrocardiograma.....	14
Espectro de energía de una señal de ECG	20
Principales arritmias cardíacas	21
Procesamiento de la señal de ECG	27
Detección de latidos	27
Filtros enteros	28
Extracción de Características	31
Análisis de Componentes Independientes	31
Definición de Análisis de Componentes Independientes (ICA) [5]	32
Principios de la estimación ICA	34
Medida de la No-Gaussianidad	35
Pre-procesamiento para ICA	37
Algoritmo de punto fijo usando negentropía – FastICA	39
Análisis de Componentes Principales.....	41
Cálculo de los componentes principales	41
Propiedades de los componentes principales	43
Selección del número de componentes principales	45
Clasificación	46
SVM: Máquinas de Vectores de Soporte.....	46
SVM aplicado a problemas de clasificación de una clase.....	49
Capítulo 2: Implementación	53

Detección de complejos QRS	53
<i>Pre-procesamiento de la señal.....</i>	53
<i>Algoritmo de Detección</i>	60
Segmentación	63
Extracción de características	63
<i>Implementación del Análisis de Componentes Independientes.....</i>	65
Clasificación de los latidos.....	68
<i>Implementación de la Máquina de Vectores de Soporte.....</i>	70
Capítulo 3: Resultados	73
Evaluación del desempeño del algoritmo.....	73
<i>Detección de complejos QRS.....</i>	73
<i>Clasificación de los latidos</i>	75
Influencia del ruido en la señal	76
<i>Influencia del ruido en la detección</i>	76
<i>Influencia del ruido en la clasificación</i>	77
Prueba de concepto	77
Conclusiones	82
Propuestas para Investigaciones Futuras	83
Bibliografía y Referencias	84
Anexos	86
Anexo A – Código fuente del algoritmo propuesto	87
Anexo B – Parámetros de svm_train	95
Anexo C – Código fuente empleado en la prueba de concepto.....	97

Índice de Ilustraciones

Ilustración 1 – Estructuras anatómicas del corazón.	9
Ilustración 2 – Sistema de conducción del corazón.	11
Ilustración 3 – Circulación mayor y menor.....	12
Ilustración 4 – Derivaciones estándar de las extremidades.....	15
Ilustración 5 – Derivaciones aumentadas de las extremidades.....	16
Ilustración 6 – Derivaciones precordiales.	17
Ilustración 7 – Ondas, segmentos e intervalos característicos de un ECG.....	18
Ilustración 8 - Espectro de energía de una señal de ECG.....	20
Ilustración 9 – Ritmo sinusal normal.....	21
Ilustración 10 – Contracción auricular prematura.	24
Ilustración 11 – Contracción ventricular prematura.....	25
Ilustración 12 – Bloqueo de rama izquierda.	25
Ilustración 13 – Bloque de rama derecha.	26
Ilustración 14 – Latido iniciado por marcapasos externo implantable.....	26
Ilustración 15 - Esquema general de las etapas de procesamiento.....	27
Ilustración 16 - Filtro pasa-bajo con $m = 10$. Respuesta en magnitud y en fase.....	30
Ilustración 17 - Diagrama de bloque de un filtro pasa-alto basado en substracción de filtro.....	31
Ilustración 18 - – Representación gráfica de la relación entre componentes principales y estandarización multi-variante.	45
Ilustración 19 – (a) Hiperplano que separa las dos clases; (b) Hiperplanos de mayor margen negativo y positivo.	47
Ilustración 20 – Hiperplanos y margen.	48
Ilustración 21 – SVM de una clase.....	49
Ilustración 22 – Diferentes soluciones controladas por el parámetro v	51
Ilustración 23 - Etapas de filtrado del detector de QRS.....	53
Ilustración 24 - Electrocardiograma muestreado a 360 muestras por segundo.....	54
Ilustración 25 - Filtro Pasa-Bajo. a) Respuesta en magnitud. b) Respuesta en fase.	55
Ilustración 26 - Señal de ECG filtrada con filtro pasa-bajo.....	56
Ilustración 27 - Filtro Pasa-Alto. a) Respuesta en magnitud. b) Respuesta en fase.....	57
Ilustración 28 - Filtro Pasa-Banda. A) Respuesta en magnitud. b) Respuesta en fase.....	57
Ilustración 29 - Señal de ECG filtrada con filtro pasa-banda.....	58
Ilustración 30 - Señal de ECG luego de la etapa de diferenciación.....	58
Ilustración 31 – Derivada de la señal de ECG luego de elevarla al cuadrado.....	59
Ilustración 32 - Señal de ECG luego de la etapa de integración.....	60
Ilustración 33 - Algoritmo de detección de complejos QRS.....	61
Ilustración 34 - Detección de pico de señal.	62
Ilustración 35 - Precisión en función del número de muestras para obtención de componentes independientes.	65
Ilustración 36 – Valores propios de las señales.....	66

Ilustración 37 – Precisión vs Cantidad de componentes principales.	66
Ilustración 38 – Aplicación de la pseudo-inversa para el cálculo de los pesos.	68
Ilustración 39 – Latidos normales para diferentes registros.....	69
Ilustración 40 – Porcentaje de aciertos en función del tipo de núcleo para SVM.	70
Ilustración 41 – Precisión vs cantidad de muestras consideradas para el entrenamiento del modelo.	71
Ilustración 42 – Precisión en función de gamma y nu.	71
Ilustración 43 – (a) Aciertos de latidos anormales vs nu; (b) Precisión vs nu.	72
Ilustración 44 - Registro para entrenamiento.....	78
Ilustración 45 - Latidos detectados y segmentados.....	78
Ilustración 46 - Modelo de entrenamiento.....	79
Ilustración 47 - Modelo de clasificación.....	80
Ilustración 48 - Resultados de la primera simulación.	80
Ilustración 49 - Latidos generados por marcapasos externo implantable.....	81
Ilustración 50 - Resultados de la segunda simulación.....	81

Índice de Tablas

Tabla 1- Velocidad de conducción del tejido cardíaco.....	13
Tabla 2 - Principales eventos del ECG y su significado.....	20
Tabla 3- Únicas localizaciones donde $2\cos(\theta)$ resulta en coeficientes enteros.	29
Tabla 4 - Desempeño del detector en la base de datos del MIT-BIH.....	74
Tabla 5 - Desempeño del algoritmo.....	75
Tabla 6 - Porcentaje de aciertos discriminado en patologías.	75
Tabla 7 - Ahorro de memoria/transmisión para algunos registros analizados.....	76
Tabla 8 - Influencia del ruido en la detección.	76
Tabla 9 - Influencia del ruido en la clasificación.....	77

Introducción

El corazón es uno de los órganos más importantes del cuerpo humano. Su función es la de bombear la sangre hacia las distintas regiones del organismo, aportando de este modo los nutrientes necesarios para la función celular y conduciendo los desechos metabólicos hacia los órganos en los cuales se produce su eliminación. Esto es logrado mediante una secuencia ordenada de contracciones musculares que generan un gradiente de presiones en el circuito cerrado que constituye el sistema circulatorio.

La actividad central del corazón en la circulación sanguínea, hace que cualquier afección en el funcionamiento del mismo repercute sobre los demás órganos del cuerpo. Es, por lo tanto, de gran importancia médica la detección de anomalías cardíacas.

El electrocardiograma es una de las fuentes de información diagnóstica de mayor valor en el estudio del funcionamiento del corazón. Su no invasividad, facilidad de obtención y escaso costo, sumado a la gran variedad de patologías cardíacas que se pueden detectar mediante su análisis, lo han convertido en un estudio de rutina en las consultas cardiológicas.

Muchas de las patologías cardíacas pueden ser detectadas mediante un electrocardiograma de corta duración, que no supera los dos o tres minutos. Existen, no obstante, un importante número de enfermedades de aparición esporádica (o causada por un evento no reproducible en el momento del estudio) que requiere la realización de monitoreo móvil, de los que se obtienen registros electrocardiográficos de varias horas de duración. Los dispositivos utilizados para este monitoreo se limitan únicamente a la adquisición y almacenamiento de datos electrocardiográficos, sin analizar si los mismos son relevantes o no. Dado que los equipos de monitoreo móvil tienen memorias limitadas para almacenar datos, se tiene un almacenamiento ineficiente.

Si bien numerosas técnicas y sistemas de diagnóstico asistido han sido desarrolladas intentando dar solución al problema de analizar estudios de larga duración luego de realizado el mismo (procesamiento off-line), existen muy pocas propuestas para dotar a los dispositivos de monitoreo móvil con la capacidad de análisis de los datos adquiridos (procesamiento on-line). La incorporación de algún esquema simple de procesamiento en el equipo de adquisición, como puede ser filtrar datos no relevantes, detectar patrones anormales, etc., posibilitará tanto reducir la cantidad de información a almacenar en la limitada memoria del equipo portátil, como reducir la comunicación inalámbrica de los datos en sistemas de adquisición en tiempo real, lo cual influye en el consumo de energía del equipo.

Sobre este tipo de procesamiento in situ en el equipo médico de adquisición no hay referencias conocidas, y se sabe que es un área en la que se está trabajando a nivel de investigación en diferentes instituciones.

Objetivos

Objetivos Generales

El objetivo principal de este proyecto integrador es dotar a los dispositivos electrónicos de estrategias de inteligencia artificial que permitan pasar de un esquema totalmente de adquisición y transmisión o almacenamiento, a un esquema más eficiente e inteligente de adquisición, y procesamiento local para determinar si es necesario realizar la transmisión, o almacenamiento, de los mismos. No se pretende realizar un diagnóstico preciso a partir de los datos recibidos, sino detectar un comportamiento anormal para que sea tenido en cuenta por el profesional médico. Como objetivo opcional, se pretende realizar una prueba de concepto del sistema a desarrollar.

Objetivos Específicos

Son objetivos específicos del presente proyecto integrador:

- Elaborar un sistema de detección de latidos en tiempo real.
- Realizar una caracterización, mediante una técnica de bajo costo computacional, de los latidos detectados en el apartado anterior.
- Seleccionar una estrategia de inteligencia artificial que pueda ser implementada en procesadores de bajos recursos.
- Entrenar el sistema para realizar la clasificación de los latidos, analizando la influencia de distintos parámetros de entrenamiento.
- Ensamblar los distintos subsistemas antes mencionados (detección, caracterización y clasificación).
- Caracterizar el algoritmo de clasificación, teniendo en cuenta la influencia de distintos parámetros propios del sistema y externos al mismo.
- Realizar una prueba de concepto del sistema desarrollado.

Capítulo 1: Marco Teórico

En el presente capítulo se pretende introducir al lector en los fundamentos teóricos necesarios para la comprensión del desarrollo del presente informe. Considerando que la temática central de este proyecto integrador es el procesamiento de señales electrocardiográficas que permita la clasificación de los latidos que se encuentran en ellas, podemos agrupar los temas a tratar en dos grupos:

- Principios anatómicos y fisiológicos relacionados al funcionamiento del corazón, que permitirán entender los temas relacionados al origen de la señal a trabajar, las características principales de la misma y los distintos patrones normales y patológicos a tratar.
- Principios matemáticos asociados al procesamiento de la señal electrocardiográfica. Para una mayor comprensión, estos principios se agruparán en tres secciones, las cuales representan las tres etapas de procesamiento a trabajar; estas son: detección de latidos, extracción de características y clasificación.

El Corazón

El corazón es el órgano principal del sistema circulatorio. Como tal, su función es propulsar la sangre a través de los vasos sanguíneos hacia las distintas regiones del organismo, permitiendo de este modo la llegada de nutrientes y oxígeno a cada célula y la eliminación de los residuos resultantes de la actividad metabólica.

En un individuo sano, el corazón pesa entre 200 y 250 gramos y su volumen es apenas mayor que el puño de la persona. La forma de este órgano es similar a un cono o a una pirámide, en el que se reconoce una base, dirigida hacia atrás, arriba y algo a la derecha, y un vértice (ápex) situado hacia adelante y a la izquierda; de modo tal que el eje geométrico general del corazón se acerca más a la horizontal que a la vertical. De esta manera, el corazón se ubica en la porción inferior del mediastino medio, desarrollándose en su mayor parte hacia la izquierda de la línea media torácica; y es fijado y mantenido en su lugar por los grandes vasos que llegan y salen de él (venas Cava Inferior y Superior, y Pulmonares Derecha e Izquierda, y arterias Aorta y Pulmonar).

Externamente, el corazón está tapizado por el epicardio y se encuentra rodeado por un saco fibroso a través del cual se encuentra unido a las diferentes estructuras de la pared torácica y del mediastino: el pericardio. Este último, está formado por dos capas: una capa visceral (más interna) y una capa parietal (más externa), entre las cuales se forma una cavidad virtual hacia la que es secretado un líquido lubricante que evita la fricción mecánica debida a los movimientos del corazón.

Desde el punto de vista anatómico, el mismo es un órgano muscular hueco dividido en dos mitades (izquierda y derecha) por un tabique músculo-membranoso. Cada una de estas mitades constituye a su vez dos cavidades: una superior, la aurícula, y una inferior, el ventrículo. Cada

aurícula comunica con el ventrículo correspondiente mediante un orificio (orificio aurículo-ventricular) provisto de válvulas denominadas válvula mitral o bicúspide, para el corazón izquierdo, y válvula tricúspide para el derecho. Por otra parte los ventrículos poseen un segundo sistema de válvulas (válvulas semilunares) en el origen de las arterias que de ellos salen (válvula pulmonar y aórtica), que fuerzan a la sangre a circular en un único sentido. Todas las cavidades antes mencionadas, se encuentran tapizadas interiormente por el endocardio, superficie endotelial que se une sin solución de continuidad con la íntima de los grandes vasos. En la ilustración 1 se esquematizan las principales estructuras anatómicas del corazón.

Histológicamente, el miocardio está compuesto por un tipo muscular especializado: el músculo cardíaco. El mismo presenta estriaciones formadas por el ordenamiento de miofibrillas de actina y miosina, al igual que el músculo esquelético, pero a diferencia de éste, las células miocárdicas poseen sólo uno o dos núcleos y se comunican entre sí mediante un tipo de unión característica: los discos intercalares. Este tipo de unión brinda cohesión entre las células, a la vez que representa un sitio de baja resistencia para la difusión de iones y por lo tanto del impulso excitatorio, permitiendo que el músculo cardíaco se comporte como una célula multinucleada sin que verdaderamente lo sea. Esta última característica permite la contracción simultánea de un gran conjunto de células musculares, favoreciendo la producción del latido cardíaco.

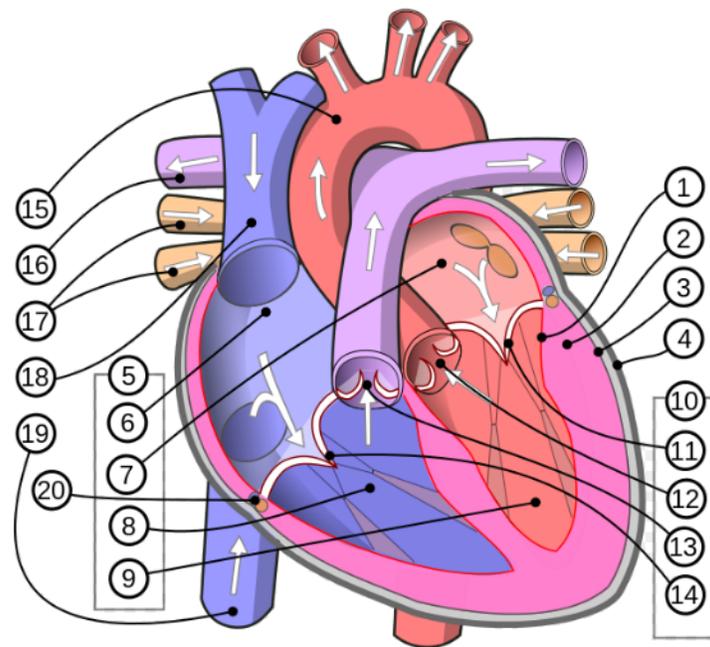


Ilustración 1 – Estructuras anatómicas del corazón. ¹

¹ Referencias: 1) Endocardio. 2) Miocardio. 3) Epicardio. 4) Pericardio. 5) Cámaras del corazón. 6) Atrio derecho. 7) Atrio izquierdo. 8) Ventrículo derecho. 9) Ventrículo izquierdo. 10) Válvulas cardíacas. 11) Válvula Aórtica. 12) Válvula Pulmonar. 13) Válvula Tricúspide. 14) Válvula Mitral. 15) Arteria Aorta. 16) Arteria Pulmonar. 17) Venas Pulmonares. 18) Vena Cava Superior. 19) Vena Cava Inferior. 20) Vasos Coronarios.

Las propiedades características del músculo cardíaco son las siguientes:

- Función cronotrópica o automatismo: capacidad del miocardio de generar su propio impulso excitatorio. El corazón continúa latiendo aún después de la escisión de toda conexión nerviosa; es más, si el corazón es cortado en trozos, los mismos continúan latiendo hasta que la falta de nutrientes causen la muerte celular.
- Función dromotrópica o conductividad: capacidad de conducción de un estímulo eléctrico, permitiendo que el mismo se difunda rápidamente al resto del corazón.
- Función batmotrópica o excitabilidad: capacidad de responder a un estímulo generando un potencial de acción.
- Función inotrópica o contractilidad: capacidad del músculo cardíaco de desarrollar fuerza y acortarse.

Las fibras musculares que constituyen el corazón pueden ser clasificadas en dos grupos: fibras musculares de contracción y fibras musculares especializadas de excitación y conducción. Dentro del primer grupo se encuentra el músculo cardíaco auricular y ventricular, siendo el último más voluminoso, sobretodo alrededor del ventrículo izquierdo. Cada uno de estos grupos de músculos contráctiles funciona como un sincitio independiente y poseen una gran capacidad de contracción.

A diferencia de las anteriores, las fibras musculares especializadas de excitación y contracción se contraen débilmente, pero presentan la singularidad de generar descargas eléctricas rítmicas automáticas en forma de potenciales de acción, así como de conducir dichos impulsos a través del corazón. Estas fibras se ubican subendocárdicamente y constituyen lo que se conoce como el sistema cardionector o sistema de conducción del corazón. Este sistema está constituido por el nodo sinusal (ubicado en la aurícula derecha, en las proximidades del orificio de la vena cava superior) y el conjunto aurículo-ventricular, formado por:

- Nodo aurículo-ventricular ubicado en la pared del ventrículo derecho, en el tabique aurículo-ventricular.
- Haz de His o fascículo aurículo-ventricular que parte del nodo anterior y se divide en una rama derecha y una izquierda (que a su vez se ramifica en un fascículo anterior y uno posterior).
- Fibras de Purkinje que parten de los fascículos antes mencionados y se dispersan por todo el miocardio ventricular.

En la ilustración 2 puede observarse un esquema del sistema cardionector.

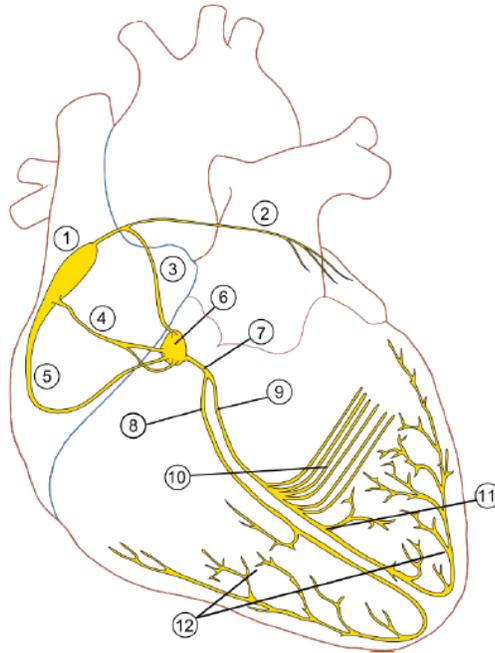


Ilustración 2 – Sistema de conducción del corazón.²

Es importante destacar que si bien se han descrito vías de conducción internodales e interauriculares (que comunican el nodo sinusal y el aurículo-ventricular y las aurículas entre sí respectivamente), en la actualidad se considera que las mismas no están formadas por músculo cardíaco especializado sino que es el mismo músculo contráctil el que transmite el potencial de acción en estas zonas.

Actividad eléctrica del corazón

Desde el punto de vista físico, el corazón constituye un sistema de dos bombas conectadas en serie: el corazón derecho se encarga de bombear la sangre con bajo contenido de oxígeno, proveniente de la circulación mayor o sistémica, y enviarla hacia los pulmones; mientras que, el corazón izquierdo, toma la sangre oxigenada proveniente de la circulación menor (de los pulmones) y la retoma nuevamente a la circulación sistémica.

El bombeo de la sangre por parte del corazón es producido gracias a una secuencia ordenada de fenómenos mecánicos y eléctricos que suceden desde el comienzo de un latido hasta el comienzo del siguiente, y son conocidos como ciclo cardíaco. El mismo está compuesto por un período de contracción o sístole, seguido de un período de relajación o diástole.

² Referencias: 1) Nodo sinusal. 2) 3) 4) 5) Vías internodales e interauriculares. 6) Nodo aurículo-ventricular. 7) Haz de His. 8) Rama derecha. 9) Rama izquierda. 10) Fascículo anterior izquierdo. 11) Fascículo posterior izquierdo. 12) Fibras de Purkinje.

Imagen obtenida de https://es.wikipedia.org/wiki/N%C3%B3dulo_sinoauricular.

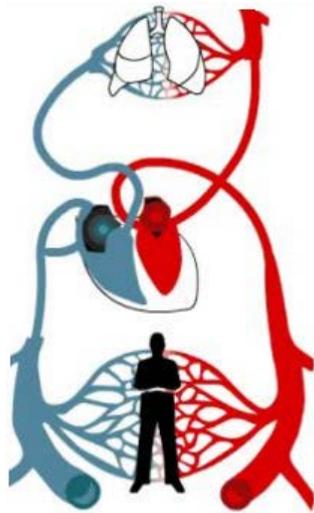


Ilustración 3 – Circulación mayor y menor.³

Dado que estos períodos de contracción y relajación no se producen en simultáneo en todo el corazón, se pueden distinguir tres tiempos distintos durante el ciclo cardíaco: la sístole auricular, la sístole ventricular, y la diástole:

- Durante el período de diástole, las cuatro cámaras del corazón se relajan, y se produce el llenado de manera pasiva de un 80% del volumen de los ventrículos (esto se debe a que normalmente la sangre fluye de manera constante desde las venas hacia el corazón).
- Durante la sístole auricular, las aurículas se contraen produciendo el llenado del 20% restante del volumen ventricular.[1]
- Durante la sístole ventricular ocurre el vaciado de los ventrículos. En este, se distinguen dos etapas: en primer lugar se produce una contracción ventricular isovolumétrica que eleva con rapidez la presión intraventricular, hasta superar la presión existente en las arterias (80 mmHg), momento en el cual se abren las válvulas semilunares y comienza la fase de expulsión ventricular.

El ciclo cardíaco se repite unas 50 a 100 veces por minuto en un individuo sano (frecuencia cardíaca normal), lo que establece una duración de entre 0,6 y 1,2 segundos por latido.

Si bien todo el músculo cardíaco tiene la capacidad de automatismo, y por lo tanto cada fibra muscular del corazón puede contraerse de manera independiente, la coordinación de la contracción miocárdica es fundamental para una función cardíaca normal. Dado que, como se mencionó anteriormente, el corazón aislado de todo estímulo nervioso externo continúa latiendo de manera organizada, es evidente que esta coordinación es una capacidad intrínseca del corazón: la automatización y organización del ciclo cardíaco se debe a las células especializadas del sistema cardionector. Estas células (también conocidas como células marcapasos) tienen la capacidad de generar potenciales marcapasos, es decir potenciales de acción rítmicos espontáneos, que son las

³ Imagen obtenida de <https://sites.google.com/site/msistemacirculatorio/d--circulacion-mayor-y-menor>.

responsables de la despolarización del músculo cardíaco, y por consiguiente de la contracción del mismo. Los fenómenos eléctricos que determinan el ciclo cardíaco puede describirse de la siguiente manera:

1. Las células marcapaso del nodo sinusal generan un potencial de acción marcapasos que inicia la despolarización del miocardio auricular.
2. La despolarización originada en el nodo sinusal se propaga por las aurículas, para converger en el nodo aurículo-ventricular. Dicho nodo constituye la única vía de transmisión del potencial de acción entre el sincitio auricular y el ventricular (la existencia de dos sincitios independientes y la presencia del nodo aurículo-ventricular es el motivo por el cual no se produce la contracción completa del corazón al realizarse la descarga del nodo sinusal).
3. La conducción del potencial marcapasos por parte del nodo aurículo-ventricular es lenta debido a la mayor resistencia al paso de los iones que originan la despolarización entre una fibra y otra. Esto causa un retraso de aproximadamente 0,1 segundo, lo que permite que las aurículas se contraigan por completo, vertiendo la sangre en los ventrículos, antes de que se inicie la contracción ventricular.
4. Una vez pasado el nodo aurículo-ventricular, el impulso excitatorio es conducido rápidamente por el sistema de His-Purkinje. Una característica particular de este sistema de conducción es la imposibilidad de conducir estímulos de manera retrógrada (los potenciales de acción viajan a través de él en un único sentido), de modo tal que en condiciones fisiológicas normales, es imposible que se produzca la reentrada del potencial marcapasos a las aurículas.
5. Cuando el impulso llega a los extremos de las fibras de Purkinje, se propaga a través de las propias fibras musculares miocárdicas. Nuevamente la reentrada de este impulso a las aurículas se encuentra impedida por la aislación que constituye el tejido fibroso que rodea las válvulas cardíacas. De esta manera, la despolarización de los ventrículos se produce en una secuencia que recuerda el recorrido del sistema cardionector dentro de los mismos, propagándose desde la porción superior izquierda del tabique interventricular, hacia la derecha y abajo hasta llegar al ápex, regresando a través de las paredes ventriculares hasta el surco aurículo-ventricular, desde adentro hacia afuera (endocardio a epicardio).

Tejido	Velocidad de conducción (m/s)
Nodo sinusal	0,05
Músculo auricular	0,3 a 0,5
Vías internodales	1,0
Nodo aurículo-ventricular	0,05
Sistema de His-Purkinje	1,5 a 4,0
Músculo ventricular	0,3 a 0,5

Tabla 1- Velocidad de conducción del tejido cardíaco.⁴

⁴ Datos obtenidos de A. C. Guyton; J. E. Hall. Tratado de fisiología médica. 12^a Edición. Madrid.

Tal como se describió en los puntos anteriores, las células marcapaso del nodo sinusal son las que, en condiciones normales, inician el ciclo cardíaco y determinan la frecuencia del mismo. No obstante, en un corazón sano se puede establecer una jerarquía de marcapasos, que está determinada por la frecuencia de descarga propia del conjunto de fibras del que se trate, a saber:

- Nodo sinusal: 60 a 100 veces por minuto.
- Nodo aurículo-ventricular: 50 a 60 veces por minuto.
- Sistema de His-Purkinje: 30 a 40 veces por minuto.

Por lo tanto, si el nodo de mayor jerarquía funciona correctamente, los demás no generan potenciales de acción, sino que se limitan a la conducción de los mismos; mientras que, si se produce un fallo en el funcionamiento de uno de los nodos, el siguiente puede tomar el mando, generando potenciales marcapaso a una frecuencia propia.

Llegado este punto, es importante mencionar que si bien el corazón es un órgano autónomo, existe un control nervioso y endocrino sobre el ritmo cardíaco que actúa sobre la frecuencia de disparo espontánea de los distintos nodos marcapaso.

El electrocardiograma

Dado que el cuerpo humano es un conductor de volumen (debido a la buena conducción eléctrica por parte de los líquidos corporales), las fluctuaciones de potencial que se generan por la despolarización del miocardio se propagan también a los tejidos adyacentes y en menor proporción, a la superficie corporal. Por lo tanto, si sobre la piel se colocan electrodos en posiciones adecuadas, es posible registrar dichas fluctuaciones. Al registro gráfico de las variaciones de potencial causadas por las despolarizaciones y repolarizaciones del miocardio en su totalidad se lo conoce como electrocardiograma (ECG).[2]

Como se sabe, solo es posible registrar una diferencia de potencial entre dos puntos distintos. En la electrocardiografía esto puede lograrse ya sea empleando dos electrodos activos (registrando la diferencia de polaridad entre ellos, en una técnica que se conoce como registro bipolar), o mediante un electrodo activo y uno indiferente que se conecta a potencial cero (técnica conocida como registro unipolar). La disposición de las conexiones de cada par de electrodos se conoce como derivación. En un estudio electrocardiográfico normal, es posible distinguir doce derivaciones, que pueden ser clasificadas en tres grupos: derivaciones estándar de las extremidades, derivaciones aumentadas de las extremidades y derivaciones precordiales.

En las derivaciones estándar de las extremidades, también conocidas como derivaciones de Einthoven (por su precursor Willem Einthoven), se registran las diferencias de potenciales entre dos extremidades. Los electrodos son conectados en el brazo izquierdo, el brazo derecho y la pierna izquierda (triángulo de Einthoven) y se obtienen tres derivaciones distintas:

- DI: diferencia de potencial entre el brazo izquierdo y el brazo derecho, considerando al primero como electrodo positivo (activo).

- DII: diferencia de potencial entre el brazo derecho y la pierna izquierda, siendo este último el electrodo positivo.
- DIII: diferencia de potencial entre el brazo izquierdo y la pierna izquierda. En este caso también se considera como positivo al electrodo ubicado en la pierna.

En la ilustración 4 puede observarse un esquema de las derivaciones de Einthoven.

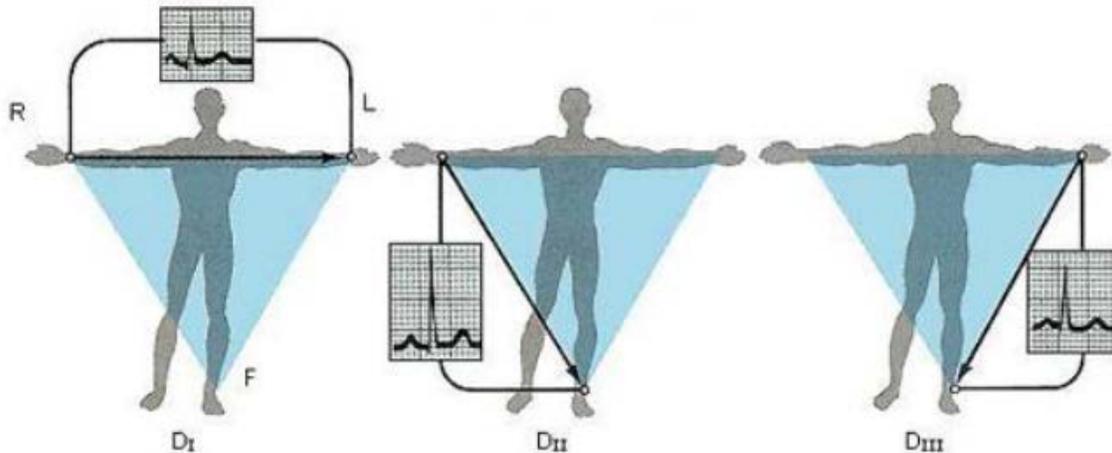


Ilustración 4 – Derivaciones estándar de las extremidades.⁵

Un hecho importante para destacar es que, dado que la corriente solo es conducida por los líquidos corporales, es casi indistinta la posición en la que se coloquen los electrodos en las extremidades, logrando los mismos resultados que si éstos estuvieran colocados en los hombros y cadera respectivamente. No obstante, dado que en muchos casos existe una fuerte interferencia causada por la actividad muscular, la respiración y otros artefactos, es común la colocación de los electrodos en dichos puntos (ya que se reduce considerablemente la distorsión de la señal). Dicha colocación es conocida como modificación de Mason-Likar (por el apellido de quienes la propusieron), y es muy utilizada en estudios ambulatorios o de larga duración.

Según lo establecido anteriormente, las derivaciones estándar de las extremidades constituyen registros bipolares. Dado que en un conductor de volumen la suma de los potenciales en los puntos de un triángulo equilátero con una fuente de corriente central es siempre de cero [1], es posible obtener un registro unipolar si en lugar de tomar la diferencia de potencial entre dos extremidades, se la considera entre una de ellas y un electrodo virtual cuyo potencial instantáneo es el promedio de los tres electrodos utilizados en las derivaciones de Einthoven (denominado electrodo central de Wilson en honor a Frank N. Wilson que propuso su utilización). En 1942, E. Goldberger demostró que era posible amplificar pasivamente un 50% las señales obtenidas utilizando el terminal central de Wilson, si para su cálculo se excluía la información provista por el electrodo activo.

⁵ Imagen obtenida de <http://www.electrocardiografia.es/derivaciones.html>.

Las derivaciones obtenidas de esta manera se llaman derivaciones amplificadas de las extremidades y son las siguientes:

- aVI (augmented VI): derivación unipolar aumentada del brazo izquierdo.
- aVr (augmented Vr): derivación unipolar aumentada del brazo derecho.
- aVf (augmented Vf): derivación unipolar aumentada de la pierna izquierda.

Es importante recalcar que tanto la información otorgada por las derivaciones unipolares de las extremidades (utilizando el terminal central de Wilson) y las derivaciones aumentadas de las extremidades, es bastante redundante respecto a las provistas por las derivaciones de Einthoven. En la ilustración 5 puede observarse un esquema de las derivaciones aumentadas de las extremidades.

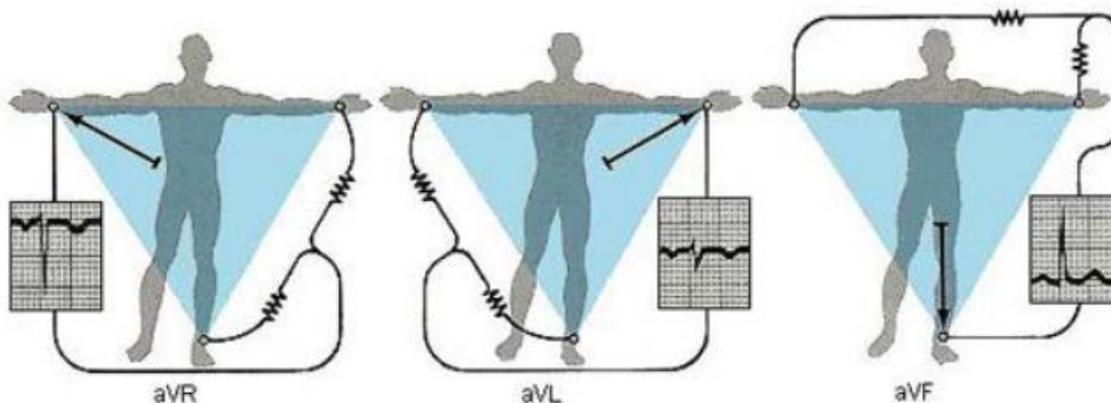


Ilustración 5 – Derivaciones aumentadas de las extremidades.⁶

Finalmente existen un conjunto de derivaciones unipolares, llamadas V1 a V6, que miden la diferencia de potencial entre el electrodo central de Wilson y distintos puntos del tórax. La ubicación del electrodo activo en cada una de estas derivaciones son las siguientes:

- V1: Cuarto espacio intercostal, dos centímetros a la derecha de la línea medio-esternal.
- V2: Cuarto espacio intercostal, dos centímetros a la izquierda de la línea medio-esternal.
- V3: Punto medio entre V2 y V4.
- V4: Quinto espacio intercostal izquierdo, en la línea media claviclar.
- V5: Quinto espacio intercostal izquierdo, en la línea axilar anterior.
- V6: Quinto espacio intercostal izquierdo, en la línea axilar media.

Estas derivaciones precordiales, permiten registrar la actividad eléctrica del corazón en el plano horizontal, a diferencia de todas las antes mencionadas que lo hacen en un plano coronal. En la ilustración 6 se observa un esquema de estas derivaciones.

⁶ Imagen obtenida de <http://www.electrocardiografia.es/derivaciones.html>.

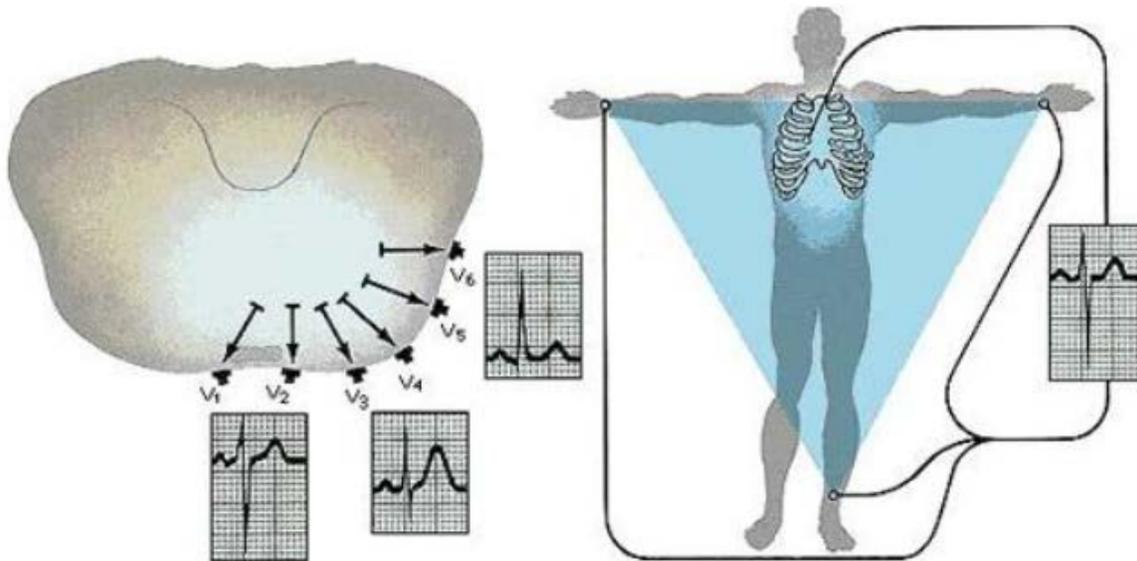


Ilustración 6 – Derivaciones precordiales.⁷

Independientemente de la derivación que se esté analizando, cuando una onda de despolarización o repolarización se dirige físicamente hacia el electrodo activo, el mismo se tornará positivo respecto del electrodo positivo. Por convención en dicho caso se registra una deflexión positiva (hacia arriba) del trazo electrocardiográfico; en el caso contrario ocurre cuando es el electrodo pasivo quien se torna positivo respecto del activo.

De este modo el registro electrocardiográfico se compone de un grupo de ondas (deflexiones positivas o negativas) que representan los fenómenos de despolarización y repolarización de las distintas estructuras del corazón a lo largo de un ciclo cardíaco. En aquellos momentos en los que no existen variaciones del potencial eléctrico (fase isoeletrica) el trazo obtenido en un ECG constituye la línea de base o línea isoeletrica. La fase isoeletrica comprendida entre el final de una onda y el principio de la siguiente se denomina segmento, mientras que un conjunto de ondas sin la existencia de fase isoeletrica entre ellas se denomina complejo. Por el contrario, un intervalo es un conjunto de ondas y segmentos.

En un electrocardiograma normal, los fenómenos eléctricos que causan la sístole y diástole auricular y ventricular, se manifiestan como una secuencia ordenada de ondas de características bien definidas, que por convención se designan como ondas "P", "Q", "R", "S" y "T". En la ilustración 7 puede observarse los nombres de las principales ondas, segmentos e intervalos del ECG normal.

⁷ Imagen obtenida de <http://www.electrocardiografia.es/derivaciones.html>.

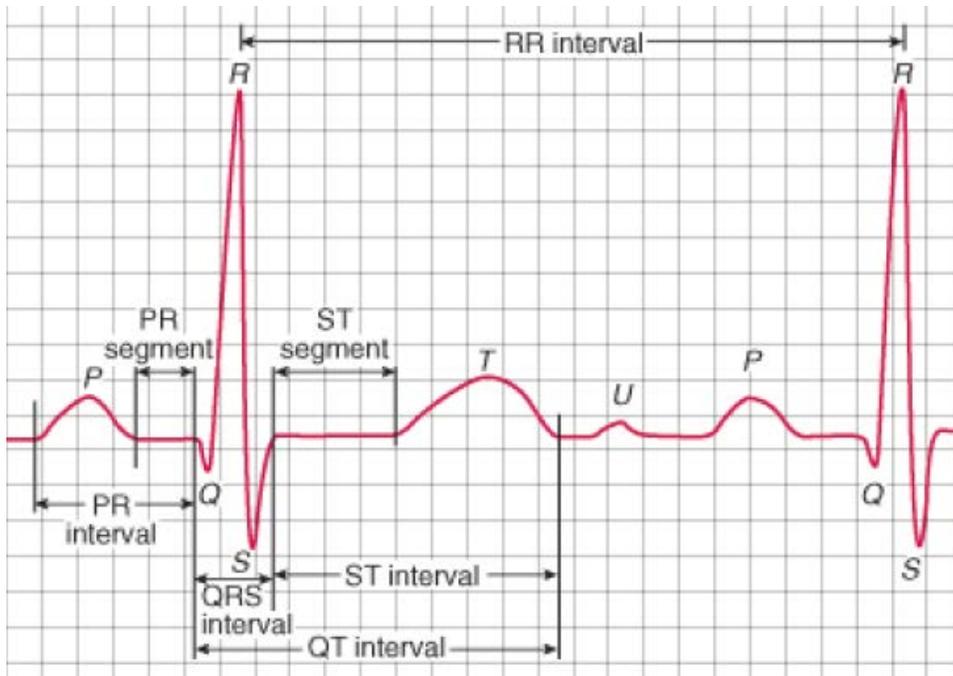


Ilustración 7 – Ondas, segmentos e intervalos característicos de un ECG.⁸

A continuación se describen los aspectos característicos más importantes de las ondas normales de un ECG.

Onda P

La onda P se produce por la despolarización de las aurículas. Como la misma ocurre de derecha a izquierda (dada la posición del nodo sinusal), el primer tercio de la onda P representa la despolarización de la aurícula derecha, el tercio final la despolarización de la aurícula izquierda, y el tercio medio una combinación de ambas.

Esta onda se presenta como una deflexión positiva en la mayoría de las derivaciones, salvo en aVI (en la que siempre es negativa). Su duración no supera los 120 mSeg, y su amplitud es menor a 0,25 mV en las derivaciones de los brazos, y menor a 0,15 mV en las precordiales.

La repolarización de las aurículas comienza de 150 a 200 mSeg después de la finalización de la onda P. Dado que este fenómeno se superpone con el complejo QRS, la onda correspondiente (onda T auricular) se halla oculta por el mismo.

Intervalo PQ o PR

El intervalo PQ está constituido por la onda P y el segmento PR o PQ, y representa la despolarización auricular y la conducción a través del nodo aurículo-ventricular. Su duración es de entre 120 a 200 mSeg.

⁸ Imagen obtenida de <http://www.merckmanuals.com>.

Complejo QRS

El complejo QRS es originado por la rápida despolarización de los ventrículos (lo cual implica la conducción del impulso eléctrico por el sistema His-Purkinje).

Independientemente de la derivación que se considere, se trata del conjunto de ondas de mayor amplitud del ECG, alcanzando hasta 1,5 mV en las derivaciones de las extremidades. Respecto a su duración, la misma no supera los 100 mSeg ni es menos a 80 mSeg.

Onda T

La onda T representa la repolarización ventricular. Se trata de una onda positiva en la mayoría de las derivaciones del ECG, salvo en aVr (en la que siempre es negativa) y V1. Su duración es de aproximadamente 150 mSeg y su amplitud no supera los 0,5 mV.

Como puede observarse, pese a que representan la despolarización y repolarización de la misma estructura cardíaca, la onda T posee menor amplitud que el complejo QRS. Esta diferencia se debe a que la despolarización es un fenómeno abrupto y de corta duración que ocurre casi simultáneamente en todas las fibras del miocardio ventricular, mientras que la repolarización es un proceso lento, que comienza entre 200 y 350 mSeg. después del inicio del complejo QRS.

Intervalo QT

El intervalo QT representa los fenómenos eléctricos ocurridos durante toda la sístole ventricular. Su duración está fuertemente influenciada por la frecuencia cardíaca: a mayor frecuencia cardíaca menor duración del intervalo y viceversa; siendo su duración promedio de 400 mSeg.

Onda U

La aparición de la onda U es un fenómeno poco común en individuos sanos, aun así es posible encontrarla sin que la misma represente una afección cardíaca. Aunque su origen fisiológico es incierto, se supone que se debe a fenómenos de repolarización lenta o potenciales resultantes de fuerzas mecánicas en la pared ventricular.

Intervalo RR

El intervalo RR representa el conjunto de fenómenos eléctricos que transcurren durante un ciclo cardíaco y se define como todas las ondas y segmentos comprendidos entre el transcurso de una onda R y la siguiente. Los motivos por los cuales se prefiere tomar al complejo QRS como marca de referencia para la determinación de la duración del ciclo cardíaco son:

- Facilidad de identificación: el complejo QRS constituye el conjunto de ondas de mayor amplitud para su identificación ya sea visualmente o mediante distintos métodos matemáticos.
- Imposibilidad de no existencia: dado que la mayor parte de la función cardíaca es llevado a cabo por los ventrículos, la ausencia de la contracción ventricular es incompatible con la vida, de modo tal que el complejo QRS siempre está presente en un ECG. No ocurre lo mismo con las demás ondas, que pueden no existir o hallarse ocultas por otros fenómenos eléctricos.

Un intervalo RR en el que se pueden identificar todas las ondas y segmentos y los mismos poseen las características antes mencionadas, es decir un intervalo RR que transcurre durante un ciclo cardíaco normal, es denominado intervalo NN (por normal-normal).

En la tabla 2 se presenta un resumen de los principales eventos del ECG.

Evento del ECG	Duración	Fenómenos Cardíacos
Onda P	0,12 seg.	Despolarización auricular.
Intervalo PQ	0,12 – 0,2 seg.	Despolarización auricular y conducción por el nodo aurículo-ventricular.
Complejo QRS	0,08 – 0,1 seg.	Despolarización ventricular. Oculta la repolarización auricular.
Intervalo QT	0,4 seg.*	Sístole ventricular completa.
Onda T	0,15 seg.	Repolarización ventricular.
Intervalo RR	0,6 a 1 seg.*	Ciclo cardíaco completo.

*Depende del ritmo cardíaco

Tabla 2 - Principales eventos del ECG y su significado.⁹

Espectro de energía de una señal de ECG

En la señal de ECG coexisten diversos componentes como los complejos QRS, las ondas P y T, ruido muscular, interferencia electromagnética, etc. Se muestran los resultados del análisis del espectro de energía de un ECG en la ilustración 8.[3]

En ella se puede apreciar que la energía de los complejos QRS se encuentra centrada alrededor de los 10Hz. Este resultado permite diseñar un filtro que aisle en frecuencia la componente del ECG correspondiente a los complejos QRS

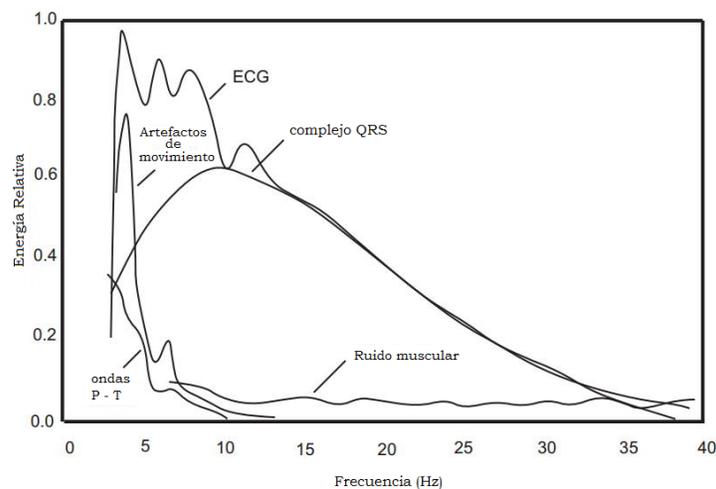


Ilustración 8 - Espectro de energía de una señal de ECG.

⁹ Datos obtenidos de W. F. Ganong. Fisiología Médica. 23a Edición. México, 2010.

Principales arritmias cardíacas¹⁰

De acuerdo a lo visto anteriormente, un corazón sano late entre 60 y 100 veces por minuto. Cada uno de estos latidos constituye un proceso ordenado de contracciones y relajaciones auriculares y ventriculares. Dichas contracciones son el resultado de un potencial marcapaso que se origina en el nodo sinusal y es conducido por el resto del sistema cardionector y el músculo cardíaco de manera organizada. Bajo estas condiciones se dice que el ritmo cardíaco es sinusal normal y todos los eventos registrados en el electrocardiograma poseen las características mencionadas en el apartado anterior. En la ilustración 9 puede observarse un segmento de ECG con ritmo sinusal normal.

Se denomina arritmia a cualquier alteración en la sucesión de los eventos que definen el latido cardíaco. Estas alteraciones pueden ser tanto en la frecuencia, como en la regularidad de los eventos. En el primero de los casos no se observa ninguna alteración morfológica en el ECG, salvo por una mayor o menor duración de los intervalos RR según se trate de bradicardia (disminución de la frecuencia cardíaca por debajo de 60 latidos por minuto) o taquicardia (aumento de la frecuencia cardíaca en reposo por encima de los 100 latidos por minuto). Por otra parte, en las arritmias irregulares, las ondas o los segmentos del ECG presentan patrones distintos a los normales. Según su punto de origen, las arritmias pueden ser clasificadas en atriales o auriculares, ventriculares y nodales (cuando se inician en el nodo aurículo-ventricular).

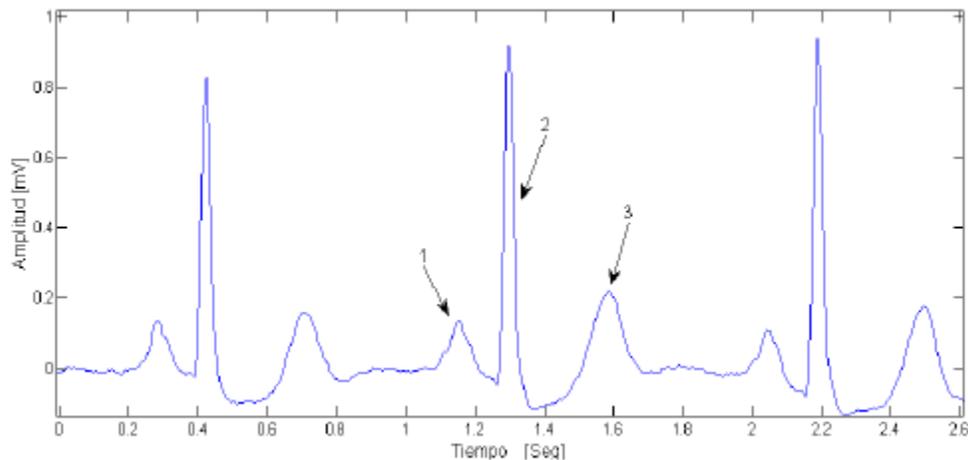


Ilustración 9 – Ritmo sinusal normal.¹¹

Las causas de las arritmias cardíacas más comunes pueden ser una o la combinación de alguna de las siguientes:

- Marcapasos ectópicos: En condiciones normales, las células miocárdicas no producen potenciales marcapaso y es el nodo sinusal quien lleva el ritmo del corazón. Sin embargo es posible que, bajo ciertas circunstancias (patológicas o no), se produzcan focos de

¹⁰ Extraído de “caracterización de Electrocardiograma mediante redes neuronales y transformada ondita” con la autorización de Jeremy A. Fischer.

¹¹ Referencias: 1) Onda P. 2) Complejo QRS. 3) Onda T.

excitación externos al nodo sinusal. Estos focos ectópicos pueden encontrarse dentro o fuera del sistema cardionector, ya sea en las aurículas o los ventrículos. Dependiendo de la periodicidad de descarga, la generación de potenciales marcapaso ectópicos puede ocasionar extrasístoles o contracciones prematuras (si se trata de una descarga aislada) o taquicardias paroxísticas (si la frecuencia de descarga es mayor a la del nodo sinusal).

- Bloqueos: Se habla de bloqueo cuando la conducción del potencial de acción a través del sistema cardionector, se halla impedida. De esta manera, el potencial de acción puede extinguirse en el punto bloqueado, ser conducido por el sistema cardionector a una velocidad menor de la normal, o ser conducido por el miocardio circundante. Los bloqueos se clasifican en tres grados dependiendo de su severidad. Los bloqueos de primer grado son los más leves, sólo producen un retraso de la conducción. En el bloqueo de segundo grado se produce una ausencia intermitente de la conducción mientras que en el bloque de tercer grado, se produce una interrupción total de la conducción. Los bloqueos más comunes son los del nodo aurículo-ventricular y de las ramas del haz de His. Un bloqueo puede ser causado debido a daños en el tejido (por ejemplo infarto miocárdico) o a que las fibras en cuestión se encuentran en período refractario absoluto y no pueden ser excitadas en ese momento (fenómeno que suele ocurrir durante la contracción de potenciales marcapasos ectópicos).
- Mecanismos de reentrada: Un mecanismo de reentrada es “un trastorno en la conducción, que permite que una onda de excitación se propague en forma continua dentro de un circuito cerrado” [1]. En condiciones normales, una vez producida la despolarización del miocardio, el potencial de acción se extingue ya que todas las fibras se encuentran en período refractario y no pueden volver a ser excitadas. Sin embargo, en ciertas patologías es posible que este proceso se vea alterado y el impulso excitatorio continúe circulando por tiempo indefinido. Los mecanismos de reentrada pueden producirse tanto en las aurículas como en los ventrículos o en el nodo aurículo-ventricular.
- Fibrilación: Durante la fibrilación, cada fibra muscular se contrae de manera independiente y completamente desorganizada. El fenómeno de fibrilación está fuertemente asociado al de los mecanismos de reentrada, y puede producirse en las aurículas y los ventrículos de manera independiente. La fibrilación ventricular sostenida por más de tres minutos es mortal ya que al no producirse contracción muscular coordinada, se deja de bombear sangre al resto del cuerpo.
- Fármacos: Son muchos los fármacos que alteran el ritmo cardíaco, especialmente los que se usan de forma controlada para el tratamiento de enfermedades cardíacas. También las drogas ilegales, como la cocaína, que aumenta el ritmo cardíaco y el riesgo de arritmias y de infarto al disminuir el aporte sanguíneo al corazón.

De la combinación de las causas antes mencionadas, pueden producirse numerosas arritmias distintas, con patrones más o menos complejos, cuyo estudio escapa al presente proyecto integrador. La elección de las patologías a incluir en el trabajo se basó en la cantidad de ejemplos del latido que posee la base de datos utilizada. Por este motivo se excluyeron patologías como las

isquemias e infartos que, si bien son patologías cardíacas de gran importancia, no poseen un número significativo de muestras que permita formar un conjunto de validación de la clasificación.

A continuación se presentan las características más importantes de las arritmias sobre las cuales se desarrolla el resto del presente proyecto integrador:

Contracciones Prematuras

Una contracción prematura, latido prematuro o extrasístole es la contracción del músculo cardíaco antes del momento en que se debería haber producido una contracción normal. Tal como se dijo anteriormente, la causa más frecuente de este tipo de arritmia es la existencia de marcapasos ectópicos, aunque también puede presentarse en individuos sanos. Las contracciones prematuras pueden ocurrir en las aurículas, en los ventrículos y con menor frecuencia en el nodo aurículo-ventricular.

En una contracción auricular prematura la despolarización de las aurículas ocurre demasiado temprano en el ciclo cardíaco, por lo que la onda P del ECG suele encontrarse superpuesta sobre la onda T del latido anterior. Dicha despolarización puede haber sido iniciada por el nodo sinusal o en un foco ectópico de las aurículas, en cuyo caso el intervalo PQ poseerá una duración menor dependiendo de la distancia entre el mismo y el nodo aurículo-ventricular. Por otro lado, el potencial de acción generado por el marcapaso anormal puede despolarizar el nodo sinusal, ocasionando un retardo del latido siguiente (fenómeno que se conoce como pausa compensadora).

Dependiendo de qué tan prematuramente se produzca el latido, la despolarización auricular puede seguir tres caminos distintos:

- Ser bloqueado por el nodo aurículo-ventricular que se encuentra en período refractario, lo que resulta en una onda P aislada.
- Ser conducido por el nodo AV pero encontrar parte del sistema His-Purkinje en período refractario (conducción aberrante). En este caso se produce un complejo QRS de duración mayor a la normal.
- Ser conducido normalmente.

En la ilustración 10 se muestra un segmento electrocardiográfico que contiene una extrasístole auricular conducida con normalidad.

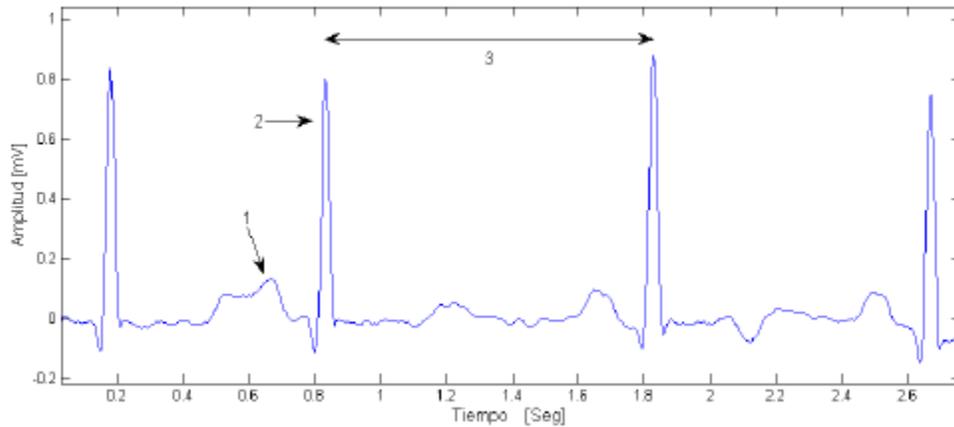


Ilustración 10 – Contracción auricular prematura.¹²

En una contracción ventricular prematura el foco ectópico se encuentra en los ventrículos. De este modo, la despolarización viaja principalmente por las fibras musculares de conducción lenta y no por el sistema de His-Purkinje, lo cual se manifiesta de tres maneras distintas en el ECG:

- Complejo QRS de duración mayor a la normal, debido a la baja velocidad de conducción del músculo miocárdico.
- Mayor amplitud del complejo QRS, provocado por la no simultaneidad de la contracción ventricular izquierda y derecha.
- Onda T de polaridad opuesta al complejo QRS, dado que las fibras que primero se despolarizaron se repolarizan antes (por la baja velocidad de conducción ventricular).

En este tipo de contracción prematura también existe una pausa compensadora, aunque la misma es de mayor duración y se debe a que la despolarización auricular se produce simultáneamente con la ventricular (por lo tanto la onda T queda tapada por el complejo QRS) y en consecuencia no se produce la propagación del mismo a los ventrículos.

En la ilustración 11 puede observarse un segmento de ECG que contiene una extrasístole ventricular.

¹² Referencias: 1) Onda P anormal. 2) Complejo QRS normal. 3) Pausa compensadora.

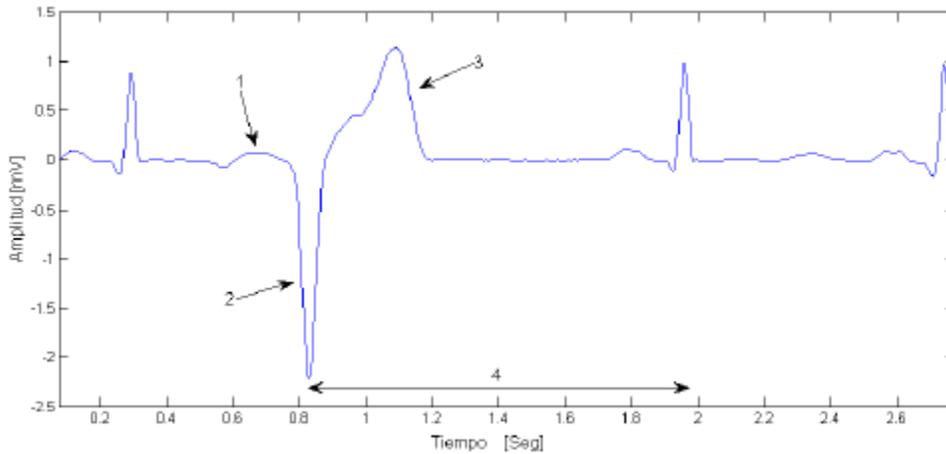


Ilustración 11 – Contracción ventricular prematura.¹³

Bloqueo de rama

Tal como su nombre lo indica, este tipo de arritmias, consisten en bloqueos de las ramas del haz de His, ya sea izquierda o derecha (ilustraciones 12 y 13 respectivamente). En estos casos, el potencial de acción es conducido normalmente por la rama intacta, regresando a través del músculo cardíaco para producir la despolarización de la zona afectada. De este modo, la duración del intervalo RR es normal, pero el complejo QRS es más prolongado. Las características morfológicas típicas del complejo QRS se deben a la no simultaneidad en la contracción de los ventrículos.

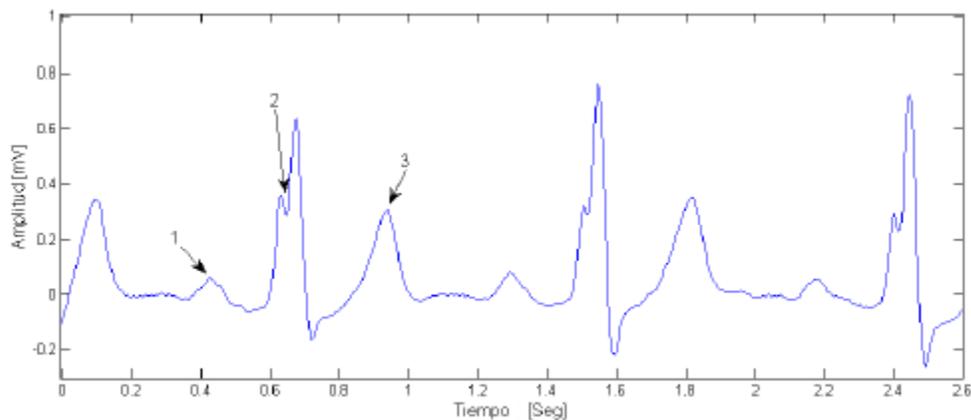


Ilustración 12 – Bloqueo de rama izquierda.¹⁴

¹³ Referencias: 1) Ausencia de onda P. 2) Complejo QRS prolongado y de mayor amplitud. 3) Onda de polaridad opuesta al complejo QRS. 4) Pausa compensadora.

¹⁴ Referencias: 1) 3) Onda P y T normal. 2) Complejo QRS prolongado y dentado.

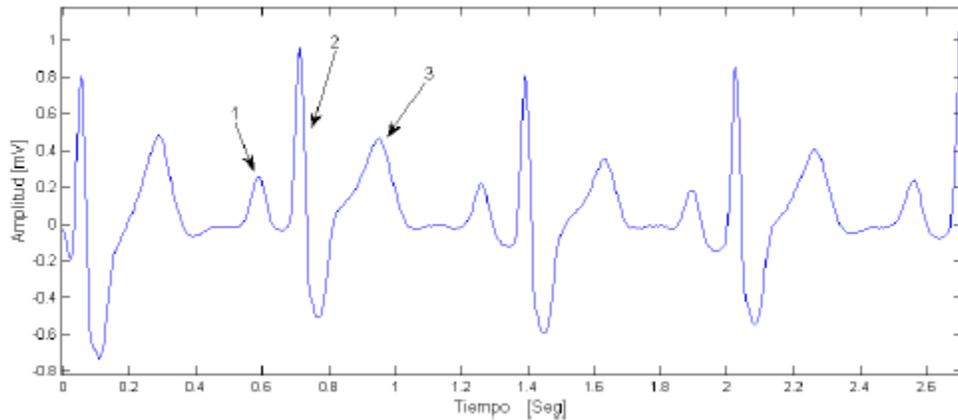


Ilustración 13 – Bloque de rama derecha.¹⁵

Latidos originados por marcapasos externos implantables

En determinados casos, debido a diversas patologías del automatismo cardíaco, es necesario implantar un dispositivo que genere la despolarización del miocardio a través de impulsos eléctricos. En esta situación, el latido se origina en aquellos lugares en los que se encuentren implantados los electrodos del marcapasos, y la morfología de las ondas del ECG dependerá en gran manera de las características del mismo. En la ilustración 14 se puede observar un latido generado por un marcapasos externo implantable extraído de la base de datos de arritmias del MIT-BIH.

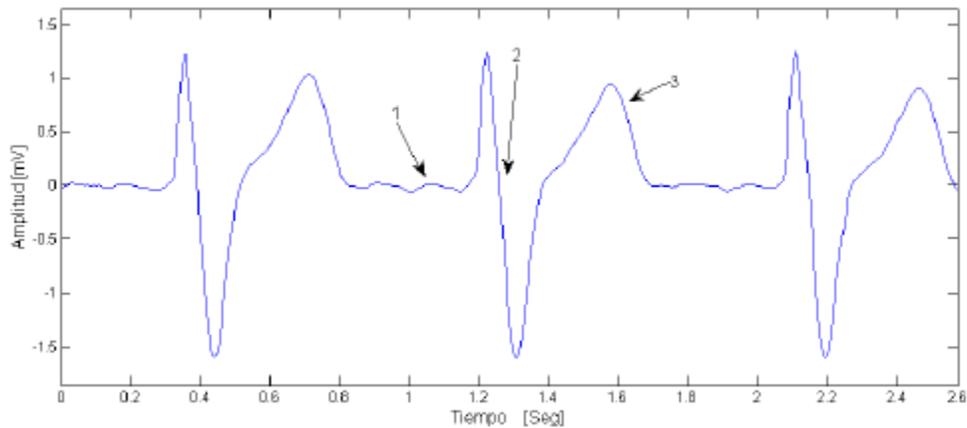


Ilustración 14 – Latido iniciado por marcapasos externo implantable.¹⁶

¹⁵ Referencias: 1) 3) Onda P y T normal. 2) Complejo QRS prolongado y bifásico.

¹⁶ Referencias: 1) Ausencia de onda P. 2) Complejo QRS prolongado y bifásico. 3) Onda T normal.

Procesamiento de la señal de ECG

Para lograr clasificar los latidos procedentes de un registro electrocardiográfico, es necesaria la manipulación del mismo para poder utilizar la información que contiene. Para ello, se debe realizar una serie de procesos, los cuales pueden agruparse en tres etapas principales que son: la detección de latidos, la extracción de características y su clasificación. A continuación se muestra un diagrama en bloques general de las etapas de procesamiento que se realizan.

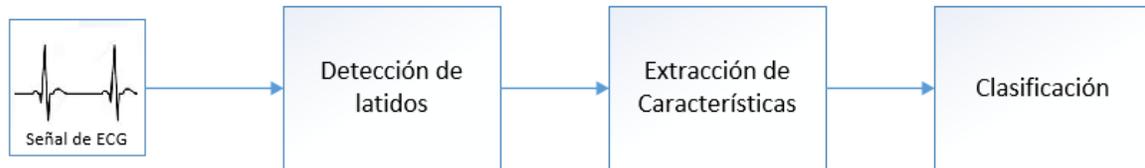


Ilustración 15 - Esquema general de las etapas de procesamiento.

En la etapa de Detección, la señal se filtra para luego, mediante un algoritmo, detectar cada latido y segmentarlo individualmente. Posteriormente, se extraen características de estos latidos para luego ser clasificados mediante estrategias de inteligencia artificial.

En las siguientes secciones se describen las técnicas empleadas para la realización de cada una de las etapas mencionadas.

Detección de latidos

Dependiendo de las condiciones de tiempo disponible para observar la señal de ECG, extraer los complejos QRS y clasificarlos, existen tres tipos de algoritmos de detección:

- Algoritmo Off-line: Son aquellos en los que se cuenta con toda la señal a analizar de antemano, pudiendo el algoritmo recorrer la señal hacia atrás y hacia adelante sin restricciones.
- Algoritmo On-line: Son aquellos que trabajan sobre una señal que está siendo generada en ese momento, es decir que no pueden conocer la señal más allá de la última muestra que se recibió. Estos algoritmos tienen un margen entre dos y tres segundos para detectar los latidos, lo cual les da la posibilidad de repasar la señal en caso de perderse un latido.
- Algoritmos Real-Time: Estos son similares a los On-line con la diferencia de que deben detectar cada latido antes de que llegue el próximo.

Cuando los filtros digitales son utilizados en tiempo real, muchos diseños de filtros se vuelven inadecuados debido al tiempo computacional requerido. Una considerable reducción de tiempo computacional se logra reemplazando los coeficientes de punto flotante por pequeños coeficientes enteros en las ecuaciones de los filtros. Esto se debe a que al utilizar coeficientes enteros se evita la resolución de multiplicaciones de punto flotante reemplazándolas por instrucciones de suma y cambios de posición de bits. Estas instrucciones requieren menos ciclos de clock que los cálculos de punto flotante.

Filtros enteros

Los filtros enteros son una clase especial de filtro digital que tienen solamente coeficientes enteros en las ecuaciones que los definen. Esto conduce a algunas restricciones de diseño que a menudo puede hacer que sean difíciles de lograr características tales como una frecuencia de corte definida. Desde que los filtros enteros pueden operar a mayor velocidad que los diseños tradicionales, generalmente son el tipo de filtro adecuado cuando se utilizan microprocesadores lentos.

Lynn [4] presentó la mejor técnica conocida para el diseño de filtros enteros. Se resume este método en dos pasos:

1. Se ubica un número de ceros¹⁷ igualmente espaciados alrededor del círculo unidad. Estos ceros atenúan completamente las frecuencias correspondientes a su ubicación.
2. El siguiente paso es elegir polos¹⁸ que también se encuentran en el círculo unidad, para eliminar algunos de los ceros.

Cuando un polo cancela un cero, la frecuencia correspondiente a su localización ya no es atenuada. Dado que cada punto en el círculo unidad representa una frecuencia, la localización de los polos y ceros determina la respuesta de frecuencia del filtro. Estos filtros son un tipo restringido de filtros recursivos.

Forma general de la función de transferencia

La forma general de la función de transferencia del filtro utilizada por Lynn es:

$$H_1(z) = \frac{[1 - z^{-m}]^p}{[1 - 2 \cos(\theta) z^{-1} + z^{-2}]^t}$$

$$H_2(z) = \frac{[1 + z^{-m}]^p}{[1 - 2 \cos(\theta) z^{-1} + z^{-2}]^t}$$

El exponente m representa cuantos ceros igualmente espaciados se encuentran alrededor del círculo unidad. El ángulo θ representa la posición angular de los polos. Las potencias p y t representan el orden de magnitud del filtro, el cual tiene una relación directa con la ganancia y la atenuación de los lóbulos laterales. Aumentar p y t en cantidades enteras iguales tiene el efecto de poner filtros idénticos en cascada. Para que el filtro sea útil y físicamente realizable, p y t tienen que ser enteros positivos.

Ubicación de los polos

El denominador de la función de transferencia viene de la multiplicación de un par de polos complejos conjugados que siempre se encuentran exactamente en el círculo unidad. La relación de

¹⁷ Ceros: raíces del numerador.

¹⁸ Polos: raíces del denominador.

Euler, $e^{j\theta} = \cos(\theta) + j\text{sen}(\theta)$ muestra que todos los valores de $e^{j\theta}$ se ubican en el círculo unidad.

$$(z - e^{j\theta})(z - e^{-j\theta})$$

Multiplicando los dos factores,

$$z^2 - (e^{j\theta} + e^{-j\theta})z + (e^{j\theta}e^{-j\theta})$$

Usando la identidad

$$\cos(\theta) = \frac{e^{j\theta} + e^{-j\theta}}{2}$$

Llegamos a

$$\text{denominador} = 1 - 2\cos(\theta)z^{-1} + z^{-2}$$

El par de polos complejos conjugados del denominador de la función de transferencia provee coeficientes enteros solo cuando $2\cos(\theta)$ es entero. Para ello θ tiene que ser igual a 0° , $\pm 60^\circ$, $\pm 90^\circ$, $\pm 120^\circ$, y 180° .

θ	$\cos(\theta)$	$2\cos(\theta)$
0°	1	2
$\pm 60^\circ$	$+1/2$	1
$\pm 90^\circ$	0	0
$\pm 120^\circ$	$-1/2$	-1
180°	-1	-2

Tabla 3- Únicas localizaciones donde $2\cos(\theta)$ resulta en coeficientes enteros.

Estos filtros tienen un beneficio agregado que es su característica de tener fase lineal, es decir, que todas las componentes en frecuencia de la señal sufren el mismo retraso en la transmisión a través del filtro. Esto es importante cuando se desea preservar el tiempo relativo de los picos y las características de la forma de onda de la salida.

Filtro entero pasa-bajo

Usando la función de transferencia general, podemos diseñar un filtro pasa-bajo ubicando un polo en $z = (1,0)$. Sin embargo, el denominador produce dos polos en $z = (1,0)$ cuando $\cos(\theta) = 0$. Este problema puede resolverse agregando otro cero en $z = (1,0)$ o quitando un polo. La mejor solución es remover un polo ya que esto crea una función de transferencia más corta y, por lo tanto, más eficiente. La función de transferencia para un filtro entero recursivo pasa-bajo es:

$$H(z) = \frac{1 - z^{-m}}{1 - z^{-1}}$$

Este filtro tiene un lóbulo de baja frecuencia que es mayor en magnitud que los lóbulos de alta frecuencia; así, amplifica más las bajas frecuencias que las altas ubicadas en los lóbulos laterales. Estos lóbulos laterales resultan de la localización de polos en el origen del plano z .

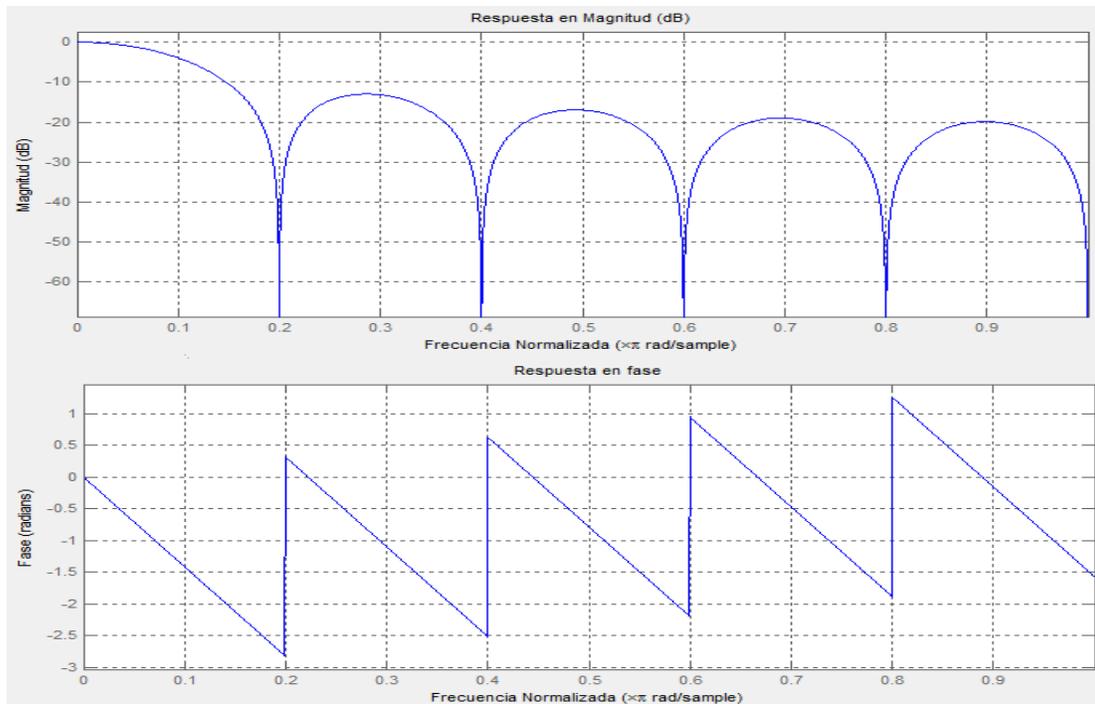


Ilustración 16 - Filtro pasa-bajo con $m = 10$. Respuesta en magnitud y en fase.

Filtro entero pasa-alto

La respuesta en frecuencia de un filtro compuesto formado por la suma de dos, o más, filtros de fase lineal con el mismo retraso de transmisión es igual a la suma algebraica de las respuestas individuales (Alstrom y Tompkins). Un filtro pasa-alto $H_{high}(z)$ puede ser diseñado por la substracción de un filtro pasa-bajo $H_{low}(z)$ desde un filtro pasa-todo. Un filtro pasa-todo es una red de puro retraso con ganancia constante. Su función de transferencia puede ser representada como:

$$H_a(z) = Az^{-m}$$

Donde A es igual a la ganancia y m es igual al número de ceros. Idealmente $H_a(z)$ debe tener la misma ganancia que $H_{low}(z)$ para dc para que la diferencia sea cero. Además, el filtro puede operar más rápidamente si $A = 2^i$ donde i es un entero para poder escalar el filtro pasa-alto con instrucciones de cambio de posición. Para lograr la menor distorsión de fase, el número de unidades de retraso del filtro pasa-todo debe ser igual al número de unidades de retraso del filtro pasa-bajo. De modo que se requiere que $\angle H_a(z) = \angle H_{low}(z)$. Un filtro pasa-bajo con muchos ceros y, por ende, una baja frecuencia de corte produce un filtro pasa-alto con una baja frecuencia de corte.

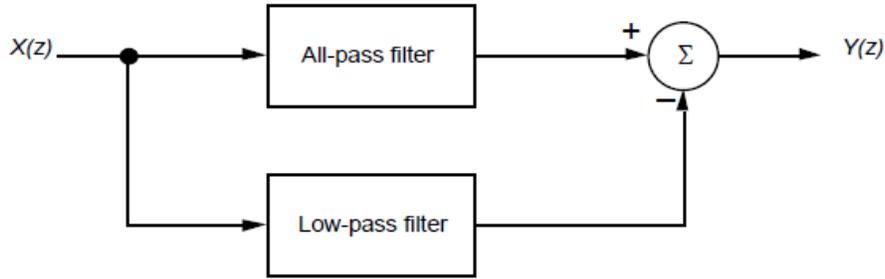


Ilustración 17 - Diagrama de bloque de un filtro pasa-alto basado en sustracción de filtro.¹⁹

Extracción de Características

La etapa de extracción de características es la etapa clave para lograr la correcta clasificación de los latidos cardíacos utilizando la señal de ECG. Cualquier información extraída del latido, utilizada para determinar su clasificación, puede ser considerada como característica.

Análisis de Componentes Independientes

Antes de adentrarnos en el análisis de componentes independientes, es necesario definir dos conceptos: “Cocktail party problem” y “Blind source separation”.

Cocktail Party Problem y Blind Source Separation.

Imaginemos que estamos en un cuarto donde dos personas hablan simultáneamente. En el mismo se tienen dos micrófonos dispuestos en diferentes lugares, los cuales reciben dos señales temporales que podemos llamar $x_1(t)$ y $x_2(t)$. Cada una de estas señales es una suma ponderada de las señales emitidas por las personas que hablan, que llamaremos $s_1(t)$ y $s_2(t)$. Entonces podemos expresar:

$$\begin{aligned} x_1(t) &= a_{11}s_1(t) + a_{12}s_2(t) \\ x_2(t) &= a_{21}s_1(t) + a_{22}s_2(t) \end{aligned}$$

Donde los a_{ij} dependen de la ubicación de los micrófonos con respecto a las personas.

Resulta de interés poder estimar las señales $s_1(t)$ y $s_2(t)$ utilizando únicamente la información contenida en las señales grabadas $x_1(t)$ y $x_2(t)$. A esto se lo conoce como “cocktail party problem”.

Por su parte, Blind source separation (BSS) consiste en la recuperación de señales independientes desconocidas desde la observación de una combinación lineal de las mismas. Para ello suponemos que se tienen m señales desconocidas, una matriz de mezcla $A_{n \times m}$, y por tanto n combinaciones lineales de las señales desconocidas:

$$x = A \cdot s$$

¹⁹ Imagen obtenida de W.J. Tompkins; V.X. Afonso. Biomedical Digital Signal Processing.

Donde:

$$\begin{aligned}x(t) &= [x_1(t) \ x_2(t) \ \dots \ x_n(t)]^T \\s(t) &= [s_1(t) \ s_2(t) \ \dots \ s_m(t)]^T\end{aligned}$$

El supraíndice T hace referencia a la transpuesta (se denomina transpuesta de una matriz, a la matriz que se obtiene cambiando ordenadamente las filas por las columnas). La columna a_k de A es denominada vector direccional asociado a la k-ésima fuente $s_k(t)$.

El Análisis de Componentes Independientes (ICA) es una clase de técnica para aplicar BSS, la cual requiere que el número de señales recibidas n sea al menos igual al número de fuentes independientes m . Por simplicidad, supondremos que la matriz de mezcla es cuadrada, es decir que $m=n$. En la mayoría de las aplicaciones, o bien A no está disponible, o los errores en A hacen que las estimaciones no sirvan. En estas situaciones lo mejor es no asumir nada acerca de A.

Definición de Análisis de Componentes Independientes (ICA) [5]

Supongamos, como antes, que observamos n mezclas lineales x_1, x_2, \dots, x_n de n componentes independientes:

$$x_j = a_{j1}s_1 + a_{j2}s_2 + \dots + a_{jn}s_n \quad \text{para todo } j$$

En el modelo de ICA, asumimos que cada mezcla x_j y cada componente independiente s_k es una variable aleatoria. Entonces los valores observados $x_j(t)$ son muestras de dicha variable aleatoria. Sin pérdidas de generalidad, podemos asumir que tanto las mezclas como las fuentes son de media nula. Si no fuera cierto, las variables x_j pueden ser centradas, restando la media de las muestras. Es conveniente utilizar una notación vectorial, por lo que llamamos a x al vector aleatorio cuyas componentes son las mezclas x_1, \dots, x_n y de la misma manera el vector s aleatorio, con elementos s_1, \dots, s_n . Llamamos A a la matriz con elementos a_{ij} . Siempre consideraremos los vectores como columnas, entonces, x^T es un vector fila.

$$x = A \cdot s$$

La ecuación anterior es llamada modelo ICA y describe como son generados los datos observados mediante la mezcla de las fuentes s . Las fuentes no pueden ser directamente observadas. También se asume desconocida la matriz A. Todo lo que tenemos es el vector aleatorio x , y con éste debemos estimar A y s . Esto debe hacerse bajo supuestos lo más generales posibles.

El punto de partida de ICA es una simple suposición de que los componentes s son estadísticamente independientes. Se verá más adelante que se debe asumir que los componentes independientes deben tener distribuciones no-gaussianas. Sin embargo, en el modelo básico, no asumimos conocidas esas distribuciones.

Luego de estimar A, podemos calcular su inversa, llámese W, y obtener los componentes independientes:

$$s = W \cdot x$$

Ambigüedades de ICA

No podemos determinar las varianzas (energías) de las componentes independientes. La razón es que, al ser A y s desconocidos, cualquier factor multiplicativo en una de las fuentes s_i puede siempre cancelarse, dividiendo la correspondiente columna a_i de A por el mismo factor. Como consecuencia, debemos adecuar las magnitudes de las componentes independientes. Como son variables aleatorias, la forma más natural de hacerlo es asumir varianza unitaria: $E\{s_i^2\}=1$.²⁰ La matriz A será adaptada en la solución de ICA, teniendo en cuenta esta restricción.

No podemos determinar el orden de las componentes independientes. Al ser A y s desconocidos, podemos cambiar libremente el subíndice de los s_i . Formalmente, una matriz de permutación P y su inversa pueden agregarse al modelos para dar $x=AP^{-1}Ps$. Los elementos de Ps son las s_i originales, pero cambiadas de orden. La matriz AP^{-1} es una nueva matriz de mezcla desconocida, a determinarse por los algoritmos de ICA.

Independencia, definición y propiedades

Consideremos dos variables aleatorias escalares y_1 e y_2 . Básicamente, dichas variables son independientes si la información de valor de una no aporta nada de información sobre el valor de la otra. Técnicamente, la independencia puede definirse por las densidades de probabilidad. Definimos que y_1 e y_2 son independientes si y solo si la función de densidad de probabilidad conjunta es factorizable de la siguiente manera:

$$p(y_1, y_2) = p(y_1) \cdot p(y_2)$$

Siendo $p(y_1)$ y $p(y_2)$ las funciones de densidad marginales de y_1 e y_2 respectivamente. Esta definición se extiende naturalmente para n variables aleatorias.

La definición puede ser usada para derivar una de las propiedades más importantes de las variables aleatorias independientes. Dadas dos funciones h_1 y h_2 , se tiene:

$$E\{h_1(y_1) \cdot h_2(y_2)\} = E\{h_1(y_1)\} \cdot E\{h_2(y_2)\}$$

Una forma más débil de independencia es la no correlación. Dos variables aleatorias y_1 e y_2 son no correlacionadas, si su covarianza es nula:

$$E\{y_1 \cdot y_2\} - E\{y_1\} \cdot E\{y_2\} = 0$$

Independencia implica no correlación, pero no al revés. Algunos métodos de ICA proceden estimando las componentes independientes con señales no correlacionadas. Esto simplifica el problema.

²⁰ $E\{\}$ es el operando Esperanza. La esperanza matemática o valor esperado de una variable aleatoria discreta es la sumatoria del producto de la probabilidad de cada suceso por el valor de dicho suceso.

Inconveniente con variables gaussianas

Para ver por qué variables gaussianas hacen imposible a ICA, supongamos que la matriz de mezcla es ortogonal, y que las s_i son gaussianas. Entonces x_1 y x_2 son gaussianas, no correlacionadas, y de varianza unitaria. Su densidad conjunta está dada por:

$$p(x_1, x_2) = \frac{1}{2\pi} e^{-\left(\frac{x_1^2 + x_2^2}{2}\right)}$$

La densidad es simétrica, por lo tanto no contiene información de la dirección de las columnas de la matriz de mezcla A , por lo que la misma no puede estimarse.

El fenómeno que explica que las matrices de mezcla ortogonales no se pueden calcular a partir de variables gaussianas, está relacionado con la propiedad que indica que variables gaussianas no correlacionadas entre si son necesariamente independientes. Por tanto, la información que reside en la independencia no nos permite llegar más allá del blanqueo.²¹ En otras palabras, la matriz de mezcla A no se puede calcular para variables gaussianas independientes. Con las variables gaussianas a lo más que podemos llegar es a blanquear los datos, por ejemplo usando PCA.

Principios de la estimación ICA

Intuitivamente hablando, la clave para la estimación del modelo ICA es la no-gaussianidad. Sin ella, la estimación se hace imposible.

El teorema central del límite dice que la distribución de una suma de variables aleatorias independientes tiende a una gaussiana, bajo ciertas condiciones. Por lo tanto, la suma de dos variables aleatorias independientes tiene, en general, una distribución más gaussiana que las originales variables aleatorias.

Ahora asumamos que el vector x está distribuido acorde al modelo ICA, o sea, es una mezcla de componentes independientes. Por simplicidad, en esta sección supongamos que todas las componentes independientes tienen idénticas distribuciones.

Para estimar uno de las componentes independientes consideramos una combinación lineal de los x_i de la forma:

$$y = w^T x = \sum_i w_i x_i$$

Donde w es un vector a determinar. Si w fuese una de las filas de la inversa de A , la combinación lineal sería igual a una de las componentes independientes. La cuestión está en cómo usar el teorema central del límite para determinar w y que sea igual, o lo más parecido, a una de las filas de la inversa de A . En la práctica no se puede determinar exactamente w , porque no conocemos la matriz A , pero se puede llegar a una buena aproximación. Para ver como esto lleva al principio básico de la estimación con ICA, hagamos un cambio de variable $z=A^T w$.

²¹ El concepto de blanqueo se explica en la sección “pre-procesamiento para ICA”.

$$y = w^T x = w^T A s = z^T s$$

Entonces, y es una combinación lineal de los s_i ponderados por z_i .

Dado que la suma de incluso dos variables aleatorias independientes es más gaussiana que las originales, $z^T s$ es más gaussiana que cualquiera de las s_i , y se vuelve menos gaussiana cuando es, de hecho, igual a alguna s_i . En este caso, obviamente solo uno de los elementos de z_i es distinto de cero. Por lo tanto, podemos tomar como w un vector que maximice la no-gaussianidad de $w^T x$. Dicho valor necesariamente se corresponderá con un z que tenga sólo un componente no nulo. Esto significa que:

$$w^T x = z^T s$$

O sea, una de las componentes independientes.

Entonces, maximizar la no-gaussianidad de $w^T x$ nos da una de las componentes independientes. De hecho, el tope de la optimización de la no-gaussianidad en dimensión n tiene $2*n$ máximos locales, dos para cada componente independiente, correspondientes a $\pm s_i$. Para encontrar todas las componentes independientes, debemos hallar todos esos máximos locales. Esto no es difícil, porque las diferentes componentes independientes son no correlacionadas y siempre podremos restringir la búsqueda al espacio que brinda estimaciones no correlacionadas con las previas. Esto corresponde a la ortogonalidad de un espacio adecuadamente transformado.

Nuestro enfoque fue puramente heurístico, pero veremos en las siguientes secciones que tiene una rigurosa justificación.

Medida de la No-Gaussianidad

Para usar la no-gaussianidad en la estimación con ICA, debemos tener una medida cuantitativa de la no-gaussianidad de una variable aleatoria, llámese y .

Para simplificar, asumamos que y está centrada (media nula) y de varianza unitaria. Más adelante se verá un procedimiento de pre-procesamiento para siempre lograr esta simplificación.

Kurtosis

La medida clásica de la no-gaussianidad es el Kurtosis o “cumulante de cuarto orden”.

$$kurt(y) = E\{y^4\} - 3(E\{y^2\})^2$$

En realidad, si asumimos varianza unitaria para y , tenemos:

$$kurt(y) = E\{y^4\} - 3$$

Esto muestra que el kurtosis es una versión normalizada del cuarto momento $E\{y^4\}$. Para una y gaussiana:

$$kurt(y) = 0 \quad \text{para } y \text{ gaussiana}$$

Para la mayoría de las variables aleatorias no gaussianas, el kurtosis no se anula. Éste puede ser positivo o negativo. Variables aleatorias con kurtosis negativo son llamadas sub-gaussianas, mientras que con kurtosis positivo son súper-gaussianas. Típicamente la no-gaussianidad se mide con el valor absoluto del kurtosis. El cuadrado del mismo también puede usarse. Ambos son nulos para variables gaussianas, y mayores que cero cuanto más no-gaussiana sea la variable.

El kurtosis, o su valor absoluto, ha sido ampliamente utilizado como medida de no-gaussianidad en ICA y campos relacionados. La principal razón es su simplicidad, tanto computacional como teórica. Computacionalmente, el kurtosis puede ser estimado usando el cuarto momento de las muestras de los datos. El análisis teórico se simplifica por la siguiente propiedad de linealidad (se deriva de la definición); Si x_1 y x_2 son dos variables aleatorias independientes:

$$\begin{aligned} kurt(x_1 + x_2) &= kurt(x_1) + kurt(x_2) \\ kurt(\alpha \cdot x_1) &= \alpha^4 \cdot kurt(x_1) \end{aligned}$$

Siendo α un escalar.

En la práctica comenzamos con cierto vector w , calculamos la dirección en la cual el kurtosis de $y=w^T x$ crece más (si $kurt > 0$) o decrece más (si $kurt < 0$), basados en las muestras disponibles $x(1), \dots, x(T)$ del vector mezclado x , y usamos algún método de gradiente o alguna de sus extensiones para encontrar un nuevo vector w .

Sin embargo el kurtosis tiene algunos puntos en contra en la práctica cuando su valor tiene que ser estimado con pocas muestras. El principal problema es que el kurtosis puede ser muy sensible a valores atípicos. En otras palabras, el kurtosis no es una medida robusta de la no-gaussianidad.

Negentropía

La entropía es el concepto básico de la teoría de la información. La entropía de una variable aleatoria puede ser interpretada como el grado de información que aporta la observación de la variable. Cuando más aleatoria, impredecible e inestructurada es la variable, mayor es su entropía. La entropía H está definida para una variable aleatoria como:

$$H(Y) = - \sum P(Y = a_i) \cdot \log P(Y = a_i) \quad \text{para } Y \text{ discreta}$$

Donde a_i son los posibles valores de Y .

Un resultado fundamental en teoría de la información es que una variable gaussiana posee la mayor entropía entre todas las variables aleatorias de igual varianza. Esto significa que la entropía puede ser usada como medida de no-gaussianidad. De hecho, esto muestra que la distribución gaussiana es la “más aleatoria”, o la menos estructurada de todas las distribuciones. La entropía es chica para distribuciones que están claramente concentradas en ciertos valores. Para obtener una medida de no-gaussianidad, que se anule para una variable gaussiana, y sea siempre no negativa, se usa una versión modificada de la definición de entropía diferencial, llamada negentropía:

$$J(y) = H(y_{gauss}) - H(y)$$

Donde y_{gauss} es una variable aleatoria gaussiana con la misma matriz de covarianza que y .

Dado lo mencionado arriba, la negentropía es siempre no negativa, y sólo se anula si y tiene distribución gaussiana. La negentropía además es invariante ante transformaciones lineales invertibles. La ventaja de usar negentropía o entropía diferencial como medida de no-gaussianidad es que tiene una fuerte justificación teórica. De hecho, en algunos sentidos es el mejor estimador de no-gaussianidad. El problema es que es computacionalmente muy costosa de calcular (requiere estimar la función de densidad, probablemente no paramétrica). Por lo tanto, las aproximaciones simples de la negentropía son muy útiles.

Aproximaciones de la negentropía

El clásico método para aproximar la negentropía es usando momentos de alto orden:

$$J(y) \sim \frac{1}{12} E\{y^3\}^2 + \frac{1}{48} kurt(y)^2$$

Se asume que la variable aleatoria y tiene media nula y varianza unitaria. Sin embargo, la validez de esta aproximación puede ser bastante limitada. En particular, esta aproximación sufre los mismos problemas de robustez que el kurtosis. Para evitar estos problemas, se desarrollaron nuevas aproximaciones basadas en el principio de máxima entropía:

$$J(y) \sim \sum_{i=1}^p k_i [E\{G_i(y)\} - E\{G_i(v)\}]^2$$

Donde k_i son constantes positivas, y v es una variable aleatoria gaussiana de media nula y varianza unitaria. Se asume que y es de media nula y varianza unitaria, y las funciones G_i son no cuadráticas. En el caso que usemos sólo una función no cuadrática G , la aproximación es:

$$J(y) \sim [E\{G(y)\} - E\{G(v)\}]^2$$

El punto está en elegir adecuadamente G para obtener mejores aproximaciones de la negentropía. En particular, eligiendo G que no crezca muy rápido, se obtienen estimadores más robustos. Las siguientes funciones son útiles:

$$G_1(u) = \frac{1}{a_1} \log \cosh a_1 u \quad G_2 = -e^{-\frac{u^2}{2}}$$

Donde $1 \leq a_1 \leq 2$ es una constante adecuada y \cosh es el coseno hiperbólico.

Pre-procesamiento para ICA

Antes de aplicar ICA sobre los datos, es muy útil realizar cierto pre-procesado. En esta sección veremos determinadas técnicas de procesamiento para hacer el problema de estimación ICA más simple y mejor condicionado.[6]

Centrado

El más básico y necesario pre-procesamiento es centrar x , restándole el vector medio $E\{x\}$, haciendo al vector x de media nula. Esto implica que también s es de media nula, dado que:

$$s = A^{-1}x$$

$$E\{s\} = A^{-1}E\{x\} = 0$$

Este pre-procesamiento es únicamente para simplificar los algoritmos ICA, no significa que las medias no puedan estimarse. Luego de estimar la matriz de mezcla A con los datos centrados, podemos completar la estimación agregando el vector de medias de s al vector estimado centrado. Dicho vector de medias está dado por $A^{-1}m$, donde m es el vector antes restado.

Blanqueo (Whitening)

Otra estrategia útil de pre-procesamiento en ICA es primero blanquear las variables observadas. Esto significa que antes de la aplicación de los algoritmos de ICA, pero después del centrado, transformamos linealmente el vector observado x para obtener un nuevo vector \tilde{x} , el cual es blanco, es decir que sus componentes son no correlacionadas y sus varianzas son unitarias. En otras palabras, la matriz covarianza de \tilde{x} es igual a la identidad.

$$E\{\tilde{x} \cdot \tilde{x}^T\} = I$$

La transformación de blanqueo es siempre posible. Un método popular para hacerlo es usar la descomposición en valores propios de la matriz de covarianza.

$$E\{x \cdot x^T\} = EDE^T$$

Donde E es la matriz ortogonal de vectores propios de $E\{x \cdot x^T\}$ y D es la matriz diagonal con sus valores propios, $D = \text{diag}(d_1, \dots, d_n)$. El blanqueo puede ahora realizarse:

$$\tilde{x} = ED^{-1/2}E^T x$$

Donde la matriz $D^{-1/2}$ es calculada elemento a elemento: $D^{-1/2} = \text{diag}(d_1^{-1/2}, \dots, d_n^{-1/2})$.

El blanqueo transforma la matriz de mezcla en una nueva \tilde{A} .

$$\tilde{x} = ED^{-1/2}E^T A s = \tilde{A} s$$

La utilidad del blanqueo reside en que la nueva matriz de mezcla \tilde{A} es ortogonal:

$$E\{\tilde{x} \cdot \tilde{x}^T\} = \tilde{A} E\{s s^T\} \tilde{A}^T = \tilde{A} \tilde{A}^T = I$$

Aquí vemos que el blanqueo reduce el número de parámetros a ser estimados. En vez de tener que estimar n^2 parámetros (elementos de A), sólo necesitamos estimar la nueva matriz de mezcla ortogonal \tilde{A} . Una matriz ortogonal posee $n(n-1)/2$ grados de libertad. Por ejemplo, en dos dimensiones, una transformación ortogonal es determinada por un único parámetro (ángulo). En

mayores dimensiones, una matriz ortogonal contiene aproximadamente la mitad del número de parámetros que una matriz cualquiera, por lo que se puede decir que el blanqueado resuelve la mitad del problema en ICA.

De ahora en adelante, asumimos que los datos han sido pre-procesados (centrados y blanqueados). Para simplificar la notación, denotaremos a los datos pre-procesados simplemente por x , y a la matriz de mezcla transformada como A .

Algoritmo de punto fijo usando negentropía – FastICA

FastICA es un algoritmo eficiente y popular para ICA desarrollado por Aapo Hyvarinen en la universidad de tecnología de Helsinki. Al igual que la mayoría de los algoritmos de ICA, fastICA busca una rotación ortogonal de los datos pre-blanqueados con el fin de maximizar una medida de no-gaussianidad de los componentes rotados.

FastICA para una unidad

Por unidad nos referimos a una unidad computacional, eventualmente una neurona artificial con un vector de pesos w , el cual dicha neurona es capaz de actualizar a través de una regla de aprendizaje. La regla de aprendizaje de fastICA encuentra una dirección, es decir un vector w tal que la proyección $w^T x$ maximiza la no-gaussianidad. Aquí la no-gaussianidad es medida por la aproximación de la negentropía:

$$J(y) \sim [E\{G(y)\} - E\{G(v)\}]^2$$

FastICA está basado en una estrategia de iteración de punto fijo para encontrar el máximo de no-gaussianidad de $w^T x$, medida como la ecuación anterior. Puede también derivarse como una iteración aproximativa de Newton. Llamamos g a la derivada de la función no cuadrática G . Por ejemplo:

$$G_1(u) = \frac{1}{a_1} \log \cosh a_1 u \quad G_2(u) = -e^{-\frac{u^2}{2}}$$

$$g_1(u) = \tanh(a_1 u) \quad g_2(u) = u \cdot e^{-\frac{u^2}{2}}$$

Donde $1 \leq a_1 \leq 2$ es una constante adecuada, usualmente igual a uno. La forma básica del algoritmo FastICA es la siguiente:

- 1) Elegir un vector de pesos inicial w (por ejemplo, aleatorio).
- 2) Se calcula: $w^+ = E\{xg(w^T x)\} - E\{g'(w^T x)\} \cdot w$
- 3) Se actualiza w : $w = \frac{w^+}{\|w^+\|}$
- 4) Si no converge, volver a 2)

La convergencia significa que el viejo y nuevo valor de w apuntan en la misma dirección, es decir, que su producto punto es próximo a uno. No es necesario que el vector converja a un único punto, ya que w y $-w$ definen la misma dirección. Esto, nuevamente, se debe a que las componentes independientes pueden ser definidas a menos de un signo.

En la práctica, las esperanzas matemáticas en FastICA deben reemplazarse por sus estimaciones, que naturalmente son las medias de las muestras. Idealmente deberían usarse todos los datos disponibles, pero no es una buena idea dado el costo computacional. Entonces los promedios pueden estimarse usando un subconjunto de muestras, cuyo tamaño puede influir demasiado en la precisión de la estimación final.

FastICA para varias unidades

El algoritmo para una unidad antes descrito estima sólo una de las componentes independientes. Para estimar varias componentes independientes, necesitamos correr el algoritmo de una unidad pero usando varias unidades (neuronas) con vectores de pesos w_1, \dots, w_n .

Para evitar que diferentes vectores converjan al mismo máximo debemos de-correlacionar las salidas $w_1^T x, \dots, w_n^T x$ luego de cada iteración. Una simple manera de lograr la no correlación es con una estrategia de deflación basada en una de-correlación similar a Gram-Schmidt. Esto significa estimar los componentes independientes uno a uno. Cuando tenemos estimados p componentes independientes, o p vectores w_1, \dots, w_p , corremos el algoritmo de punto fijo de una unidad para w_{p+1} y luego de cada paso iterativo le restamos a w_{p+1} las "proyecciones" $w_{p+1}^T w_j w_j$, $j = 1, \dots, p$ de los p vectores estimados previamente, y luego normalizamos w_{p+1} .

$$\begin{aligned}
 1) \quad w_{p+1} &= w_{p+1} - \sum_{j=1}^p w_{p+1}^T w_j w_j \\
 2) \quad w_{p+1} &= \frac{w_{p+1}}{\sqrt{w_{p+1}^T w_{p+1}}}
 \end{aligned}$$

Propiedades del algoritmo FastICA

El algoritmo fastICA tiene algunas propiedades muy buenas, en comparación a otros métodos existentes para ICA.

1. Bajo la asunción de validez del modelo ICA, la convergencia es cúbica (o al menos cuadrática). En otros algoritmos de ICA basados en descenso por el gradiente, la convergencia es sólo lineal. Esto significa una muy rápida convergencia.
2. Al contrario con los algoritmos basados en descenso por el gradiente, no hay ningún parámetro a seleccionar, lo que hace al algoritmo sencillo de usar.
3. El algoritmo fastICA encuentra directamente componentes independientes de casi cualquier distribución no-gaussiana usando cualquier no-linealidad g .
4. La performance del método puede optimizarse eligiendo adecuadamente la función g .

5. Las componentes independientes pueden ser estimadas una a una. Esto hace decrecer el costo computacional cuando no todas las componentes independientes necesitan ser estimadas.
6. El algoritmo FastICA tiene la mayoría de las ventajas de los algoritmos basados en redes neuronales; se realiza en paralelo y distribuido, es computacionalmente simple, y requiere poco espacio en memoria.

Análisis de Componentes Principales

Un problema central en el análisis de datos multi-variables es la reducción de la dimensionalidad: si es posible describir con precisión los valores de p variables por un pequeño subconjunto $r < p$ de ellas, se habrá reducido la dimensión del problema a costa de una pequeña pérdida de información. El Análisis de Componentes Principales (PCA) tiene ese objetivo: dadas n observaciones de p variables, se analiza si es posible representar adecuadamente esta información con un número menor de variables construidas como combinaciones lineales de las originales. Por ejemplo, con variables con alta dependencia es frecuente que un pequeño número de nuevas variables (menos del 20% de las originales) brinden más del 80% de la información que brindan las variables originales.

El análisis de componentes principales tiene doble utilidad:

1. Permite representar óptimamente en un espacio de dimensión pequeña observaciones de un espacio general p -dimensional. En este sentido, el análisis de componentes principales es el primer paso a identificar las posibles variables latentes, o no observadas que generan los datos.
2. Permite transformar las variables originales, en general correlacionadas, en nuevas variables no correlacionadas, facilitando la interpretación de los datos.

Supongamos que se dispone de los valores de p -variables en n elementos de una población dispuestos en una matriz X de dimensiones $n \times p$, donde las columnas contienen las variables y las filas los elementos. Vamos a suponer que previamente hemos restado a cada variable su media, de manera que las variables de la matriz X tienen media cero y su matriz de covarianzas vendrá dada por $1/n X^T X$. [7]

Cálculo de los componentes principales

A continuación se desarrolla el cálculo de los componentes principales. En primer lugar se explica cómo obtener los dos primeros componentes principales para, posteriormente, desarrollar la forma general.

Cálculo del primer componente

El primer componente principal se define como la combinación lineal de las variables originales que tiene varianza máxima. Los valores en esta primera componente de los n individuos se representarán por un vector z_1 , dado por:

$$z_1 = Xa_1$$

Como las variables originales tienen media cero, también z_1 tendrá media nula. Su varianza será:

$$\frac{1}{n} z_1^T z_1 = \frac{1}{n} a_1^T X^T X a_1 = a_1^T S a_1$$

Donde S es la matriz de varianzas y covarianzas de las observaciones.

Como se puede observar, es posible maximizar la varianza sin límite aumentando el módulo del vector a_1 . Para que la maximización de la ecuación anterior tenga solución debemos imponer una restricción al módulo de a_1 , y, sin pérdida de generalidad, impondremos que $a_1^T a_1 = 1$. Introduciremos esta restricción mediante el multiplicador de Lagrange:

$$M = a_1^T S a_1 - \lambda(a_1^T a_1 - 1)$$

Y maximizaremos esta expresión de la forma habitual derivando respecto a los componentes de a_1 e igualando a cero. Entonces:

$$\frac{\partial M}{\partial a_1} = 2S a_1 - 2\lambda a_1 = 0$$

Cuya solución es:

$$S a_1 = \lambda a_1$$

Que implica que a_1 es un vector propio de la matriz S , y λ es la varianza de z_1 . Como ésta es la cantidad que queremos maximizar, λ será el mayor valor propio de la matriz S . Su vector asociado, a_1 , define los coeficientes de cada variable en la primera componente principal.

Cálculo del segundo componente

Vamos a obtener el mejor plano de proyección de las variables X . Lo calcularemos estableciendo como función objetivo que la suma de las varianzas de $z_1 = Xa_1$ y $z_2 = Xa_2$ sea máxima, donde a_1 y a_2 son los vectores que definen el plano. La función objetivo será:

$$\phi = a_1^T S a_1 + a_2^T S a_2 - \lambda_1(a_1^T a_1 - 1) - \lambda_2(a_2^T a_2 - 1)$$

Que incorpora las restricciones de que las direcciones deben de tener módulo unitario $a_i^T a_i = 1$, $i = 1, 2$. Derivando e igualando a cero:

$$\frac{\partial \phi}{\partial a_1} = 2S a_1 - 2\lambda_1 a_1 = 0$$

$$\frac{\partial \phi}{\partial a_2} = 2S a_2 - 2\lambda_2 a_2 = 0$$

La solución de este sistema es:

$$Sa_1 = \lambda_1 a_1$$

$$Sa_2 = \lambda_2 a_2$$

Que indican que a_1 y a_2 deben ser vectores propios de S . Tomando los vectores propios de módulo unitario se obtiene que, en el máximo, la función objetivo es:

$$\phi = \lambda_1 + \lambda_2$$

Es claro que λ_1 y λ_2 deben ser los dos auto-valores mayores de la matriz S , y a_1 y a_2 sus correspondientes auto-vectores. Observemos que la covarianza entre z_1 y z_2 , dada por $a_1^T S a_2$ es cero ya que $a_1^T a_2 = 0$, y las variables z_1 y z_2 estarán no correlacionadas. Si en lugar de maximizar la suma de las varianzas, que es la traza de la matriz de covarianzas de la proyección, se maximiza la varianza generalizada (el determinante de la matriz de covarianzas) se obtiene el mismo resultado.

Generalización

Puede demostrarse análogamente que el espacio de dimensión r que mejor representa a los puntos viene definido por los vectores propios asociados a los r mayores valores propios de S . Estas direcciones se denominan direcciones principales de los datos y, a las nuevas variables por ellas definidas, componentes principales. En general, la matriz X (y por lo tanto S) tiene rango p , existiendo entonces tantas componentes principales como variables, que se obtendrán calculando los valores propios o raíces características, $\lambda_1, \dots, \lambda_p$, de la matriz de varianzas y covarianzas de las variables S mediante:

$$|S - \lambda I| = 0$$

Y sus vectores asociados son:

$$(S - \lambda_i I) a_i = 0$$

Los términos λ_i son reales, al ser la matriz S simétrica, y positivos, ya que S es definida positiva. Por ser S simétrica si λ_j y λ_h son dos raíces distintas sus vectores asociados son ortogonales. Si S fuese semi-definida positiva de rango $r < p$, lo que ocurriría si $p - r$ variables fuesen combinación lineal de las demás, es que habría solamente r raíces características positivas y el resto serían ceros.

Llamando Z a la matriz cuyas columnas son los valores de las p componentes en los n individuos, estas nuevas variables están relacionadas con las originales mediante:

$$Z = XA$$

Donde $A^T A = I$. Calcular las componentes principales equivale a aplicar una transformación ortogonal A a las variables X (ejes originales) para obtener unas nuevas variables Z no correlacionadas entre sí. Esta operación puede interpretarse como elegir unos nuevos ejes coordenados, que coincidan con los "ejes naturales" de los datos.

Propiedades de los componentes principales

Los componentes principales son nuevas variables con las propiedades siguientes:

1. Conservar la variabilidad inicial: la suma de las varianzas de los componentes es igual a la suma de las varianzas de las variables originales, y la varianza generalizada de los componentes es igual a la original.
2. La proporción de variabilidad explicada por una componente es el cociente entre su varianza, el valor propio asociado al vector propio que lo define, y la suma de los valores propios de la matriz. En efecto, la varianza de la componente es λ_h , y la suma de las varianzas de las variables originales es $\sum_{i=1}^p \lambda_i$, igual a la suma de las varianzas de las componentes. La proporción de variabilidad total explicada por la componente h es $\lambda_h / \sum \lambda_i$.
3. Las covarianzas entre cada componente principal y las variables X vienen dadas por el producto de las coordenadas del vector propio que define el componente por su valor propio:

$$Cov(z_1; x_1, \dots, x_p) = \lambda_i a_i = (\lambda_i a_{i1}, \dots, \lambda_i a_{ip})$$

Donde a_i es el vector de coeficientes del componente z_i .

4. La correlación entre una componente principal y una variable X es proporcional al coeficiente de esa variable en la definición de la componente, y el coeficiente de proporcionalidad es el cociente entre la desviación típica de la componente y la desviación típica de la variable.
5. Las r componentes principales ($r < p$) proporcionan la predicción lineal óptima con r variables del conjunto de variables X .
6. Si estandarizamos las componentes principales, dividiendo cada una por su desviación típica, se obtiene la estandarización multi-variante de los datos originales.

Estandarizando las componentes Z por sus desviaciones típicas, se obtienen las nuevas variables:

$$Y_c = ZD^{-1/2} = XAD^{-1/2}$$

Donde $D^{-1/2}$ es la matriz que contienen las inversas de las desviaciones típicas de las componentes. La estandarización multi-variante de una matriz de variables X de media cero se define como:

$$Y_s = XAD^{-1/2}A^T$$

Tanto las variables Y_c como las Y_s tienen matriz de covarianzas identidad, pero unas pueden ser una rotación de las otras. Esto no altera sus propiedades, y la estandarización multi-variante puede interpretarse como:

- Obtener los componentes principales

- Estandarizarlos para que tengan todos la misma varianza.

Esta relación se presenta gráficamente a continuación. La transformación mediante componentes principales conduce a variables no correlacionadas pero con distinta varianza. Puede interpretarse como rotar los ejes de la elipse que definen los puntos para que coincidan con sus ejes naturales. La estandarización multi-variante produce variables no correlacionadas con varianza unidad, lo que supone buscar los ejes naturales y luego estandarizarlos. En consecuencia, si estandarizamos las componentes se obtienen las variables estandarizadas de forma multi-variante.

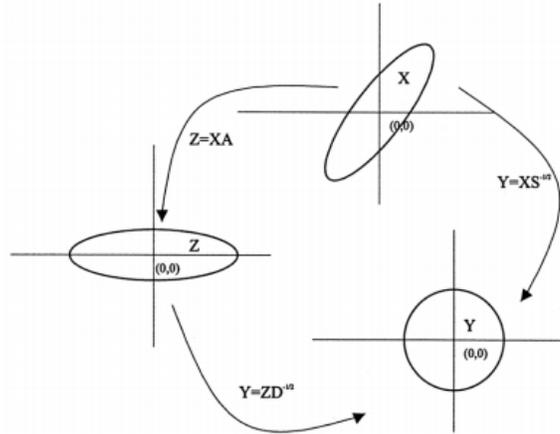


Ilustración 18 -- Representación gráfica de la relación entre componentes principales y estandarización multi-variante.²²

Selección del número de componentes principales

Se han sugerido distintas reglas para seleccionar el número de componentes:

1. Realizar un gráfico de λ_i frente a i . Comenzar seleccionando componentes hasta que los restantes tengan aproximadamente el mismo valor de λ_i . La idea es buscar un “codo” en el gráfico, es decir, un punto a partir del cual los valores propios son aproximadamente iguales. El criterio es quedarse con un número de componentes que excluyan los asociados a valores pequeños y aproximadamente del mismo tamaño.
2. Seleccionar componentes hasta cubrir una proporción determinada de varianza, como 80% o 90%. Esta regla es arbitraria y debe aplicarse con cierto cuidado.
3. Desechar aquellos componentes asociados a valores propios inferiores a una cota, que suele fijarse como la varianza media.

²² Imagen obtenida de McGraw-Hill Interamericana de España, SL.
www.mhe.es/universidad/ciencias_matematicas/pena/home/CAPITULO.PDF

Clasificación

SVM: Máquinas de Vectores de Soporte

Las Máquinas de Vectores de Soporte (SVM) fueron derivadas de la teoría de aprendizaje estadístico postulada por Vapnik y Chervonenkis.[8] Las SVM fueron presentadas en 1992 y adquirieron fama cuando dieron resultados muy superiores a las redes neuronales en el reconocimiento de letra manuscrita, usando como entrada píxeles. SVM está ganando gran popularidad como herramienta para la identificación de sistemas no lineales debido a que está basado en el principio de minimización de riesgo estructural (SRM), el cual ha demostrado ser superior al principio de minimización del riesgo empírico (ERM) utilizado por las redes neuronales.

Algunas de las razones por las que este método ha tenido éxito es que no posee mínimos locales y el modelo sólo depende de los datos que contienen más información, llamados vectores de soporte (SV). Las grandes ventajas que tiene SVM son:

- Una excelente capacidad de generalización debido a la minimización del riesgo estructurado.
- Existen pocos parámetros a ajustar, el modelo sólo depende de los datos con mayor información.
- La estimación de los parámetros se realiza a través de la optimización de una función de costo convexa, lo cual evita la existencia de un mínimo local.
- La solución de SVM es “sparse”, es decir, que la mayoría de las variables son cero en la solución de SVM, lo cual significa que el modelo final puede ser escrito como una combinación de números muy pequeños de vectores de entrada, llamados vectores de soporte.
- Lo anterior implica que la complejidad del clasificador depende de la cantidad de vectores que determinan la frontera y no de la dimensión del espacio.

SVM resuelve un problema cuadrático donde el número de coeficientes es igual al número de entradas o datos de entrenamiento. Este hecho hace que para grandes cantidades de datos, los métodos numéricos de optimización, existentes para resolver el problema cuadrático, no sean admisibles en términos computacionales. Éste es un problema que impide el uso de SVM para la identificación de sistemas no lineales en línea, esto es, en casos en los que las entradas son obtenidas de manera secuencial y el aprendizaje se realiza en cada paso.

Clasificación por hiperplanos

Supongamos que hay m observaciones y cada una consiste en un par de datos:

- Un vector $x_i \in R^n$, $i = 1, \dots, m$
- Una etiqueta $y_i \in \{+1, -1\}$.²³

²³ La etiqueta indica a que clase pertenece la muestra.

Supóngase que se tiene un hiperplano que separa las muestras positivas (+1) de las negativas (-1). Los puntos x_i que están en el hiperplano satisfacen $w x + b = 0$. Donde w es normal al hiperplano, siendo $\frac{|b|}{\|w\|}$ la distancia perpendicular del plano al origen. Lo que se quiere es definir dos hiperplanos que separen las muestras según sus etiquetas y_i , de forma que:

$$\begin{aligned} w x_i + b &= +1 & \text{para } y_i = +1 \\ w x_i + b &= -1 & \text{para } y_i = -1 \end{aligned}$$

O lo que es lo mismo

$$y_i(w x_i + b) = +1$$

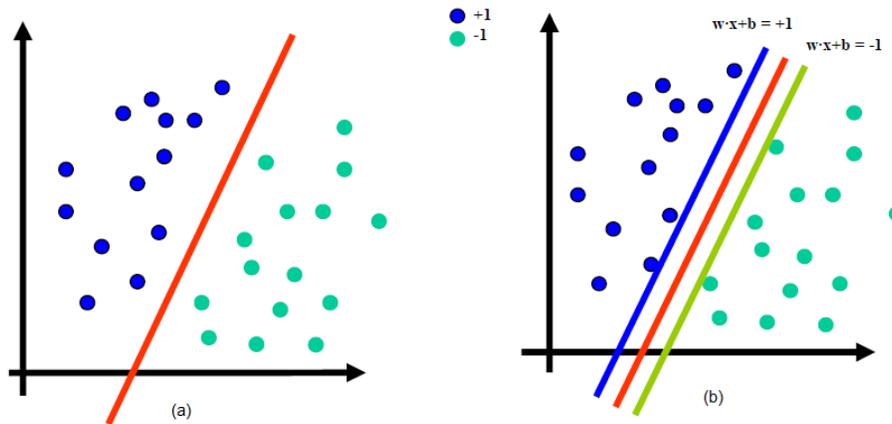


Ilustración 19 – (a) Hiperplano que separa las dos clases; (b) Hiperplanos de mayor margen negativo y positivo.²⁴

Sea d_+ (d_-) la distancia más corta entre el hiperplano positivo (negativo) y el punto positivo (negativo) más cercano. Definimos como “margen” a la distancia entre los hiperplanos “positivo” y “negativo”. El margen es igual a: $\frac{2}{\|w\|}$

²⁴ Imagen obtenida de A. Reyna. SVM de una clase: aplicación a detección de novedad. IIE, FING, 2009.

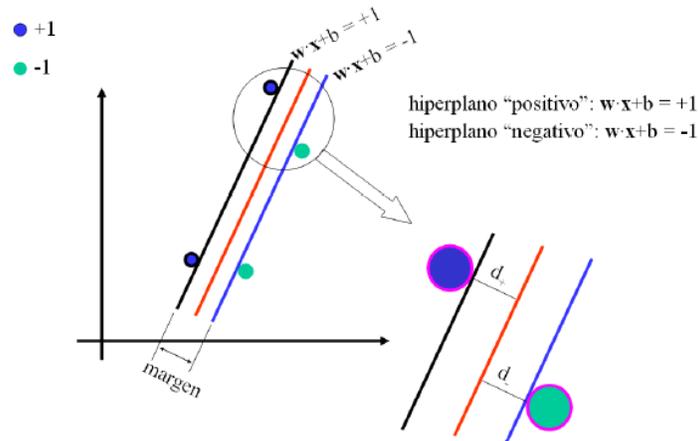


Ilustración 20 – Hiperplanos y margen.²⁵

La idea es encontrar un hiperplano con el máximo margen. Esto es un problema de optimización: maximizar $\frac{2}{\|w\|}$ condicionado a $y_i(wx_i + b) = +1$. Lo cual se puede expresar como maximizar $\|w\|^2$. Se introducen los multiplicadores de Lagrange para que todas las restricciones se agrupen en una única ecuación:

$$L_p = \frac{1}{2} \|w\|^2 - \sum_{i=1}^{m \cdot l} \alpha_i y_i (wx_i + b) + \sum_{i=1}^m \alpha_i$$

Haciendo que los gradientes de L_p respecto a w y n sean cero, se obtienen las siguientes condiciones:

$$w = \sum_{i=1}^m \alpha_i y_i x_i \quad ; \quad \sum_{i=1}^m \alpha_i y_i = 0$$

Lo que sustituyendo en L_p nos da el llamado problema dual:

$$L_D = \sum_{i=1}^m \alpha_i + \sum_{i=1, j=1}^m \alpha_i \alpha_j y_i y_j x_i x_j$$

El problema de optimización queda entonces dado por minimizar L_D sujeto a:

$$w = \sum_{i=1}^m \alpha_i y_i x_i \quad ; \quad \sum_{i=1}^m \alpha_i y_i = 0$$

Cuando los datos no se pueden separar linealmente, se hace un cambio de espacio mediante una función ϕ que transforme los datos de manera que se puedan separar linealmente en el nuevo espacio. Este nuevo espacio también tiene definido un producto interno lo que permite, utilizando

²⁵ Imagen obtenida de A. Reyna. SVM de una clase: aplicación a detección de novedad. IIE, FING, 2009.

el llamado “kernel trick”, calcular el producto interno de la imagen de las muestras en el nuevo espacio de características utilizando funciones núcleo, sin tener que explicitar el mapeo que hace ϕ .

SVM aplicado a problemas de clasificación de una clase

El algoritmo SVM es en general aplicado como un algoritmo de dos clases. Sin embargo, se presentan modificaciones a este algoritmo que permiten su aplicación a problemas donde sólo se busca establecer si una muestra pertenece o no a la clase con la cual se ha entrenado. Estos métodos son usualmente conocidos como SVM de una clase (oc-SVM).

Al igual que en SVM multi-clase, la idea es transformar el espacio de características a través de un kernel o núcleo. La diferencia en oc-SVM es que se trata al origen como el único miembro de la segunda clase, ilustración 21. Luego utilizando parámetros de relajación se separa la imagen de la clase que nos interesa estudiar del origen utilizándose las técnicas estándar de SVM para dos clases. Supongamos por ejemplo que se tiene una distribución P en el espacio de características. Se busca un subconjunto S simple dentro de dicho espacio de características, de forma que la probabilidad de que un punto de test regido por P caiga fuera del subconjunto S , está acotada por un cierto valor prefijado de antemano. En otras palabras, de todos los puntos regidos por P , no más de un cierto número puede caer fuera de dicha región S .

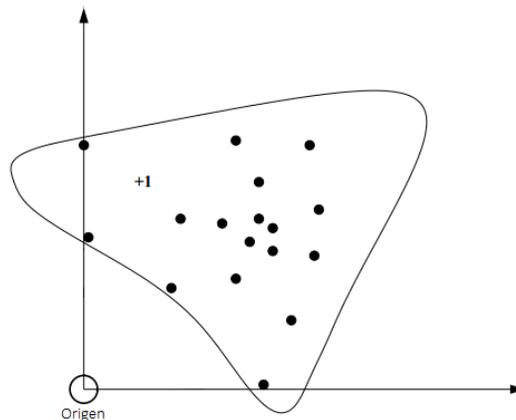


Ilustración 21 – SVM de una clase.²⁶

Llamemos $\nu \in (0,1]$ a dicho parámetro que acota la cantidad de puntos por fuera de S . La solución a este problema se obtiene estimando una función f positiva para los puntos dentro de S y negativa en el complemento de S . Es decir, se define f de forma de que valga $+1$ en una región “pequeña” que contenga a la mayoría de los vectores de datos y que valga -1 en el resto del espacio de características.

$$f(x) = \begin{cases} +1 & \text{si } x \in S \\ -1 & \text{si } x \in \bar{S} \end{cases}$$

²⁶ Imagen obtenida de L. M. Manevitz. One-Class SVMs for Document Classification. Department of Computer Science, University of Haifa. Israel, 2001.

Teniendo en cuenta esto, asumamos que $x_1, \dots, x_m \in X$ son muestras de entrenamiento pertenecientes a una clase X , siendo x un subconjunto compacto de R^n .

Sea $\phi: X \rightarrow H$ un mapeo que transforma dichas muestras en un espacio de características H con producto interno (espacio de Hilbert), de forma que el producto interno de la imagen de ϕ puede ser computado evaluando alguna función núcleo K , en forma simple en el nuevo espacio:

$$K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle = \phi(x_i)^T \phi(x_j)$$

Si bien se han planteado diversos núcleos, los más comúnmente usados son:

- Lineal:

$$K(x_i, x_j) = x_i^T x_j$$

- Polinómico:

$$K(x_i, x_j) = (\gamma x_i^T x_j + r)^d \quad \text{para } \gamma > 0$$

- Función de base radial (RBF):

$$K(x_i, x_j) = e^{-\gamma \|x_i - x_j\|^2} \quad \text{para } \gamma > 0$$

- Sigmoideo:

$$K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$$

Donde γ, r y d son los parámetros de los núcleos mencionados.

Una vez elegido el núcleo adecuado, la estrategia es mapear los datos en el espacio de características correspondiente al núcleo y separarlos del origen con el mayor margen posible. Para un nuevo x , el valor $f(x)$ queda determinado evaluando de qué lado del hiperplano cae en el nuevo espacio de características. Debido a la variedad de núcleos posibles, este simple planteo geométrico se corresponde a una igual variedad de estimadores no lineales en el espacio de partida.

Para separar entonces dichas muestras del origen, se requiere resolver el siguiente problema cuadrático:

$$\min_{w \in H, \rho \in R, \xi \in R^m} \left(\frac{1}{2} \|w\|^2 + \frac{1}{vm} \sum_{i=1}^m \xi_i - \rho \right)$$

Restringido por:

$$\langle w\phi(x_i) \rangle \geq \rho - \xi_i \quad ; \quad i = 1, 2, \dots, m \quad ; \quad \xi_i \geq 0$$

El parámetro $v \in (0, 1]$ acota la cantidad de muestras que se deja fuera de la región S del espacio de partida. Dado que las variables de relajación ξ_i son penalizadas en la función objetivo, podemos esperar que si w y ρ resuelven el problema, tenemos que la función de decisión es:

$$f(x) = \text{signo}(\langle w\phi(x) \rangle - \rho)$$

Ésta será positiva, +1, para la mayoría de las muestras x_i pertenecientes al conjunto de entrenamiento.

En la ilustración 22 se muestra un ejemplo simple en donde (a) muestra el espacio de partida con dos diferentes soluciones controladas por ν , y (b) el mapeo en el nuevo espacio del caso (a.2), donde el hiperplano separa todas las muestras menos a una.

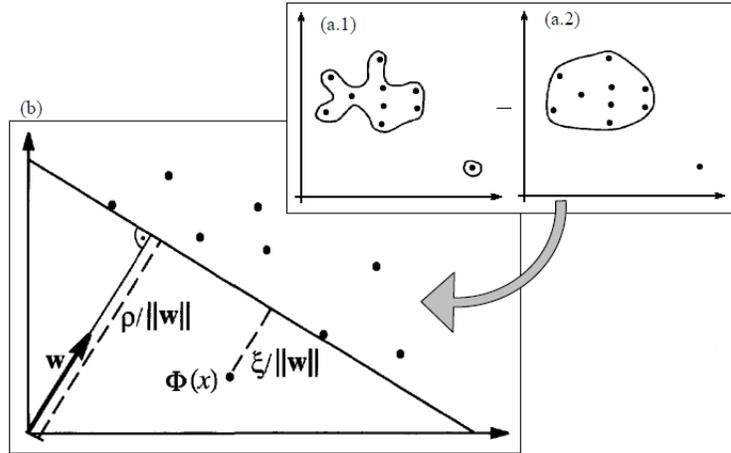


Ilustración 22 – Diferentes soluciones controladas por el parámetro ν .²⁷

Utilizando multiplicadores de Lagrange $\alpha_i, \beta_i \geq 0$ el lagrangeano que nos permite llegar a la solución con las restricciones planteadas es:

$$L(w, \xi, \rho, \alpha, \beta) = \frac{1}{2} \|w\|^2 + \frac{1}{\nu m} \sum_{i=1}^m \xi_i - \rho - \sum_{i=1}^m \alpha_i (\langle w, \phi(x_i) \rangle - \rho + \xi_i) - \sum_{i=1}^m \beta_i \xi_i$$

Igualando a cero las derivadas respecto a las variables primarias w, ξ y ρ , nos queda:

$$w = \sum_{i=1}^m \alpha_i \phi(x_i) \quad ; \quad \alpha_i = \frac{1}{\nu m} - \beta_i \leq \frac{1}{\nu m} \quad ; \quad \sum_{i=1}^m \alpha_i = 1$$

Lo que junto con lo visto para $f(x)$, nos da:

$$f(x) = \text{signo} \left(\sum_{i=1}^m \alpha_i K(x_i, x) - \rho \right)$$

Sustituyendo en el lagrangeano, obtenemos el problema cuadrático dual:

$$\min_{\alpha \in R^n} \left(\frac{1}{2} \sum_{i,j} \alpha_i \alpha_j K(x_i, x_j) \right)$$

²⁷ Imagen obtenida de A. Reyna. SVM de una clase: aplicación a detección de novedad. IIE, FING, 2009.

Restringido por:

$$0 \leq \alpha_i = \frac{1}{vm} \quad ; \quad \sum_i \alpha_i = 1$$

Capítulo 2: Implementación

Para realizar la implementación del sistema propuesto, se utilizó el programa Matlab® en su versión R2013a junto a las librerías fastICA [9] y LIBSVM [10]. Se utilizó además, la base de datos de arritmias creada por el Massachusetts Institute of Technology y el Beth Israel Hospital [11] (MIT-BIH) que se encuentra disponible para su descarga y utilización en www.physionet.org [12]. La misma está formada por 48 fragmentos electrocardiográficos de dos canales, de media hora de duración, extraídos de estudios ambulatorios. Dichos ECG fueron realizados con equipos Holter analógicos, para luego ser digitalizados a 360 muestras por segundo con una resolución de 11 bits sobre un rango de 10 mV (en el intervalo de -5 a 5 mV). El total de los latidos existentes fueron clasificados por dos o más cardiólogos, que además incorporaron información complementaria en los estudios, tales como anotaciones de ritmo, calidad de la señal y otros comentarios.

Dado que las características de la señal electrocardiográfica dependen intrínsecamente de la derivación que se esté analizando, y a los fines de lograr una mayor uniformidad en el análisis realizado, se decidió trabajar únicamente con uno de los canales de cada estudio. Dado que la derivación más utilizada en la base de datos es la ML II (Mason-Likar II), fue este el canal escogido.

Detección de complejos QRS

Pre-procesamiento de la señal

Pan y Tompkins desarrollaron un algoritmo de detección en tiempo real basado en el análisis de la pendiente de la señal en la banda de interés. Es decir, dada la señal de un ECG, se filtra en una banda de frecuencias conveniente y se analiza la señal extraída del filtro. El detector que se va a desarrollar se basa en este algoritmo.[4]

La ilustración 23 muestra los distintos tipos de filtros involucrados en el análisis de la señal de ECG. Primero se diseñó un filtro pasa-banda desde un filtro entero que requiere solo coeficientes enteros. Como no es posible diseñar el filtro pasa-banda deseado con este enfoque, se reemplaza el mismo con uno en cascada de filtros pasa-bajo y pasa-alto. Este filtro aísla la energía predominante del complejo QRS centrada en 10Hz, atenuando las bajas frecuencias características de las ondas P y T y el corrimiento de línea de base, y también las altas frecuencias asociadas a ruido muscular e interferencia de línea (50Hz).

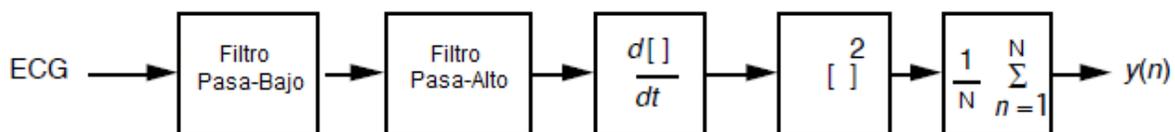


Ilustración 23 - Etapas de filtrado del detector de QRS.²⁸

²⁸ Imagen obtenida de W.J. Tompkins; V.X. Afonso. Biomedical Digital Signal Processing.

El paso siguiente es la diferenciación, una técnica estándar para encontrar las pendientes altas que normalmente distinguen al complejo QRS de las otras ondas. Hasta este punto en el algoritmo, todos los procesos son realizados con filtros digitales lineales.

Posteriormente sigue la transformación no lineal que consiste en elevar al cuadrado las muestras de la señal punto por punto. Esta transformación sirve para hacer todos los datos positivos antes de realizar la integración, y además para acentuar las pendientes altas en la señal obtenida del proceso de diferenciación. Estas pendientes son normalmente características del complejo QRS.

La señal pasa por un integrador de ventana móvil. Este integrador suma el área debajo de la señal derivada, luego de ser elevada al cuadrado, a través de un intervalo de 150 mSeg, avanza una muestra e integra la nueva ventana de 150ms. Se elige un ancho de ventana lo suficientemente grande para incluir la duración de complejos QRS anormales extendidos, pero lo suficientemente corto para no solapar el complejo QRS con la onda T.

Se aplican umbrales adaptativos a la señal integrada basados en una continua actualización de los picos estimados de señal y de ruido. Después de una detección, un proceso de toma de decisiones determina si el evento detectado es un complejo QRS o no.

Cada una de estas etapas se explicará a continuación. La ilustración 24 muestra una porción de registro de un ECG que servirá como ejemplo de señal de entrada.

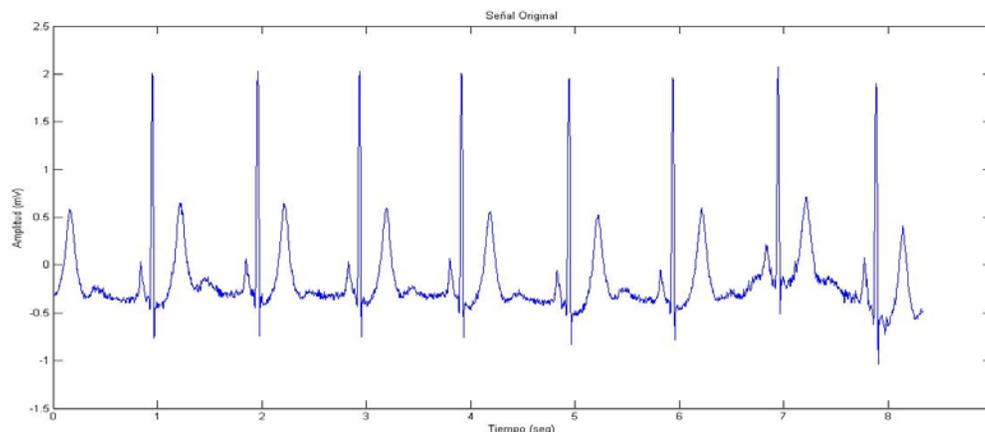


Ilustración 24 - Electrocardiograma muestreado a 360 muestras por segundo.²⁹

Filtro Pasa-Banda

El filtro pasa-banda reduce el ruido de una señal de ECG dejando pasar las frecuencias correspondientes al complejo QRS promedio que se encuentra en el rango de 5-15 Hz.

²⁹ Imagen obtenida del registro 106 de la base de datos de arritmias del MIT-BIH, muestras 1 al 3000.

Filtro Pasa-Bajo

La función de transferencia del filtro de segundo orden es

$$H(z) = \frac{(1 - z^{-6})^2}{(1 - z^{-1})^2}$$

Y su ecuación en diferencias es

$$y(nT) = 2y(nT - T) - y(nT - 2T) + x(nT) - 2x(nT - 6T) + x(nT - 12T)$$

La frecuencia de corte es aproximadamente 11 Hz, el retardo es de cinco muestras y la ganancia es 36. Para evitar la saturación, la salida se divide en 32, que es la potencia de dos más cercana a la ganancia para poder implementarla con instrucciones de cambio de posición de bits.

Este filtro tiene respuesta en fase lineal y logra atenuar las altas frecuencias en un valor de aproximadamente 25 dB. Además logra una atenuación de 27 dB para la frecuencia de 50 Hz. Esta última podría eliminarse completamente utilizando la siguiente función de transferencia,

$$H(z) = \frac{(1 - z^{-7})^2}{(1 - z^{-1})^2}$$

Pero la misma tiene una ganancia de 49 de modo que para evitar la saturación se requeriría dividir la salida por dicho valor, pero al no ser una potencia de dos no pueden utilizarse instrucciones de cambio de posición de bits. Como se pretende utilizar procesadores de bajos recursos, se les da prioridad a aquellos filtros que requieren menor costo computacional.

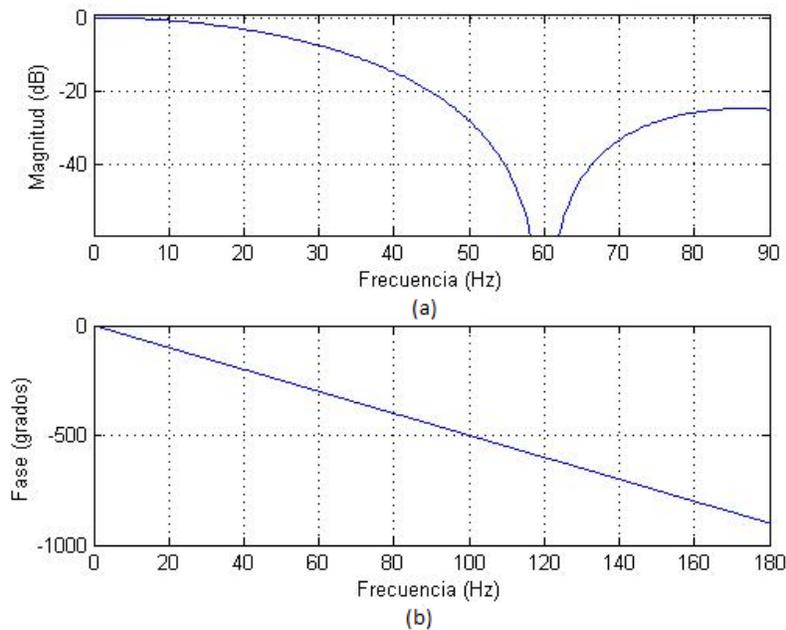


Ilustración 25 - Filtro Pasa-Bajo. a) Respuesta en magnitud. b) Respuesta en fase.

En la ilustración siguiente se puede observar el ECG de la ilustración 24 luego de ser procesada con el filtro pasa-bajo.

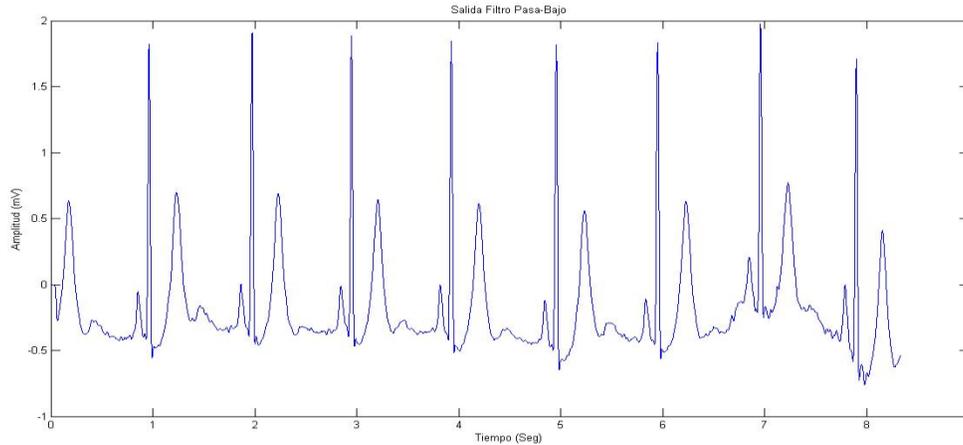


Ilustración 26 - Señal de ECG filtrada con filtro pasa-bajo.

Filtro Pasa-Alto

El filtro pasa-alto se implementó sustrayendo un filtro pasa-bajo de primer orden de un filtro pasatodo con retraso. La función de transferencia del filtro pasa-bajo es:

$$H_{lp}(z) = \frac{Y(z)}{X(z)} = \frac{1 - z^{-32}}{1 - z^{-1}}$$

Y la ecuación en diferencias:

$$y(nT) = y(nT - T) + x(nT) - x(nT - 32T)$$

Este filtro tiene una ganancia de 32 y un retraso de 15 muestras. El filtro pasa-alto se obtiene de dividir la salida del filtro pasa-bajo por la ganancia y después sustrayéndolo de la señal original. La función de transferencia es

$$H_{hp}(z) = \frac{P(z)}{X(z)} = z^{-15} - \frac{H_{lp}(z)}{32} = \frac{1 - 1 + 32z^{-15} - 32z^{-16} + z^{-32}}{1 - z^{-1}}$$

Y la ecuación en diferencias de este filtro es:

$$p(nT) = p(nT - T) - \frac{x(nT)}{32} + x(nT - 15T) - x(nT - 16T) + \frac{x(nT - 32T)}{32}$$

La frecuencia de corte del filtro es aproximadamente 8 Hz y la ganancia es uno. A continuación se muestra la respuesta del filtro.

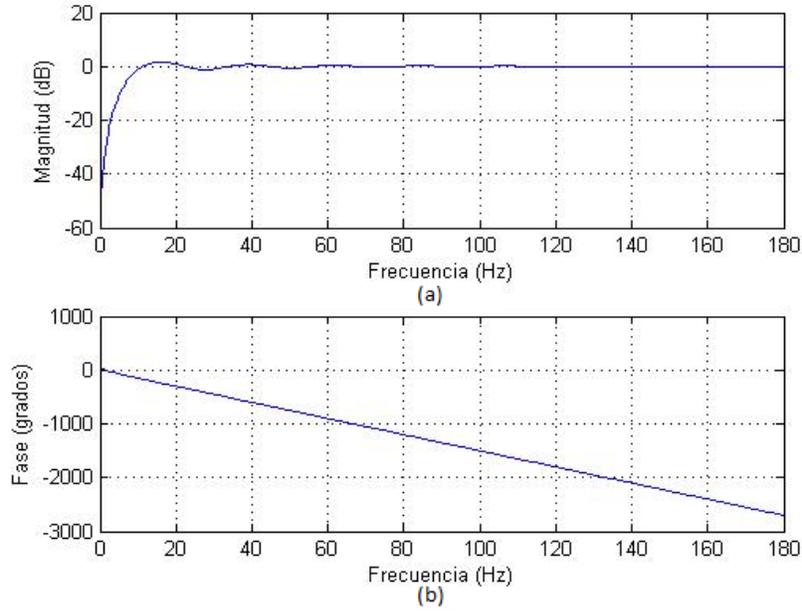


Ilustración 27 - Filtro Pasa-Alto. a) Respuesta en magnitud. b) Respuesta en fase.

Al igual que sucede con el filtro pasa-bajo, se pueden obtener mejores resultados utilizando la siguiente función de transferencia.

$$H_{hp}(z) = z^{-24} - \frac{1}{48} \frac{1 - z^{-48}}{1 - z^{-1}} = \frac{1}{48} \frac{-1 + 48z^{-24} - 48z^{-25} + z^{-48}}{1 - z^{-1}}$$

Pero nuevamente se deben realizar operaciones de división con números que no son potencia de dos. En la ilustración 28 se puede observar la respuesta del filtro pasa-banda resultante de la concatenación de los filtros anteriores. La frecuencia central del filtro es 10 Hz.

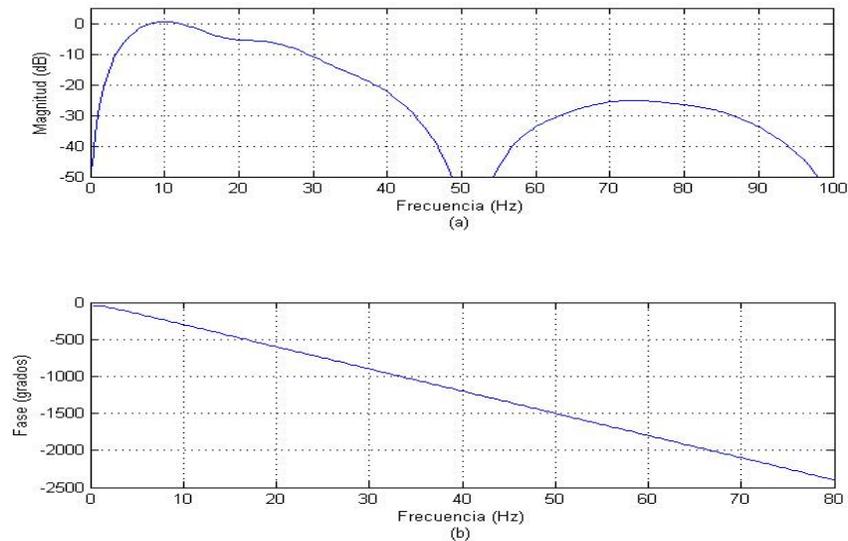


Ilustración 28 - Filtro Pasa-Banda. A) Respuesta en magnitud. b) Respuesta en fase.

La ilustración 29 es la señal resultante luego de que la señal de la ilustración 24 pase por el filtro pasa-banda. Obsérvese la atenuación de la onda T.

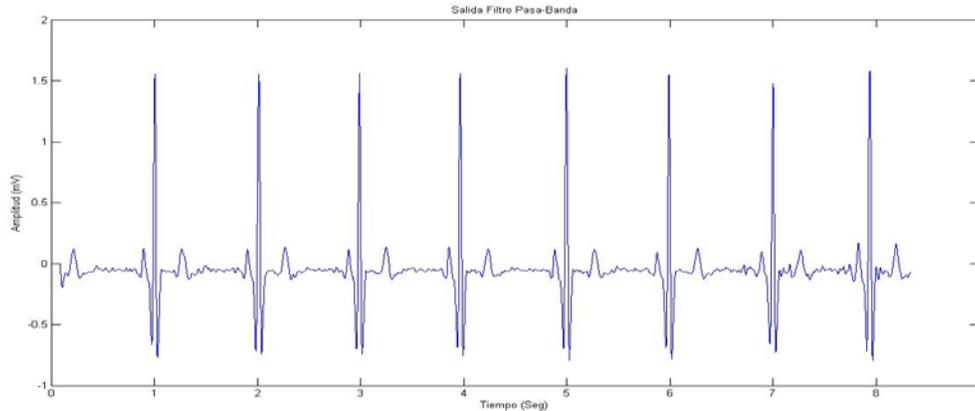


Ilustración 29 - Señal de ECG filtrada con filtro pasa-banda.

Derivación

Luego de que la señal ha sido filtrada, se deriva para obtener información sobre la pendiente del complejo QRS. Se utiliza un derivador de cinco puntos que viene dado por la siguiente función de transferencia.

$$H(z) = 0.1(2 + z^{-1} - z^{-3} - 2z^{-4})$$

Cuya ecuación en diferencias es:

$$y(nT) = \frac{2x(nT) + x(nT - T) - x(nT - 3T) - 2x(nT - 4T)}{8}$$

La fracción $1/8$ es una aproximación de la ganancia real que es 0.1 .

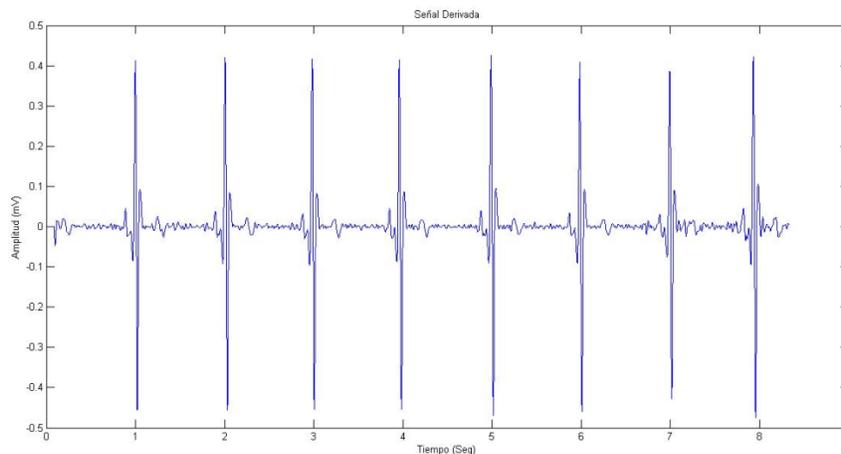


Ilustración 30 - Señal de ECG luego de la etapa de diferenciación.

Cuadrado de la señal

La ecuación que implementa esta operación es

$$y(nT) = [x(nT)]^2$$

Al elevar la señal al cuadrado hacemos positivos todos los puntos de la señal y damos mayor importancia a las frecuencias centrales de la banda pasante contra las que están en el borde y fuera de las bandas que son atenuadas.

Integración

La pendiente de la onda R por sí sola no garantiza la detección de complejos QRS. Varios complejos QRS anormales que tienen grandes amplitudes y largas duraciones (pendientes no muy pronunciadas) pueden no ser detectadas usando solamente la pendiente de la onda R. La integración con ventana móvil extrae otras características que dan más información sobre esta onda, y se implementa con la siguiente ecuación:

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

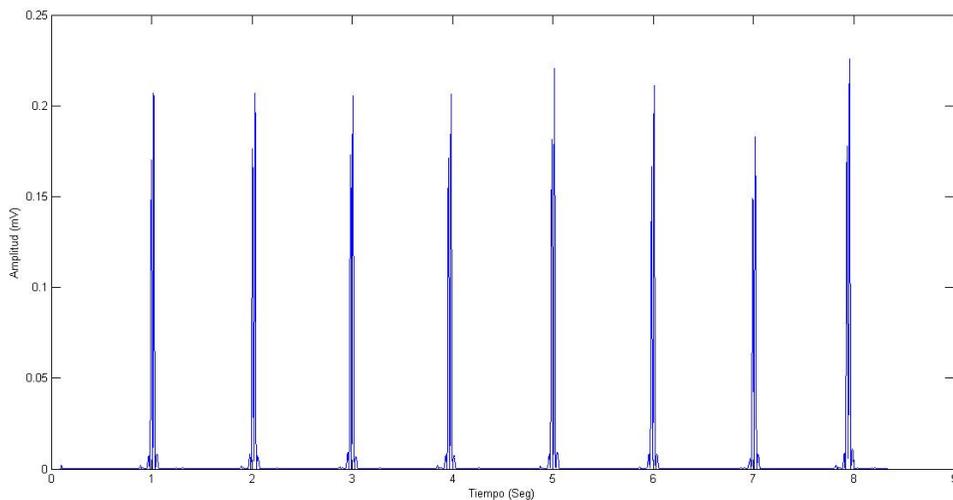


Ilustración 31 – Derivada de la señal de ECG luego de elevarla al cuadrado.

Donde N es el número de muestras del ancho de la ventana. Este valor debe elegirse cuidadosamente ya que, como se mencionó anteriormente, si la ventana es muy ancha puede fusionar el complejo QRS con la onda T; y si es muy corta, un complejo QRS puede producir varios picos. Este valor se ajusta experimentalmente, aunque se sabe que es del orden de los 150ms. Se tomó un $N=64$ (177ms para una frecuencia de muestreo de 360 Hz).

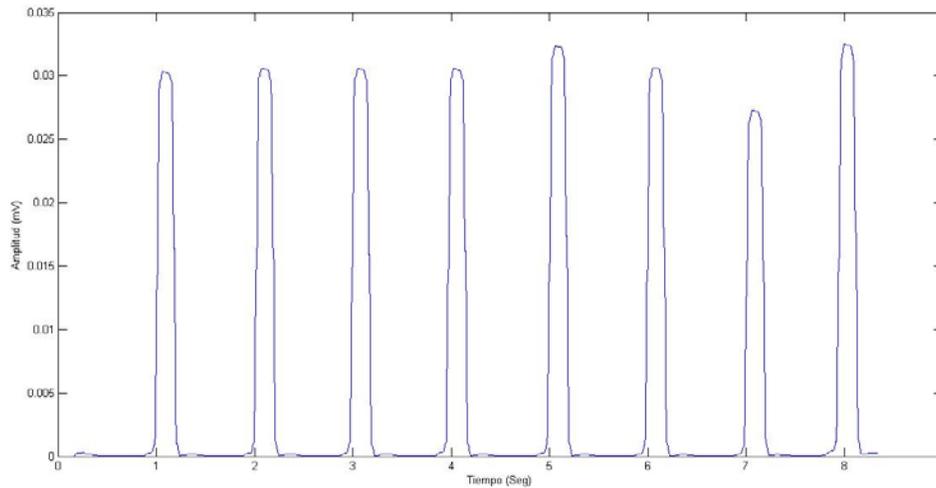


Ilustración 32 - Señal de ECG luego de la etapa de integración.

Algoritmo de Detección

Una vez procesada la señal, se debe determinar donde se ubica cada complejo QRS para las etapas de segmentación y caracterización. Para ello se utilizan umbrales adaptativos, según los cuales se determina si un pico de energía corresponde a un complejo QRS o si se debe considerar como ruido. También se implementan en el algoritmo técnicas de búsqueda hacia atrás y determinación del pico R.

La ilustración 33 muestra el algoritmo con un diagrama de flujo. En la misma se aprecia el bucle principal que recorre la señal obtenida con el procesamiento. Después de cada detección exitosa, se busca el nivel más alto de la señal (pico) hasta que la señal cae a la mitad del máximo nivel encontrado hasta el momento (ilustración 34). Cuando sucede, se determina que se ha encontrado un posible complejo QRS y se pasa al siguiente nivel donde se determina si el pico de energía encontrado corresponde efectivamente a un complejo o si se trata solamente de ruido.

Para eso se utilizan umbrales adaptativos. Cuando el nivel de pico supera el umbral, se determina que se encontró un complejo QRS y se busca el pico R en la señal original. Cuando se termina de grabar el complejo, se actualiza el umbral y se recorre la señal procesada buscando donde comienza a subir nuevamente para evitar que el algoritmo analice una y otra vez el mismo pico. A este punto se le refiere como “comienzo de loma”. Además, cuando se encuentra un complejo QRS se aplica un salto que corresponde al tiempo refractario en el cual es fisiológicamente imposible que se encuentre otro complejo.

Finalmente se mide el tiempo que ha pasado desde la última detección y en caso de ser excesivo se realiza una búsqueda hacia atrás.

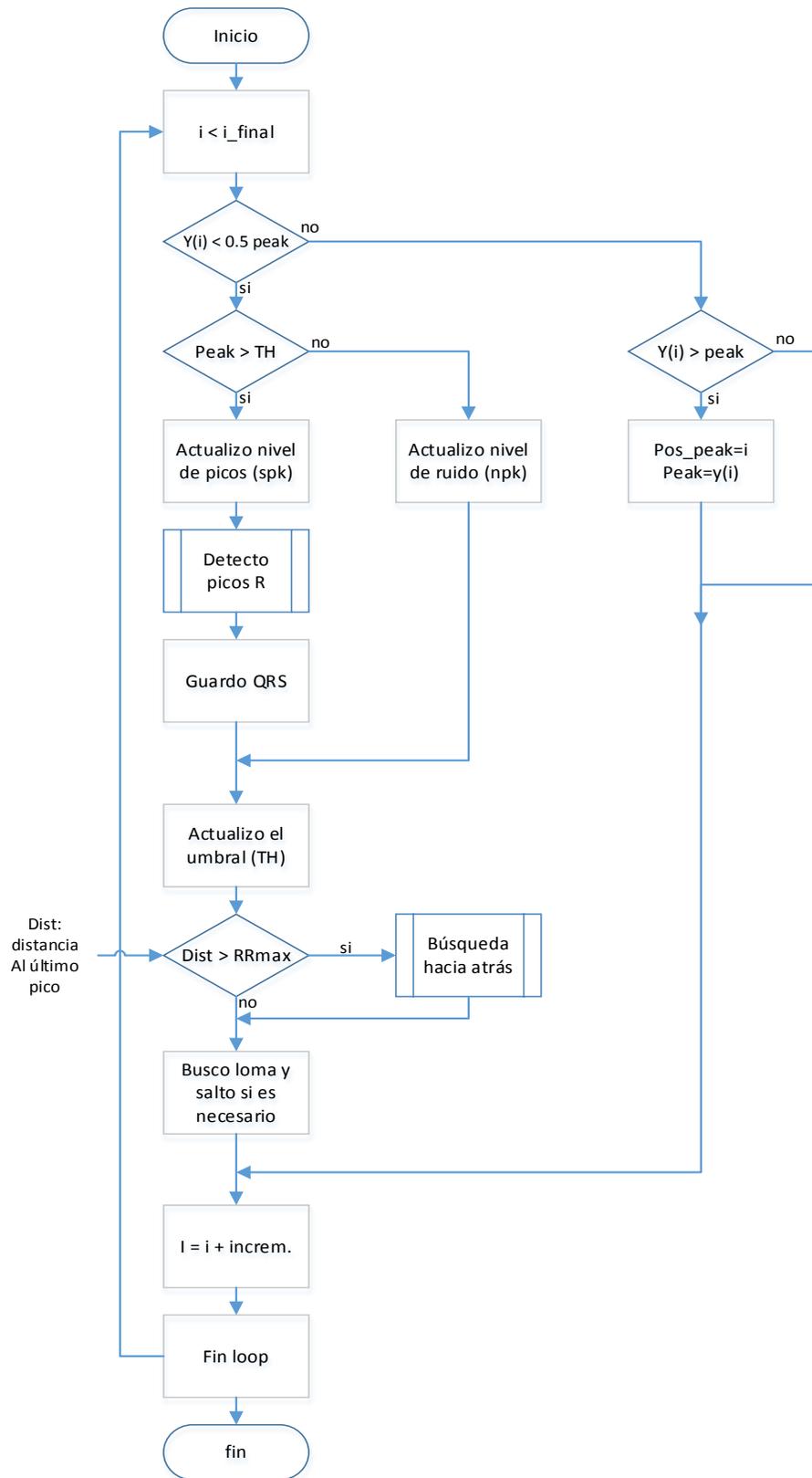


Ilustración 33 - Algoritmo de detección de complejos QRS.

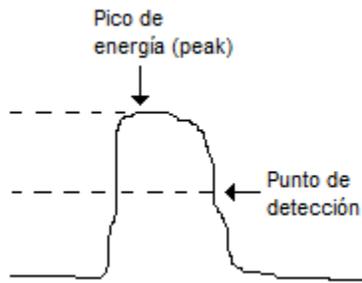


Ilustración 34 - Detección de pico de señal.

Umbrales

En los umbrales se basa la decisión de determinar cuáles picos de la señal corresponden a complejos QRS y cuales a ruido. Un umbral fijo a lo largo de toda la detección en este tipo de señales es una opción sumamente deficiente, por lo que la utilización de umbrales adaptativos es imprescindible.

En este algoritmo se utiliza un umbral que se actualiza después de cada pico detectado. Para ello se lleva un registro del nivel de picos que corresponden a señal (SPK) y otro que corresponde a los de ruido (NPK). Estos registros se realizan con un promedio ponderado del nivel del último pico y el anterior registro. Así, se le va dando cada vez menos importancia en el promedio a los niveles más alejados del pico actual, y son los niveles más cercanos los que dominan el promedio. El parámetro w_{pk} permite ajustar cuanto afecta cada nuevo pico incluido en el promedio. Su valor se obtiene experimentalmente y es igual a 0.125.

$$SPK_i = w_{pk} \cdot peak + (1 - w_{pk}) \cdot SPK_{i-1}$$

$$NPK_i = w_{pk} \cdot peak + (1 - w_{pk}) \cdot NPK_{i-1}$$

Cada uno de estos dos niveles es actualizado cada vez que se determina que un pico (cuyo valor está dado en la variable $peak$) corresponde a un complejo (SPK) o a ruido (NPK).

A partir de ellos se calcula el umbral con la ecuación:

$$TH_1 = NPK + w_{th} \cdot (SPK - NPK)$$

Así, w_{th} determina a que altura entre los dos niveles se fija el umbral. Afonso [4] sugiere un valor de aproximadamente 0.25. Se lograron mejores resultados utilizando $w_{th} = 0.21$. Este umbral es utilizado durante el funcionamiento normal, sin embargo cuando se pasa al modo “búsqueda hacia atrás” se utiliza como umbral de detección la mitad del valor de TH_1 :

$$TH_2 = \frac{TH_1}{2}$$

Búsqueda hacia atrás

Consiste en la búsqueda hacia atrás de un pico R cuando no ha sido detectado durante un lapso de tiempo luego del pico R anterior. Para implementar esta técnica se mantienen dos promedios del intervalo R-R. El primero es el promedio de los últimos ocho intervalos R-R, es decir:

$$RR_{AV_1} = \frac{RR_n + \dots + RR_{n-7}}{8}$$

El segundo es el promedio de los últimos ocho intervalos R-R “normales”. Se le llama normal al intervalo R-R que cae dentro de los siguientes límites:

$$RR_{bajo} = 0.92.RR_{AV_1}$$

$$RR_{alto} = 1.16.RR_{AV_1}$$

Entonces la búsqueda hacia atrás se aplica cuando el intervalo R-R actual supera un valor límite, esto es:

$$RR_{actual} > 1.66.RR_{AV_2}$$

Una vez que el algoritmo entra en el estado de búsqueda hacia atrás, retrocede hasta el último pico seleccionado. A partir de ese punto comienza a buscar picos disminuyendo el umbral a la mitad (usando TH_2), y actualizando los umbrales usando un peso de $w_{pk} = 0.25$. Es efecto de este cambio es darle más importancia a los picos nuevos. El algoritmo se detiene cuando encuentra un pico R o cuando llega al punto donde entró en la búsqueda en cuyo caso decide que no hubo ningún latido. Al finalizar el algoritmo, cualquiera sea el resultado, el umbral y la ponderación de picos nuevos vuelven a sus valores normales.

Segmentación

Se optó por segmentar los latidos en ventanas centradas en el pico R de 201 muestras (100 muestras de cada lado). El motivo de la elección es que de este modo es posible obtener muestras de la misma longitud que poseen el complejo QRS en la misma posición, lo que sería imposible si se segmentara en intervalos RR. La segmentación en ventanas centradas facilita considerablemente las etapas posteriores del proyecto integrador, además de brindar un mayor desempeño.

El motivo de tomar 100 muestras de cada lado se debe a que de esta forma, el complejo QRS siempre queda incluido dentro del segmento y al mismo tiempo no se incluye la onda T. Se probó utilizar una ventana de mayor tamaño de manera que ésta sea incluida, pero los resultados obtenidos fueron peores que en el caso anterior.

Extracción de características

Luego de que un latido es detectado y segmentado, se requiere extraer de él un conjunto de características que den la información suficiente para poder categorizarlo. Se estudiaron posibles

técnicas para realizar la caracterización de los latidos, de las cuales se pre-seleccionaron las siguientes:

- Transformada wavelet u ondita.
- Caracterización morfológica.
- Análisis de Componentes Independientes.
- Enfoque no lineal y Teoría del caos.

La transformada ondita es una técnica que permite el análisis de señales en los dominios temporal y de frecuencia, con la gran ventaja, frente a la transformada de Fourier, de poder realizar un análisis multi-resolución, es decir, mejorar la resolución temporal para determinadas bandas de frecuencia y mejorar la resolución en frecuencia para otras. Esta técnica se encuentra ampliamente utilizada en etapas de pre-procesamiento y clasificación de señales biológicas, y sus resultados son muy buenos, logrando un porcentaje de aciertos en la clasificación de latidos superior al 90% en los peores casos [13]. Sin embargo presenta una gran desventaja que es su alto costo computacional, esto quiere decir que no es posible su aplicación en procesadores de bajos recursos, motivo por el cuál no se utilizó este método.

La caracterización morfológica trabaja en el dominio temporal utilizando como características, por ejemplo, las duraciones y amplitudes de las ondas P, R y T, y las duraciones de los intervalos RR de la señal de ECG. Si bien no requiere de muchos recursos computacionales, utilizar únicamente esta técnica resulta en porcentajes de aciertos muy bajos. Como en la etapa de pre-procesamiento y detección se optó por sacrificar levemente la forma de la señal para ganar en velocidad de procesamiento, se decidió no utilizar las características morfológicas para la clasificación de latidos. Sí se utilizarán las duraciones de los intervalos RR.

El Análisis de Componentes Independientes es una técnica que ha ganado popularidad en el campo del procesamiento de señales biomédicas. Se utiliza generalmente cuando se requiere separar señales que han sido obtenidas a través de varios canales, en las componentes que la constituyen. Debido a su extensa utilización en diversos campos como el procesamiento de imágenes digitales, economía y mercados financieros, se desarrollaron algoritmos denominados “fastICA” que brindan la posibilidad de utilizar este análisis con eficiencia ya que requieren bajos costos de procesamiento. Se optó por utilizar esta herramienta en conjunto con el Análisis de Componentes Principales, la cual se utiliza, como se explicó anteriormente, para reducir el número de características.

Durante los últimos años ha aumentado el número de análisis de señales electrocardiográficas con el enfoque no lineal. Se han propuesto diversas medidas no lineales para estimar la complejidad de una serie temporal. Algunas de ellas han sido aplicadas a las señales de ECG, como también a la variabilidad de la frecuencia cardíaca. Similar a la transformada ondita, si bien el enfoque no lineal presenta suficiente consistencia para la clasificación de señales, los altos costos computacionales requeridos para el cálculo de las características limita su aplicación a condiciones fuera de línea (off-line), es decir que no pueden aplicarse en tiempo real.[14]

Implementación del Análisis de Componentes Independientes

Se utilizó la librería FastICA desarrollada por Aapo Hyvarinen et al., la misma es gratuita (GPL) e implementa el algoritmo “rápido” de punto fijo [9]. Ésta cuenta con una interfaz gráfica que permite trabajar con ICA de manera simple.

Recordando el modelo ICA:

$$x = A \cdot s$$

Donde x representa las muestras, A es la matriz que contiene los pesos y s los componentes independientes. Inicialmente se planteó, una vez obtenida la matriz de mezcla A , sacar con la misma los componentes independientes de cada señal y compararlos entre ellos con parámetros estadísticos como la media, la desviación estándar, entre otros. Sin embargo lo anterior no tuvo resultados positivos ya que no presentaban diferencias significativas. Debido a ello, se pretende utilizar como características los pesos, de modo que se requiere tomar un conjunto de muestras representativo para encontrar los componentes independientes, para luego, con las nuevas muestras, obtener los pesos de la siguiente forma:

$$x \cdot s^{-1} = A$$

El primer planteo que se presenta es cuántas muestras se deben utilizar para encontrar los componentes independientes. Variando la cantidad de muestras, se observó que los registros analizados presentaban alguno de estos dos comportamientos, ilustración 35. En uno de ellos, comportamiento 1, la precisión obtenida no varía significativamente a medida que se aumenta la cantidad de muestras; mientras que en el otro, comportamiento 2, la precisión va aumentando a medida que aumenta la cantidad de muestras alcanzando una meseta a partir de las cien muestras. De modo que se optó por utilizar cien muestras para la obtención de los componentes independientes.

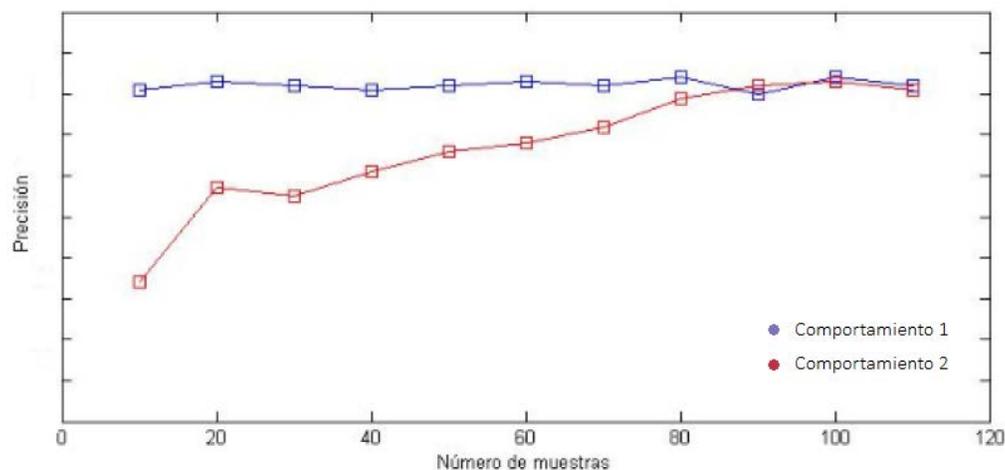


Ilustración 35 - Precisión en función del número de muestras para obtención de componentes independientes.

El siguiente paso es determinar el número de componentes a utilizar. Éste debe ser lo menor posible dado que, a medida que aumenta su número, también aumenta la cantidad de procesamiento necesario. Para lograr la reducción de componentes utilizamos el Análisis de Componentes Principales sobre las muestras de los latidos. Se utilizaron dos criterios de selección. En primer lugar se realizó un gráfico de los valores propios y se seleccionaron componentes hasta que los restantes tengan similar valor. Y en segundo lugar, que las componentes seleccionadas posean más del 95% de varianza. Una representación de los datos obtenidos se muestra a continuación.

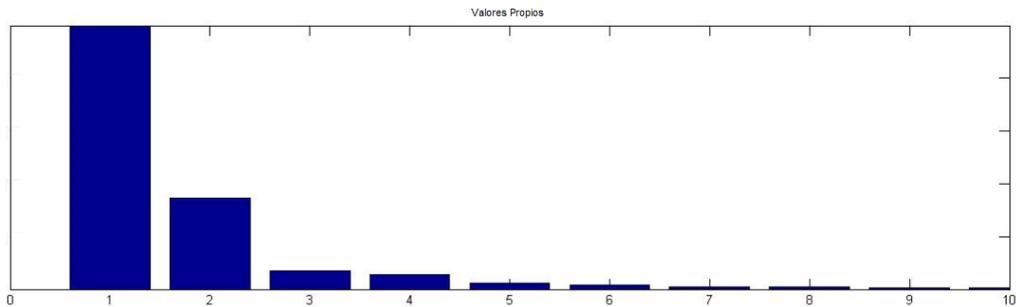


Ilustración 36 – Valores propios de las señales.

Este esquema se repite para todos los registros analizados. Variando la posición del “codo” en más o menos una componente. Si utilizamos cuatro componentes principales se logra cumplir con los dos requisitos obteniendo un porcentaje de varianza del 99%. De todos modos se observó, además, como varía la precisión en función de la cantidad de componentes empleados. Los registros analizados pudieron dividirse, según la cantidad de componentes necesarios para lograr la mayor precisión posible, en cuatro grupos, ilustración 37. Lo que difiere entre los grupos es la cantidad de ruido que posee la señal. El grupo 1 posee señales muy limpias; el grupo 2 posee señales con ruido leve; el grupo 3 posee señales con ruido moderado; y el grupo 4 posee señales con mucho ruido. Se concluyó entonces, que la cantidad de componentes necesarios para obtener la mayor precisión posible depende de la cantidad de ruido que posea la señal, es decir que mientras más limpia sea la señal menos componentes se necesitarán, y si se utilizan más componentes de lo necesario, la precisión empieza a disminuir.

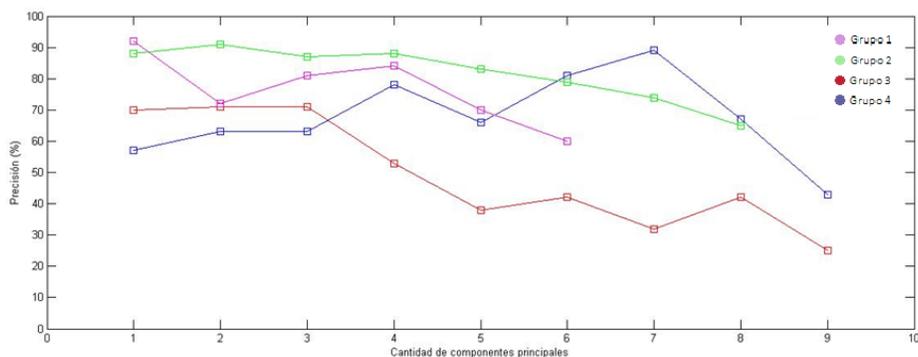


Ilustración 37 – Precisión vs Cantidad de componentes principales.

Guiándonos por el gráfico, se podría decir que conviene tomar entre uno y tres componentes principales dado que presentan el mayor porcentaje de precisión considerando los cuatro grupos. Sin embargo, al utilizar una de esas tres opciones se observó que mientras menor cantidad de componentes se emplean, mayor es el número de falsos negativos (latidos anormales que el programa detecta como normales). El objetivo es lograr un detector robusto a la detección de latidos anormales incluso a costa de la pérdida de precisión. Considerando esto, la mejor opción pasa a ser la utilización de cuatro componentes principales. De todos modos, se utilizaron cinco componentes ya que al analizar los datos obtenidos se notó que en todos los registros los latidos contiguos a aquellos que el detector no lograba detectar eran clasificados como anormales, lo que permite salvar el error provocado en la etapa anterior y de este modo aumenta considerablemente la sensibilidad hacia los latidos anormales. Tomando un número de componentes mayor a cinco sólo mejora la precisión para un número reducido de registros mientras que disminuye para el resto, por lo que se descartó esta opción.

Una vez seleccionado el número de componentes principales se realizó el Análisis de Componentes Independientes, obteniendo la matriz de mezcla y los componentes independientes. Las dimensiones de las matrices son:

$$x[100,201] = A[100,5].s[5,201]$$

Donde cada fila de x representa un latido de una longitud de 201 muestras (una ventana de 100 muestras a cada lado centrada en el pico R del complejo QRS), cada fila de s representa un componente independiente, y cada fila de la matriz de mezcla A contiene los pesos de cada componente independiente para cada señal de x .

Como se quiere despejar la matriz de mezcla, es necesario utilizar la matriz inversa de s , pero nótese que ésta no es cuadrada. Para solucionar este problema se utilizó la matriz conocida como “pseudo-inversa” definida como:

$$S^{-1} = S^T.(S.S^T)^{-1}$$

En algunas situaciones, esta simple aproximación da resultados satisfactorios pero en muchas otras se requieren estimaciones más sofisticadas [6]. En la ilustración 38 se muestra un latido obtenido con los componentes independientes y la matriz A originales, y con la utilización de la pseudo-inversa de s . Se puede observar que la estimación de la matriz inversa presenta buenos resultados en este caso particular.

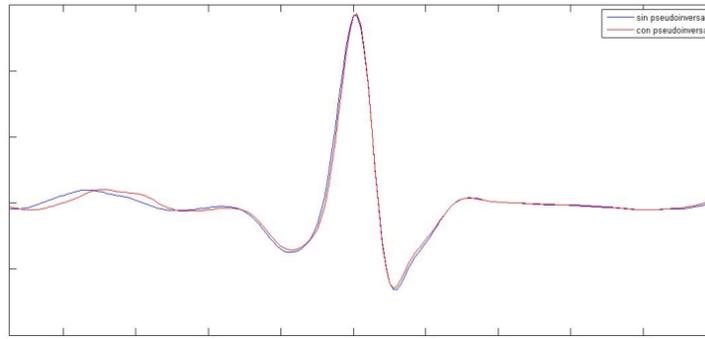


Ilustración 38 – Aplicación de la pseudo-inversa para el cálculo de los pesos.

Una vez obtenida la matriz S^{-1} , se procedió a obtener la media de los pesos de la matriz A obteniendo una nueva matriz A_m , de dimensión 1×5 . El motivo de esta matriz es que, una vez obtenidos los pesos de una determinada señal, se los divide por ella para trabajar con una relación de los pesos en vez de con los pesos mismos.

Resumiendo, los pasos para la obtención de los pesos son los siguientes:

1. Se toman cien segmentos, representando cada uno un latido, de la señal.
2. Se aplican PCA e ICA obteniendo las matrices A y s .
3. Se calcula la inversa de s a través de la pseudo-inversa.
4. Se calcula la media de cada columna de la matriz A obteniendo una nueva matriz A_m .
5. Para cada nuevo segmento de latido x_i se calculan los pesos de la forma:

$$x_i[1,201] \cdot s^{-1}[201,5] = A_i[1,5]$$

6. La matriz A_i se divide elemento a elemento por A_m :

$$A'_i = \begin{bmatrix} \frac{a_{i,1}}{a_{m,1}} & \frac{a_{i,2}}{a_{m,2}} & \dots & \frac{a_{i,5}}{a_{m,5}} \end{bmatrix}$$

7. Los elementos de la matriz A'_i son los que se utilizan para la clasificación.

Se probó como alternativa al paso 6 usar las diferencias entre los valores y las medias, $A_i - A_m$, y el cuadrado de esta diferencia, $(A_i - A_m)^2$, en vez de la división. Con estas variantes se logró incrementar hasta en un 15% la precisión pero no se salvaban todos los errores producidos por el detector, de modo que se le dio prioridad a la robustez respecto a los latidos anormales, dejando como relación la división de estas matrices, aunque esto implique una pérdida de precisión.

Clasificación de los latidos

Una vez que el conjunto de características se ha definido, se pueden construir modelos con esta información a través del uso de algoritmos de inteligencia artificial (IA) tales como el aprendizaje automático (machine learning) y la minería de datos.

Los cuatro algoritmos más populares empleados para esta tarea son: Máquinas de Vectores de Soporte (SVM), Redes Neuronales Artificiales (ANN), Discriminante Linear (LD) y Reservoir

Computing with Logistic Regression (RC) [13]. Es difícil inferir cuál de los anteriores es la mejor opción para utilizar. Esto se debe a dos causas principalmente:

- La primera es que en muchas investigaciones se utiliza lo que se conoce como “paradigma intra-paciente”, en donde se utilizan las mismas señales para la etapa de entrenamiento y prueba del algoritmo. Si bien obtienen muy buenos resultados, el escenario planteado no es muy realista. Las investigaciones que se hicieron con el “paradigma inter-paciente” tienen desempeños más bajos que las anteriores y no hay mucha diferencia entre los porcentajes de precisión en función de las técnicas de IA implementadas.
- La segunda causa se debe a la gran cantidad de formas de caracterizar la señal. Los investigadores utilizan, incluso para la misma técnica de clasificación, distintos criterios de selección de características, lo cual dificulta discernir si el error producido por el clasificador se debe al método de IA utilizado o a una mala selección del método para caracterizar la señal.

Se optó por trabajar con SVM debido a las ventajas mencionadas en el apartado teórico. Otro motivo de elección fue que SVM permite, a través del denominado SVM de una clase, entrenar el algoritmo sólo con un tipo de latido, es decir que permite personalizarlo para cada paciente en particular sin la necesidad de disponer de una gran base de datos, reemplazando la misma por un conjunto de latidos del paciente que se puede extraer en el momento. Esto es una gran ventaja debido a la variabilidad que presentan los latidos, ya sean normales o patológicos, en cada sujeto. Un ejemplo de esto se puede observar en la ilustración 39, donde se muestran latidos “normales” extraídos de registros de la base de datos de arritmias del MIT-BIH. Los latidos (a), (b), (c) y (d) no presentan ninguna patología y, si bien son parecidos, las amplitudes y duraciones de las ondas varían de uno a otro; y los latidos (e) y (f) son los latidos considerados normales, aunque en realidad muestran una patología, de pacientes que padecen, por ejemplo, bloqueo de rama derecha (e) o utilizan marcapasos (f).

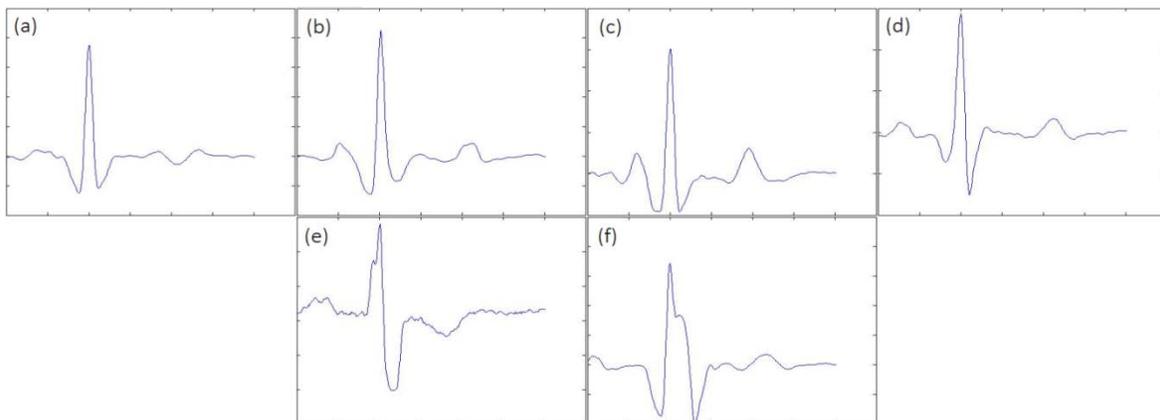


Ilustración 39 – Latidos normales para diferentes registros.³⁰

³⁰ Referencias: (a) Registro 100, (b) Registro 101, (c) Registro 106, (d) Registro 123, (e) Registro 118, (f) Registro 104

Implementación de la Máquina de Vectores de Soporte

Para la implementación, se utilizó la librería LIBSVM desarrollada por C. C. Chang y C. J. Lin. Según los autores, es un software sencillo, eficiente y de fácil uso para SVM tanto para problemas de clasificación como regresión. La librería está desarrollada para usarse con java, python, C y matlab³¹. Para una explicación de los parámetros a considerar en LIBSVM, a través del comando `svm_train`, ver el Anexo B.

Se debe comenzar por analizar qué tipo de núcleo es conveniente utilizar. Los autores de la librería sugieren empezar con una función de base radial (RBF), recordando que los tipos de núcleos utilizados son lineal, polinómico, RBF y sigmoideo. Se optó por aceptar la sugerencia y luego de utilizar como núcleo una función de base radial, se emplearon los otros tres tipos para contrastar resultados. Se utilizó como parámetro de comparación el porcentaje de aciertos de latidos anormales. La función de base radial demostró un superior desempeño frente a los otros tipos de núcleos. Esto puede observarse en la ilustración siguiente.

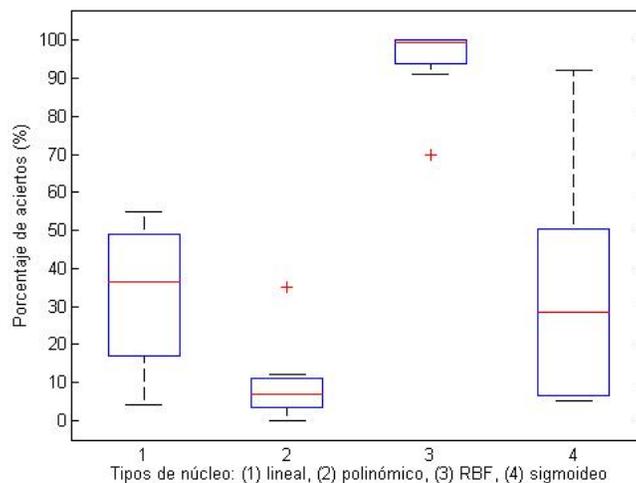


Ilustración 40 – Porcentaje de aciertos en función del tipo de núcleo para SVM.

Una vez elegido el núcleo a utilizar, resta ajustar tres parámetros más. Primero hay que definir cuántas muestras de deben analizar para entrenar la SVM y posteriormente elegir los valores de γ y ν óptimos. Para definir la cantidad de muestras se corrió el programa, variando la misma, para un conjunto de registros. Se concluyó que a medida que aumenta el número de muestras que se utiliza para el entrenamiento, también aumenta la precisión del modelo. Sin embargo, a partir de cierta cantidad, la media permanece casi constante mientras que la dispersión aumenta, ilustración 41. Se decidió, entonces, utilizar 135 muestras para el entrenamiento de SVM.

³¹ Puede encontrarse detalles sobre el algoritmo y referencias al respecto en: C. C. Chang; C. J. Lin. LIBSVM: LIBSVM – A Library for Support Vector Machine. ACM Transactions on Intelligent Systems and Technology, 2011. www.csie.ntu.edu.tw/~cjlin/libsvm/

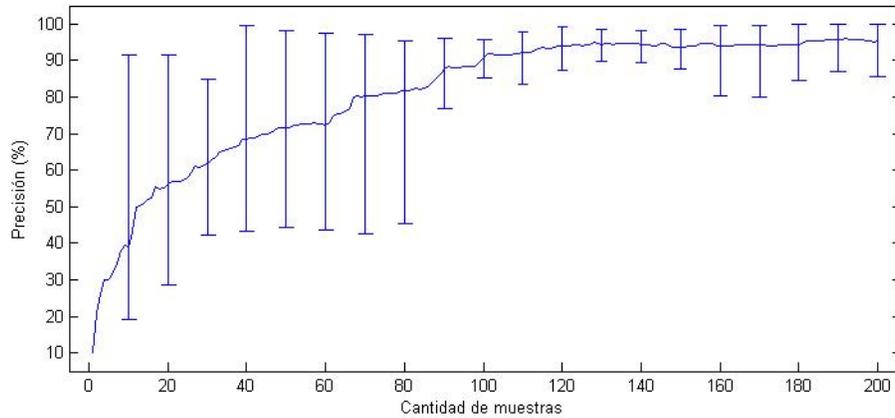


Ilustración 41 – Precisión vs cantidad de muestras consideradas para el entrenamiento del modelo.

Para determinar los valores de gamma y nu, se procedió de la misma manera que para encontrar la cantidad de muestras más conveniente. Se varió el valor entre 0.001 y 0.5 obteniéndose los resultados que se muestran a continuación.

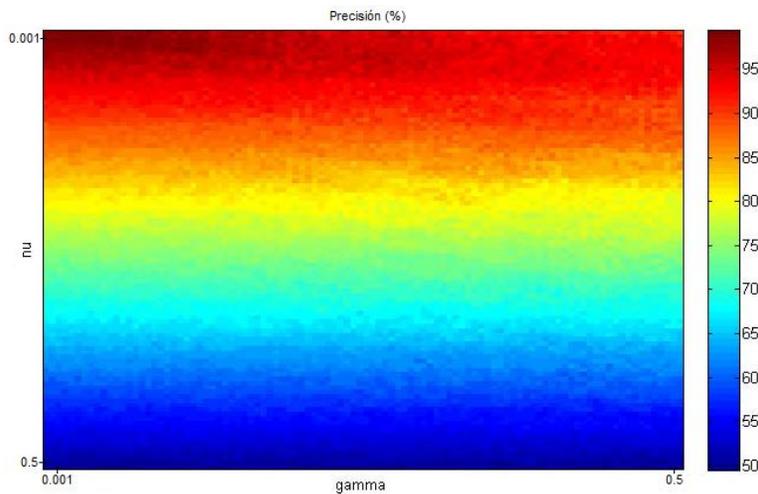


Ilustración 42 – Precisión en función de gamma y nu.

Cabe mencionar que en la ilustración anterior se tiene en cuenta la precisión con respecto a los datos de entrenamiento, es decir que, por ejemplo, una precisión del 100% significa que todos los datos que se utilizaron para entrenar el modelo son clasificados correctamente. Los resultados obtenidos son los esperados debido a que, como se mencionó anteriormente, nu es un parámetro que ajusta la tolerancia a los puntos atípicos. Mientras más chico sea nu, el modelo buscará incluir mayor cantidad de muestras. El inconveniente que esto genera es que al querer incluir todas las muestras dentro del límite de decisión, aumenta el riesgo de incluir en ese espacio, los datos que deberían quedar fuera de él. En nuestro caso en particular implica el riesgo de considerar latidos anormales como si fueran normales.

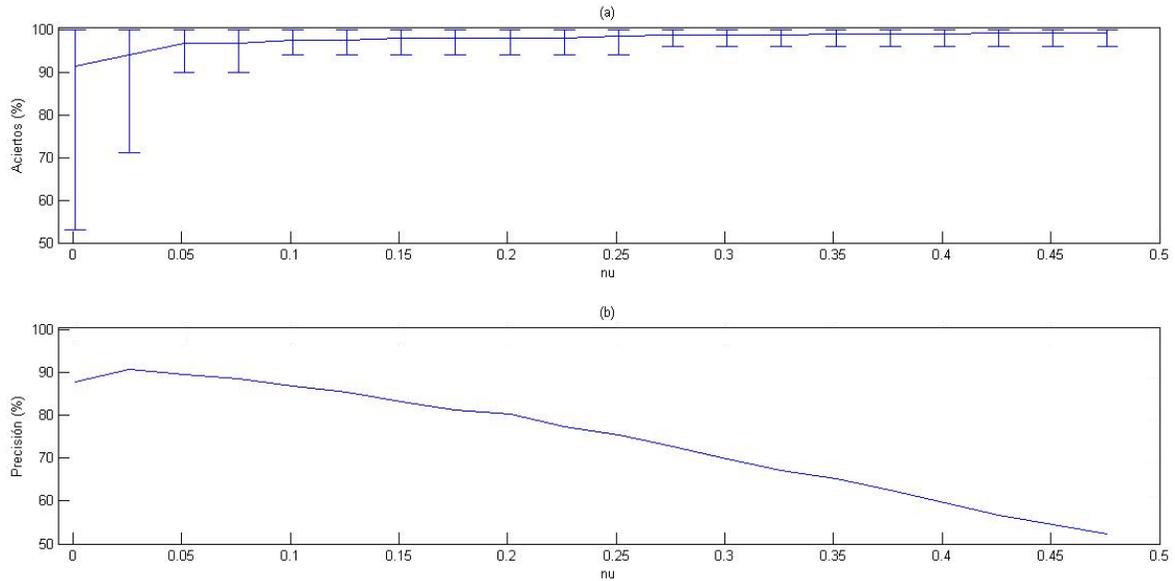


Ilustración 43 – (a) Aciertos de latidos anormales vs ν ; (b) Precisión vs ν .

Utilizar únicamente la precisión como parámetro para la elección de γ y ν no es adecuado, motivo por el cuál se usará en conjunto con el porcentaje de aciertos de latidos anormales. La idea es encontrar el punto donde el porcentaje de falsos negativos sea mínimo tratando de sacrificar lo menos posible la precisión. Como γ no tiene mucha influencia sobre la precisión, de acuerdo a la ilustración 42, se opta por tomar un valor de 0.3 para poder trabajar únicamente con ν .

Analizando los resultados obtenidos, ilustración 43, se decidió trabajar con ν igual a 0.1. El motivo de la elección es que sin importar el valor que elijamos, no es posible lograr un 100% de detección de los latidos anormales en todos los registros y, si se observan las gráficas, se aprecia que a partir de 0.1 el porcentaje de aciertos de latidos anormales aumenta muy levemente mientras que la precisión disminuye de una manera más notoria.

Capítulo 3: Resultados

Evaluación del desempeño del algoritmo

Detección de complejos QRS

El desempeño de este detector fue evaluado utilizando la base de datos del MIT-BIH Arrhythmia Database. Se tomaron al azar 27 registros con un total de 45.594 latidos normales y 13.731 latidos anormales (principalmente: contracción ventricular prematura, bloqueos de rama izquierda y derecha y contracción auricular prematura). Para poder comprender los resultados de esta evaluación se definen las siguientes clases de latidos:

- TP: Verdaderos positivos (True Positive) son los latidos correctamente detectados.
- FP: Falsos positivos (False Positive) son picos de la señal que el detector marcó erróneamente como latidos.
- FN: Falsos negativos (False Negative) son latidos que no fueron detectados.

Estas definiciones permiten calcular dos medidas del desempeño del detector:

- Valor Predictivo Positivo (+P): Es una medida de cuantos de los latidos detectados son verdaderos.

$$+P = \frac{TP}{TP + FP}$$

- Sensibilidad (S): Es una medida de cuantos latidos se detectan de entre todos los latidos analizados por el detector.

$$S = \frac{TP}{TP + FN}$$

Los resultados obtenidos se pueden observar en la siguiente tabla. En todas las clases de latidos definidas arriba, se tomaron separados los latidos normales y anormales con el objetivo de poder observar cómo se comporta la detección frente a ambos tipos.

registro	cantidad latidos		TP		FN		FP	+P	S		
	normales	anormales	normales	anormales	normales	anormales			normales	anormales	total
100	2237	34	2237	34	0	0	0	100.00%	100.00%	100.00%	100.00%
101	1859	5	1854	1	1	0	4	99.78%	99.95%	100.00%	99.95%
102	2127	60	2127	60	0	0	0	100.00%	100.00%	100.00%	100.00%
103	2081	2	2080	2	1	0	0	100.00%	99.95%	100.00%	99.95%
104	1542	686	1514	657	0	1	28	98.73%	100.00%	99.85%	99.95%
105	2525	46	2482	4	1	0	42	98.34%	99.96%	100.00%	99.96%
106	1507	520	1507	488	0	32	0	100.00%	100.00%	93.85%	98.42%
107	2077	58	2077	57	0	1	0	100.00%	100.00%	98.28%	99.95%
109	0	2530	0	2526	0	4	0	100.00%	-	99.84%	99.84%
112	2536	2	2536	2	0	0	0	100.00%	100.00%	100.00%	100.00%
113	1787	6	1787	6	0	0	0	100.00%	100.00%	100.00%	100.00%
115	1952	0	1952	0	0	0	0	100.00%	100.00%	-	100.00%
116	2301	110	2295	110	2	0	4	99.83%	99.91%	100.00%	99.92%
117	1533	1	1533	1	0	0	0	100.00%	100.00%	100.00%	100.00%
118	0	2277	0	2277	0	0	2	99.91%	-	100.00%	100.00%
122	2475	0	2475	0	0	0	0	100.00%	100.00%	-	100.00%
123	1514	3	1514	0	0	3	0	100.00%	100.00%	0.00%	99.80%
124	0	1618	0	1611	0	7	0	100.00%	-	99.57%	99.57%
200	1743	857	1739	856	0	1	4	99.85%	100.00%	99.88%	99.96%
201	1624	338	1624	278	0	60	0	100.00%	100.00%	82.25%	96.94%
202	2061	75	2061	67	0	8	0	100.00%	100.00%	89.33%	99.63%
203	2528	445	2509	427	4	18	15	99.49%	99.84%	95.96%	99.26%
205	2570	85	2568	82	2	3	0	100.00%	99.92%	96.47%	99.81%
209	2620	384	2619	384	0	0	1	99.97%	100.00%	100.00%	100.00%
214	0	2261	0	2257	0	4	2	99.91%	-	99.82%	99.82%
219	2081	72	2081	72	0	0	0	100.00%	100.00%	100.00%	100.00%
231	314	1256	314	1256	0	0	0	100.00%	100.00%	100.00%	100.00%
total	45594	13731	45485	13515	11	142	102	99.83%	99.98%	98.96%	99.74%

Tabla 4 - Desempeño del detector en la base de datos del MIT-BIH.

Si se observan los resultados totales, el detector tuvo un muy buen desempeño logrando un 99.83% de valor predictivo positivo y una sensibilidad del 99.74%. Si analizamos esta última separando los latidos normales de los anormales, las sensibilidades logradas son de 99.98% y 98.96% respectivamente. A simple vista hasta el peor de los resultados, que es la sensibilidad de los latidos anormales, se puede considerar como bueno ya que de 100 latidos analizados se perdería uno sólo. Sin embargo, si analizamos cada registro por separado se puede observar que para algunos de ellos el detector tuvo un rendimiento muy pobre llegando a tener un 0% de sensibilidad para los latidos anormales.

Dichos registros se analizaron detalladamente para poder detectar cual es el motivo por el cual el desempeño disminuye tanto. Se determinó que el algoritmo es más propenso a fallar cuando se da alguna de las siguientes condiciones:

- En los cambios de ritmo bruscos suele perder de uno a tres latidos hasta acondicionarse.
- Alternancia entre latidos normales y contracciones ventriculares prematuras con frecuencias altas.
- Cuando hay una contracción ventricular prematura entre dos latidos normales y el intervalo R-R entre estos dos últimos se asemeja al R-R promedio.
- Cuando el registro presenta contracciones auriculares prematuras aberrantes.

Clasificación de los latidos

Para la comprobación de la técnica estudiada se emplearon 26.877 latidos de los cuales 20.500 corresponden a latidos sanos y 6.377 corresponden a latidos patológicos. Se utilizaron registros con diferentes tipos de patologías y niveles de ruido, no agregado, para poder observar cómo se desempeña el algoritmo en diversos ambientes. En el conjunto de latidos “normales” se agruparon latidos sanos, latidos generados por marcapasos externos implantables y latidos con bloqueo de rama; y el conjunto de latidos “anormales” está formado por las siguientes patologías: contracción ventricular prematura (C.V.P), contracción auricular prematura (C.A.P), contracción nodal prematura (C.N.P), latidos inclasificables (Q) y fusión entre CVP y latido normal (F).

Los resultados obtenidos se muestran a continuación.

Tipo de latidos	Porcentaje de aciertos	
	Medio	Rango
Latidos Normales	69,42 %	50 % - 86 %
Latidos Anormales	98.93 %	88 % - 100 %

Tabla 5 - Desempeño del algoritmo.

Donde el porcentaje de aciertos medio es el promedio de los porcentajes de aciertos de los registros analizados, y el rango muestra el peor y el mejor porcentaje obtenido, respectivamente. Si desglosamos el desempeño de los latidos anormales en las distintas patologías analizadas, los resultados son los siguientes:

Patología	Porcentaje de aciertos
C.A.P	86,98 %
C.V.P	99,69 %
C.N.P	95,52 %
Q	100 %
F	100 %

Tabla 6 - Porcentaje de aciertos discriminado en patologías.

Si analizamos los datos de manera general, o sea trabajando con promedios, y considerando que del 100% de los latidos de cada registro, tan sólo un 7,5% corresponde a latidos anormales, la

implementación de este algoritmo en dispositivos de monitoreo móviles permite un ahorro de memoria, o de transmisión de datos según el caso, de un 63,83%. En la tabla 7 se muestran los ahorros obtenidos para algunos de los registros analizados.

Registro	ALN*	ALA*	Ahorro memoria/transmisión
100	72 %	94,11 %	70,91 %
101	62 %	100 %	61,83 %
102	86 %	98,74 %	84,57 %
106	86 %	99,42 %	56,32 %
123	86 %	100 %	85,83 %
201	77 %	99,65 %	60,98 %
203	63 %	99,77 %	51,91 %
205	50 %	100 %	48,35 %
219	67 %	93,05 %	64,68 %

*ALN: Aciertos latidos normales; ALA: Aciertos latidos anormales

Tabla 7 - Ahorro de memoria/transmisión para algunos registros analizados.

El registro 205 fue el que presentó el peor desempeño de todos los registros analizados, consiguiendo un ahorro del 48,35%. A éste le sigue el registro 115, que no se encuentra en la tabla anterior, que consiguió un ahorro del 50%. Traduciendo estos resultados a nuestro caso en particular, considerando el caso de almacenamiento y no el de transmisión, tener un ahorro del 48,35% significa que se logra casi duplicar la capacidad de almacenamiento sin necesidad de hacer cambios de hardware.

Influencia del ruido en la señal

Influencia del ruido en la detección

Para estudiar cómo es el comportamiento del algoritmo frente al ruido, se agregó a las señales originales ruido gaussiano con distintas relaciones señal/ruido (RSR). En la tabla 8 se pueden observar los resultados. El algoritmo de detección mostró un buen desempeño frente al ruido, logrando los mismos resultados que los obtenidos con la señal original, hasta con una RSR igual a 5. Para valores menores a 5, el algoritmo empieza a fallar detectando ruido como si fuese un latido.

Relación señal/ruido*	Porcentaje de aciertos
100	100 %
50	100 %
25	100 %
10	100 %
5	100 %
2	55 %
1	37 %

*medido linealmente

Tabla 8 - Influencia del ruido en la detección.

Influencia del ruido en la clasificación

Al igual que con la etapa de detección, se pretendió estudiar como es el comportamiento del algoritmo en la etapa de clasificación frente al ruido. Se procedió de similar manera que en la sección anterior y los resultados obtenidos se muestran en la tabla 9. Para señales con una RSR mayor a 10 el algoritmo tiene el mismo desempeño que con las señales originales. A partir de señales con RSR menores a 10 el desempeño disminuye rápidamente, siendo muy bajo para RSR menores que 5.

Relación señal/ruido*	Porcentaje de aciertos
100	100 %
50	100 %
25	100 %
10	96 %
5	70 %
2	16 %
1	6 %
*medido linealmente	

Tabla 9 - Influencia del ruido en la clasificación.

Prueba de concepto

Debido a que todo el desarrollo del proyecto integrador se realizó de manera off-line, es decir que se utilizó todo el registro entero del electrocardiograma, una vez obtenidos los resultados se procedió a adaptar el algoritmo para su implementación en tiempo real. Para realizar la prueba de concepto³² se utilizó el programa Simulink® que es de los mismos desarrolladores de Matlab® y en consecuencia, la adaptación no requirió una migración del lenguaje de programación utilizado ya que Simulink® es compatible con la mayoría de las funciones definidas en Matlab®.

Como señales de entrada se utilizaron los mismos registros de la base de datos del MIT-BIH. Por lo tanto, el primer paso fue configurar la entrada para que lea los registros una muestra a la vez, simulando un sistema de adquisición de tiempo real. De esta forma, en un instante determinado no se conocen los valores futuros y, salvo que se hayan guardado en un buffer, se pierden los valores anteriores.

La prueba de concepto consta de dos modelos realizados en Simulink®, uno para el entrenamiento y otro para la clasificación, y un código en Matlab® que recibe los datos del modelo de entrenamiento, les aplica el Análisis de Componentes Independientes y entrena la Máquina de Vectores de Soporte. Una vez realizado el entrenamiento, los valores obtenidos son enviados al modelo de clasificación para, como su nombre lo indica, proceder a clasificar los latidos. Este procedimiento puede observarse con más detalles a continuación.

³² Prueba de concepto: implementación, resumida o incompleta de un método o una idea, realizada con el propósito de verificar que el concepto o teoría en cuestión es susceptible de ser explotada de una manera útil.

Se presenta, con un ejemplo, la realización de la simulación de la prueba de concepto. Primero se seleccionó un registro que posee todos los latidos normales. Una porción de este registro se muestra a continuación.

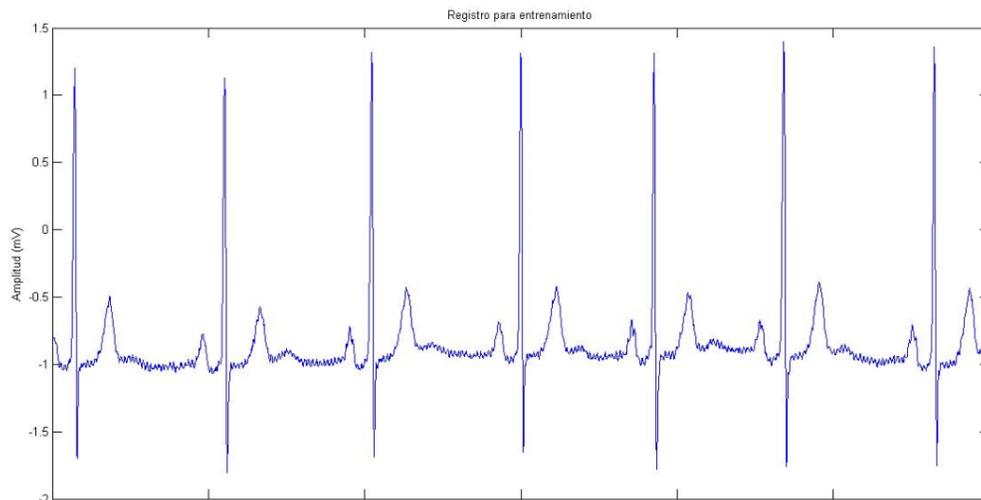


Ilustración 44 - Registro para entrenamiento.³³

Ese registro, modificado para ser simulado como si fuera en tiempo real, se utiliza como entrada del modelo de entrenamiento. Éste detecta y guarda los 200 primeros latidos con sus respectivos valores de intervalo RR, que es la cantidad de latidos necesarios para realizar el entrenamiento de la Máquina de Vectores de Soporte. En las siguientes ilustraciones se muestran, primero los latidos segmentados que luego son enviados a Matlab® y en segundo lugar, el esquema del modelo de entrenamiento.

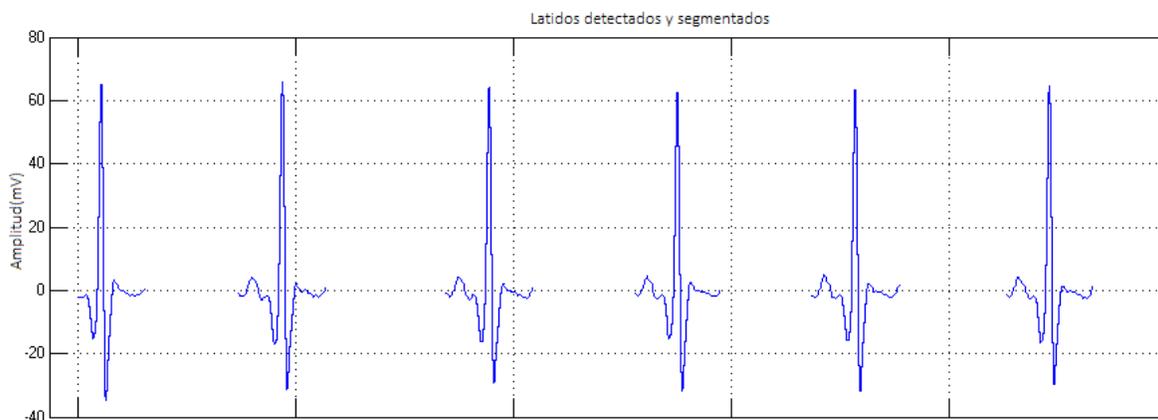


Ilustración 45 - Latidos detectados y segmentados.

³³ Registro 123 de la base de datos de arritmias del MIT-BIH.

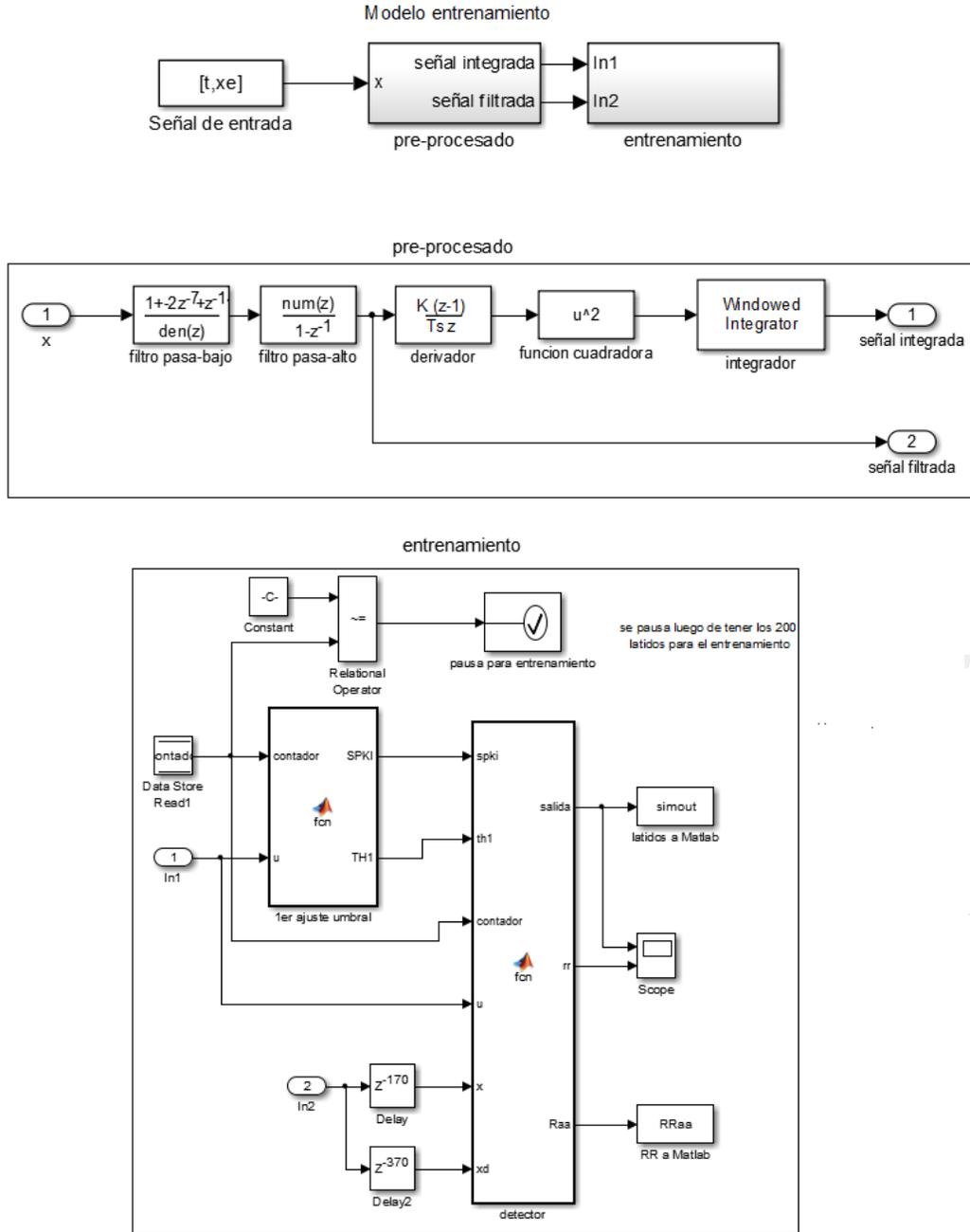


Ilustración 46 - Modelo de entrenamiento.

Una vez obtenidos los primeros 200 latidos se ejecuta el código desarrollado en Matlab® con lo que obtenemos los vectores de soporte y otros parámetros extraídos de SVM que nos permitirán clasificar los latidos que se vayan detectando. Dicho código se muestra en el Anexo C.

Con estos datos es posible correr el modelo de clasificación que, a medida que detecta un latido, lo clasifica e incrementa un contador dependiendo si éste fue clasificado como normal o anormal. A continuación se muestra el esquema de este modelo. El bloque de pre-procesado es idéntico al

del modelo anterior, y el bloque “detector” es similar al de “entrenamiento”, de modo que estos dos bloques no se muestran.

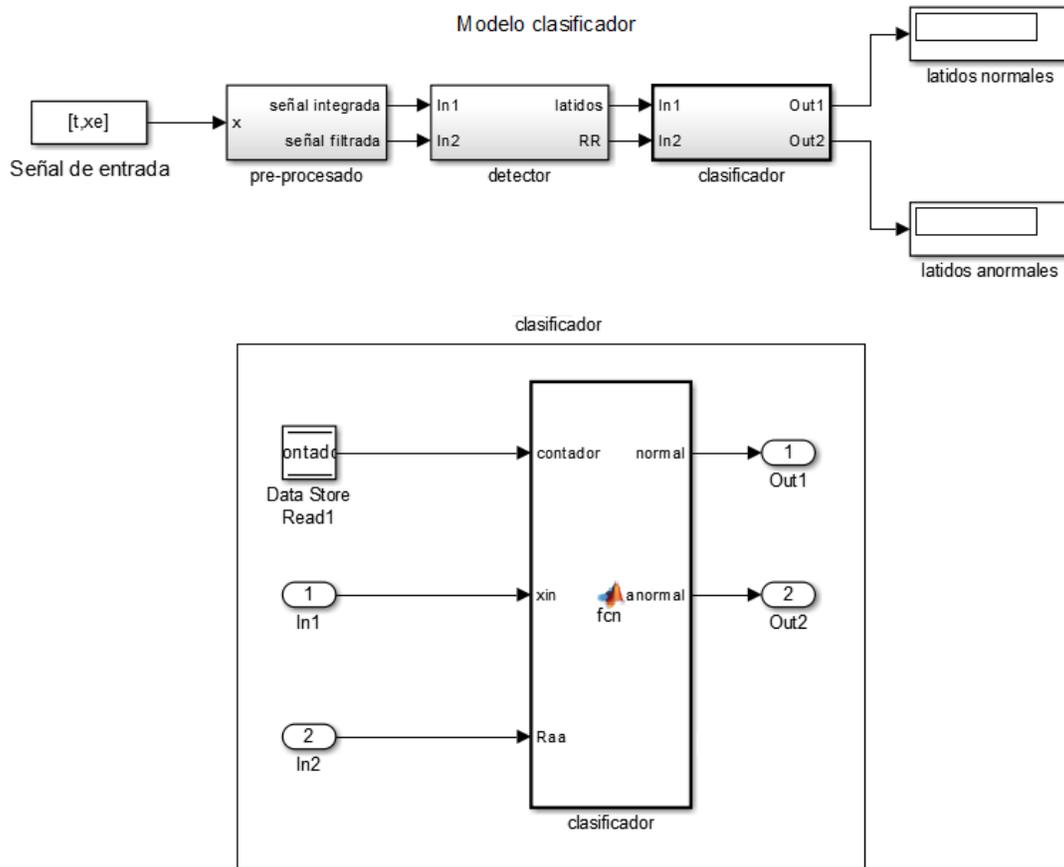


Ilustración 47 - Modelo de clasificación.

Al correr el modelo de clasificación con el mismo registro utilizado para el entrenamiento, se obtuvieron los siguientes resultados:

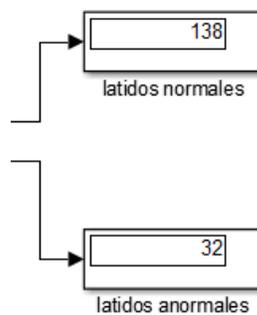


Ilustración 48 - Resultados de la primera simulación.³⁴

³⁴ Utilizando el mismo registro empleado en el entrenamiento.

De los 170 latidos clasificados, 32 fueron mal clasificados debido a que todos eran normales. Por lo tanto el porcentaje de acierto fue del 76,8%. A continuación, se corrió de nuevo el modelo de clasificación pero utilizando como entrada, un registro que posee latidos generados por un marcapasos externo implantable.

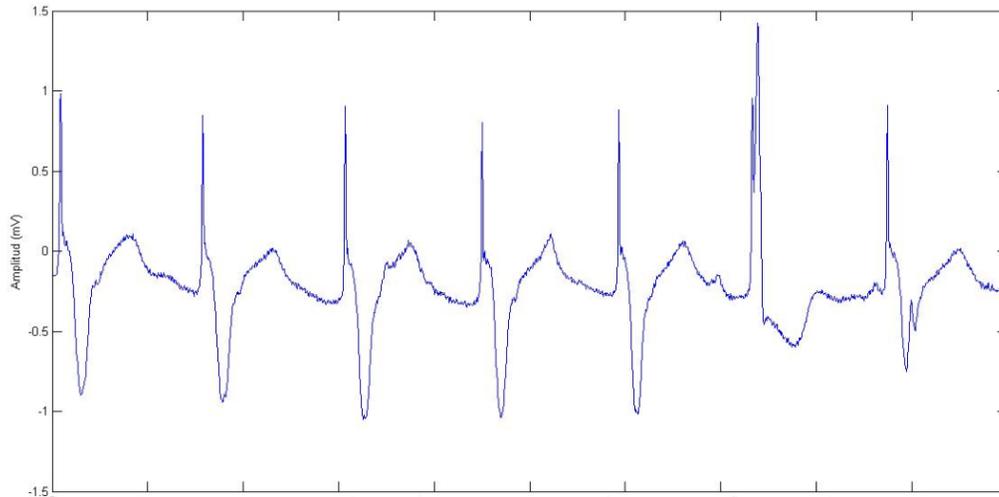


Ilustración 49 - Latidos generados por marcapasos externo implantable.³⁵

Los resultados obtenidos se muestran a continuación:

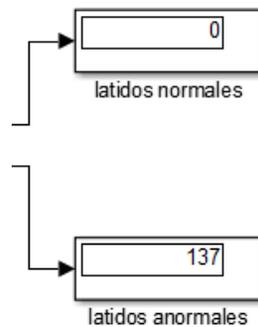


Ilustración 50 - Resultados de la segunda simulación.³⁶

Se puede observar que todos los latidos fueron clasificados correctamente ya que ninguno era normal. Con estas dos simulaciones se puede observar también lo que se planteó a lo largo del trabajo integrador, que es darle más importancia a la clasificación correcta de latidos anormales aunque esto produzca una disminución del desempeño del clasificador.

³⁵ Registro 104 de la base de datos de arritmias del MIT-BIH.

³⁶ Utilizando un registro con latidos generados por un marcapasos externo implantable.

Conclusiones

Se logró implementar un algoritmo de detección de complejos QRS on-line que posea un buen desempeño al mismo tiempo que no requiera grandes recursos computacionales. Bajo ciertas condiciones, el desempeño del algoritmo disminuye considerablemente, sin embargo, se logró salvar este error en las etapas posteriores.

Mediante la utilización del Análisis de Componentes Independientes se obtuvo un número reducido de características, cinco obtenidas con ICA y tres con duraciones de los intervalos RR. Debido a esto, se limita el clasificador a uno de tipo binario, es decir que sólo puede diferenciar entre latidos normales y anormales, no pudiendo distinguir entre las diversas patologías que puedan presentarse. Este hecho no representa un problema para el trabajo, ya que el objetivo es reducir la cantidad de información transmitida o almacenada, dejando el diagnóstico de la patología a un especialista. Cabe resaltar que, según los antecedentes encontrados, los trabajos similares de clasificación binaria utilizan más del doble de características y consiguen simplemente un porcentaje de aciertos un poco mayor [15]. Por otro lado, se encontraron trabajos de clasificación multi-clase con un desempeño muy superior pero utilizan gran cantidad de características, y todos ellos son para implementación off-line [13, 16, 17].

Gracias a las Máquinas de Vectores de Soporte de una clase, se evitó la necesidad de disponer de grandes bases de datos para el entrenamiento de la red. Otra ventaja importante es que puede ajustarse el considerado latido “normal” a cada paciente, incluso si este latido presenta alguna patología que se desee omitir durante el estudio, como por ejemplo, el uso del marcapasos.

En cuanto a los resultados obtenidos, se puede llegar a las siguientes conclusiones:

- El algoritmo alcanzó una precisión media de casi 70% permitiendo un ahorro de memoria, o de transmisión de datos, de un 64%, teniendo como peor resultado un ahorro del 48% que permite duplicar la capacidad de almacenamiento sin necesidad de hacer modificaciones de hardware. Es posible mejorar esa precisión, sin embargo, se le dio más importancia a la correcta clasificación de latidos anormales, logrando un porcentaje de aciertos medio del 99%. Aumentar la precisión del programa implica una disminución de aciertos de latidos anormales.
- La patología que más inconvenientes presenta al ser clasificada es la contracción auricular prematura. Esto se debe a la similitud que posee con un latido sano, sobre todo debido a que la parte del registro más relevante para la clasificación es el complejo QRS.
- Los análisis realizados para ver el comportamiento del algoritmo frente al ruido mostraron que tanto la etapa de detección como de clasificación presentan buena tolerancia al mismo. El ruido es más desfavorable para el detector debido a que le impide detectar el latido, implicando una pérdida de información. Mientras que para el clasificador sólo disminuye el desempeño sin perder información, ya que el latido con ruido es clasificado como anormal.

Finalmente, se logró realizar una prueba de concepto del sistema desarrollado, demostrando que éste podría ser utilizado en aplicaciones de tiempo real. Para ello, se utilizó el programa Simulink®.

Propuestas para Investigaciones Futuras

En base a los distintos resultados obtenidos a lo largo de la etapa de experimentación y caracterización del sistema, se propone una serie de mejoras:

Etapa de extracción de características: Debido a que en esta etapa sólo se utilizó el algoritmo de Análisis de Componentes Independientes propuesto por A. Hyvarinen [9], se debería evaluar la utilización de otros algoritmos fastICA para analizar si es posible lograr mejores desempeños. Además, se debería probar extraer de la etapa de detección, otras características morfológicas como ser las duraciones y amplitudes de las distintas ondas, principalmente para lograr corregir la deficiencia que posee el sistema desarrollado frente a las contracciones auriculares prematuras. Esto último recordando que el objetivo es la implementación en procesadores de bajos recursos por lo que se debe tener cuidado a la hora de agregar características.

Etapa de clasificación: Modificaciones sobre la etapa de clasificación incluyen la implementación de diferentes funciones kernel así como la utilización de diferentes estrategias de inteligencia artificial.

En cuanto a desarrollos futuros, se mencionan algunos a continuación:

- Implementación del sistema en un microprocesador para conectarlo a un dispositivo de adquisición, utilizando por ejemplo, una placa Raspberry Pi.
- Utilizar el algoritmo para monitorizar deportistas de alto rendimiento con el fin de enviar alarmas al servicio de emergencias durante el desarrollo de la actividad.
- Adaptación del algoritmo propuesto para implementación en teléfonos inteligentes para que al detectar éstos una anomalía, alerten al servicio de emergencia y envíen la información del registro electrocardiográfico para poder analizar la patología.
- Desarrollo de un sistema de filtrado con Análisis de Componentes Independientes y Análisis de Componentes Principales, utilizando un conjunto de derivaciones.

Bibliografía y Referencias

- [1] C. Guyton; J. E. Hall. Tratado de fisiología médica. 12^a Edición. Madrid. W. F. Ganong. Fisiología Médica. 23^a Edición. México, 2010.
- [2] W. F. Ganong. Fisiología Médica. 23^a Edición. México, 2010.
- [3] N.V. Thakor; J.G. Webster; W.J. Tompkins. Optimal QRS detection. Medical and Biological Engineering, pages 343–350, 1983.
- [4] W.J. Tompkins; V.X. Afonso. Biomedical Digital Signal Processing: C-Language Examples and Laboratory Experiments for the IBM PC, chapter 12, pages 236–264. Englewood Cliffs, NJ: Prentice-Hall, 1993.
- [5] Hyvarinen; E. Oja. Independent Component Analysis: Algorithms and Applications. Neural Networks Research Centre. Helsinki University of Technology. 2000.
- [6] Hyvarinen; J. Karhunen; E. Oja. Independent Component Analysis. John Wiley and sons, inc. 2001.
- [7] McGraw-Hill Interamericana de España, SL. www.mhe.es/universidad/ciencias_matematicas/pena/home/CAPITULO.PDF
- [8] Reyna. SVM de una clase: aplicación a detección de novedad. IIE, FING, 2009.
- [9] Hyvarinen; E. Oja. A Fast Fixed-Point Algorithm for Independent Component Analysis. Helsinki University of Technology. Finlandia, 1997.
- [10] C. Chang; C. J. Lin. LIBSVM: LIBSVM – A Library for Support Vector Machine. ACM Transactions on Intelligent Systems and Technology, 2011. www.csie.ntu.edu.tw/~cjlin/libsvm/
- [11] Moody GB, Mark RG. The impact of the MIT-BIH Arrhythmia Database. *IEEE Eng in Med and Biol* 20(3):45-50 (May-June 2001). (PMID: 11446209).
- [12] Goldberger AL, Amaral LAN, Glass L, Hausdorff JM, Ivanov PCh, Mark RG, Mietus JE, Moody GB, Peng C-K, Stanley HE. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. *Circulation* 101(23):e215-e220 [Circulation Electronic Pages; <http://circ.ahajournals.org/cgi/content/full/101/23/e215>]; 2000 (June 13).

[13] E. J. da Silva Luz; W. R. Schwartz; G. Cámara Chávez; D. Menotti. ECG-based Heartbeat Classification for Arrhythmia Detection: A Survey. Article in Computers methods and programs in biomedicine, 2015.

[14] E. Delgado Trejos. Ventricular Arrhythmia Detection Using Non-linear Dynamics Techniques on ECG Signals. Universidad Nacional de Colombia. Colombia, 2003. [15] T. Jolliffe. Principal Component Analysis, Second Edition. Springer. UK, 2002.

[15] G. Bortolan; I. I. Christov; W. Pedrycz. Hiperbox classifiers for ECG beat analysis, in: Computers in Cardiology, 2007.

[16] J.A. Fischer. Caracterización de electrocardiogramas mediante redes neuronales y transformada ondita. FCEFYN, Universidad Nacional de Córdoba, Córdoba, Argentina, 2014..

[17] P. de Chazar. Automatic Classification of Heartbeats Using ECG Morphology and Heartbeat Interval Features. IEEE Transactions on Biomedical Engineering, Vol. 51, July 2004.

Anexos

Anexo A – Código fuente del algoritmo propuesto

A continuación se presenta el código en lenguaje .m

```
%%DETECTOR DE LATIDOS

%Toma los datos de la base de datos y lo trabaja para
%mostrarlo correctamente
x=val(1,:);
x=x-1024;
x=x./200;

%Frecuencia de muestreo utilizada por el ECG
Fs=360;

%definen variables para pasos posteriores
Y=zeros(1,length(x));
PEAK=Y;
Ypb=Y;
Ypa=Y;
Yd=Y;
Ymw=Y;
W=Y;

%filtro pasa-bajo
%aplicando el filtro en forma directa 2
%la frecuencia de cutoff es de aprox. 11 Hz y la ganancia 49.
%retrasa 6 muestras

for i=15:length(x)
    W(i)=x(i)+2*W(i-1)-W(i-2);
end
clear i
for i=15:length(x)
    Ypb(i)=(W(i)-2*W(i-7)+W(i-14))/49;
end
clear i

%filtro pasa-alto
%aplicando el filtro en forma directa 2
%la frecuencia de cutoff es de aprox. 5 Hz y retrasa 24 muestras

for i=15:length(x)
    W(i)=Ypb(i)+W(i-1);
end
clear i
for i=49:length(x)
    Ypa(i)=(-W(i)+48*W(i-24)-48*W(i-25)+W(i-48))/48;
end
clear i

%ecuacion derivador de 5 puntos
%rta casi lineal entre dc y 30 Hz con retraso de 2 muestras
```

```

for i=5:length(x)
    Yd(i)=(2*Ypa(i)+Ypa(i-1)-Ypa(i-3)-2*Ypa(i-4))/8;
end
clear i

%funcion "cuadradora"
%se eleva la señal al cuadrado pto a pto
Yc=Yd.^2;

%Integracion moving-window
%N=64 -> 177ms para Fs=360Hz para dividir por una potencia de dos
N=64;
for i=N:length(x)
    X=0;
    %Se resuelve el vector X=x(i-(N-1))+X(i-(N-2))+...+x(i))
    for m=(N-1):-1:0
        X=X+Yc(i-m);
    end
    %se divide por 1/N
    Ymw(i)=X/N;
end
clear i & m

%Deteccion del primer pico R
%se toma el primer segundo (360 muestras) y se busca el valor maximo
%se guarda en posicion la ubicacion de la muestra con el valor maximo
peak=0;
pos_peak=0;
for i=Fs:2*Fs
    if Ymw(i)>peak
        peak=Ymw(i);
    end
end
clear i & m

%Ajuste de los umbrales de deteccion
%SPKI es el estimado del pico de señal
%NPKI es el estimado del pico de ruido
%THRESHOLD1 es el primer umbral aplicado
%THRESHOLD2 es el segundo umbral aplicado.
SPKI=peak/2;
SPKI=0.125*peak+0.875*SPKI;
NPKI=0;

THRESHOLD1=peak/4;
THRESHOLD2=0.5*THRESHOLD1;

%DETECCION DE R

%se definen las variables
regRv=zeros(1,200000);
regRp=regRv;
RR=regRv;
bkRR=zeros(1,3);

```

```

bkpR=zeros(1,3);
i=Fs;
k=1;
z=1;
salto=0;
guardar=0;
l=8;
rrav1=zeros(1,8);
lat_x=zeros(3010,201);
lat_f=zeros(3010,201);

while i < length(x)
    %busco la primer loma
    if z==1;
        while Ymw(i) <= Ymw(i-1)
            i=i+1;
            if i==length(x)
                break
            end
        end
        peak=Ymw(i);
        pos_peak=i;
        z=0;
    end

    %Busqueda hacia adelante
    %tiene que superar el umbral en la señal filtrada
    if Ymw(i)<(peak/2)
        if (peak>THRESHOLD1)
            %actualiacion del umbral adaptativo (debido a la señal)
            SPKI=0.125*peak+0.875*SPKI;

            %guardamos la ubicacion de Ymw(i) para la busqueda hacia
atras
            bkRR(3)=bkRR(2);
            bkRR(2)=bkRR(1);
            bkRR(1)=pos_peak;

            %Buscamos el pico en la señal original
            %(ventana de N m con delay de 30m)
            regRv(k)=-128;
            for j=(pos_peak-77):1:(pos_peak-13)
                if (x(j)>regRv(k))

                    %guardamos en dos vectores la posicion
                    %y el valor del pico
                    regRp(k)=j;
                    regRv(k)=x(j);
                end
            end

            bkpR(3)=bkpR(2);
            bkpR(2)=bkpR(1);
            bkpR(1)=regRp(k);

```

```

%Analizamos el intervalo RR
%Se saca el promedio de los ultimos 8 intervalos (rrav1)
%y el promedio de los ultimos 8 intervalos normales (rrav2)

%se guarda el intervalo
if k>1
    RR(k-1)=regRp(k)-regRp(k-1);
end
%se saca el primer promedio
if (k>8)
    rrav1=[RR(k-1) RR(k-2) RR(k-3) RR(k-4),...
          RR(k-5) RR(k-6) RR(k-7) RR(k-8)];
    RRav1=(sum(rrav1))/8;
end
%se saca el segundo promedio
if (k==9)
    RRn=RR;
    RRav2=RRav1;
elseif (k>9 && RR(k-1)>(0.92*RRav1) && RR(k-1)<(1.16*RRav1))
    RRn(1)=RR(k-1);
    rrav2=[RRn(1) RRn(1-1) RRn(1-2) RRn(1-3),...
          RRn(1-4) RRn(1-5) RRn(1-6) RRn(1-7)];
    RRav2=(sum(rrav2))/8;
    l=1+1;
end

%se ajustan las banderas para guardar el latido y saltar
salto=1;
guardar=1;
z=1;

else

%actualiacion del umbral adaptativo (debido al ruido)
NPKI=0.125*peak+0.875*NPKI;
z=1;

end

%actualiacion del umbral adaptativo
THRESHOLD1=NPKI+0.21*(SPKI-NPKI);
THRESHOLD2=0.5*THRESHOLD1;

%se verifica si el pico se encontro entre el RR permitido
%que es 1.66*RRav2
if (k>9)
    if (RR(k-1)>(1.66*RRav2))

        %% se inicia la busqueda hacia atras %%
m=bkRR(2)+72+N/2;
z=1;

while m<bkRR(1)-N

```

```

%busco la primer loma
if z==1;
    while Ymw(m) <= Ymw(m-1)
        m=m+1;
        if m>=length(x)
            break
        end
    end
    peak=Ymw(m);
    pos_peak=m;
    z=0;
end

if Ymw(m)<(peak/2)
    if (peak>THRESHOLD2)

        %actualiacion del umbral adaptativo (debido a la señal)
        SPKI=0.25*peak+0.75*SPKI;
        bkRR(1)=pos_peak;

        %Buscamos el pico en la señal original
        %(ventana de 41m con delay de 25m)
        regRv(k)=-128;
        for j=(pos_peak-77):1:(pos_peak-13)
            if (x(j)>regRv(k))
                %guardamos en dos vectores la posicion
                %y el valor del pico
                regRp(k)=j;
                regRv(k)=x(j);
            end
        end
        bkpR(1)=regRp(k);

        %acomodo i para el nuevo pico encontrado
        %fuerzo el final de la busqueda ya que
        %encontre el latido
        i=m;
        salto=1;
        guardar=1;
        z=1;

        break
    else
        %actualiacion del umbral adaptativo (debido al ruido)
        NPKI=0.25*peak+0.75*NPKI;
        z=1;
    end

    %actualiacion del umbral adaptativo
    %dandole mas peso al ultimo pico
    THRESHOLD1=NPKI+0.25*(SPKI-NPKI);

```

```

        THRESHOLD2=0.5*THRESHOLD1;

        elseif Ymw(m)>peak

            peak=Ymw(m);
            pos_peak=m;

        end

            m=m+1;
end

        %% fin busqueda hacia atras %%

        end
    end

elseif Ymw(i)>peak

    peak=Ymw(i);
    pos_peak=i;

end

%guardo el latido (ventana de 201 muestras centrada en R)
if guardar==1 && bkpR(2)>73
    o=1;
    for j=bkpR(2)-100:1:bkpR(2)+100
        lat_x(k,o)=x(j);
        lat_f(k,o)=Ypa(j+30);
        o=o+1;
    end

    guardar=0;
    clear o

    %guardo los intervalos RR actual y anterior y promedio
    %y las relaciones act/ant, act/prom y ant/prom
    RRactual(k)=bkpR(1)-bkpR(2);
    RRanterior(k)=bkpR(2)-bkpR(3);
    if k>9
        RRpromedio(k)=RRav1;

        Raa(k)=RRactual(k)/RRanterior(k);
        Racp(k)=RRactual(k)/RRpromedio(k);
        Ranp(k)=RRanterior(k)/RRpromedio(k);

    end
    ubicacion(k)=bkpR(2);
    cont(k)=k;
end

%Salta 200ms donde no es posible encontrar QRS

```

```

    if salto==1
        k=k+1;
        i=i+72;
        salto=0;
    end

    i=i+1;
end

%%EXTRACCION DE CARACTERISTICAS - ICA

%seleccionamos las muestras para ICA y corremos el programa FastICA
xica=lat_f(10:100,:);

%numeros de IC a considerar
nic=5;

%corre la libreria fastica reduciendo la dimensi3n de xica a 5 (lasEig)
%y tomando nic componentes independientes
[IC_fastica] = fastica(xica, 'lastEig', 5, 'numOfIC', nic);

%obtengo la invIC para poder calcular A
invIC=transpose(IC_fastica)*inv(IC_fastica*transpose(IC_fastica));

%saco la matriz A media
nm=10;

matrizA=zeros(nm,nic);
A_media=zeros(1,nic);

for i=1:nm
    matrizA(i,:)=xica(i,:)*invIC;
end
for i=1:nic
    A_media(i)=mean(matrizA(:,i));
end

%Se extrae los pesos de A para cada latido nuevo luego de centrar la
señal
%dsp se dividen los pesos por A media para tener las caracteristicas
for i=1:length(cont)
    lat_cen=lat_f(i,:)-mean(lat_f(i,:));

    A_actual(i,:)=lat_f(i,:)*invIC;

    A_norm(i,1:nic)=A_actual(i,:)./A_media;
end

%Se arman las matrices de entrenamiento
data_train=[A_norm(10:209,:) Raa(10:209)' Racp(10:209)' Ranp(10:209)'];
clas_train=ones(length(data_train(:,1)),1);

```

```
%%CLASIFICACION - SVM

% Se corre la libreria SVM

%se utilizan los siguientes parametros
%tipo de svm (-s) - SVM de una clase(2)
%tipo de nucleo (-t) - RBF
%nu=0.1, gamma=0.3, epsilon=0.001
cmd = '-s 2 -t 2 -n 0.1 -g 0.3 -e 0.001';

%se entrena la red y se extrae el modelo
modelo = svmtrain(clas_train,data_train, cmd);

%Se clasifican los latidos
[predict_label] = svmpredict(clas_test,data_test, modelo);
```

Anexo B – Parámetros de svm_train (Implementación en matlab)

Modelo = svmtrain (vector de clasificación, matriz de entrenamiento, ['opciones']);

- Vector de clasificación: vector de dimensión $m \times 1$ con las etiquetas de los vectores de entrenamiento.
- Matriz de entrenamiento: matriz de dimensión $m \times n$, con m vectores de entrenamiento de n características.
- Opciones (iguales para todos los lenguajes de programación):
 - –s tipo de SVM: selecciona el tipo de SVM a utilizar (por defecto 0).
 - 0 -- C-SVC
 - 1 -- nu-SVC
 - 2 -- SVM de una clase
 - 3 -- épsilon-SVR (regresión por vectores de soporte)
 - 4 -- un-SVR (regresión por vectores de soporte)
 - –t tipo de núcleo: selecciona el tipo de núcleo (por defecto 2).
 - 0 -- lineal: $u \cdot v$
 - 1 -- polinómico: $(\gamma \cdot u \cdot v + \text{coef0})^{\text{grado}}$
 - 2 -- RBF: $\exp(-\gamma \cdot |u-v|^2)$
 - 3 -- sigmoideo: $\tanh(\gamma \cdot u \cdot v + \text{coef0})$
 - 4 -- núcleo pre-computado (el archivo de entrenamiento tiene los valores computados)
 - –d grado: selecciona el grado del núcleo (por defecto 3).
 - –g gamma (γ): selecciona γ en función del núcleo (por defecto $1/(\text{máximo valor de las características})$).
 - –r coef0: selecciona coef0 de la función núcleo (por defecto 0).
 - –c cost0: selecciona el parámetro C de costo de C-SVM, épsilon-SVR y un-SVR (por defecto 1).
 - –n nu (ν): selecciona el parámetro ν de nu-SVC, SVM de una clase y nu-SVR (por defecto 0.5).
 - –p épsilon: selecciona épsilon en la función de pérdida de épsilon-SVR (por defecto 0.1).
 - –m cachesize: selecciona el caché de datos en memoria en MB (por defecto 100).
 - –e épsilon: selecciona la tolerancia del criterio de terminación (por defecto 0.001).

- -h reducción: determina si se usa o no la heurística de reducción (shrinking) al resolver el problema de optimización de SVM, 0 o 1 (por defecto 1).
- -b estimaciones de probabilidad: determina se entrena un modelo SVC o SVR para estimación de probabilidad, 0 o 1 (por defecto 0).
- -wi pesos: selecciona el parámetro C de la clase i a $\text{peso} * C$ en C-SVC (por defecto 1 en todos).
- -v n: validación cruzada en n grupos.

Anexo C – Código fuente empleado en la prueba de concepto

El código .m que se utiliza para el entrenamiento se muestra a continuación:

```
%% toma las primeras 200 muestras para el entrenamiento
clear xica

%eliminamos los elementos NaN, estos son los datos que
%se extraen del modelo de entrenamiento
simout(isnan(simout))=[];
RRaa(isnan(RRaa))=[];

%separamos los latidos
k=1;
pico=-10;
muestras=zeros(200,201);

for j=1:200
    for i=k:k+200
        if simout(i)>pico
            pico=simout(i);
            pos=i;
        end
    end
    muestras(j,:)=simout(pos-100:pos+100);
    k=k+200;
    pico=-10;
end

xica=muestras(2:136,:);
Raa=RRaa(1:200);

for i=1:135, plot(xica(i,:), hold on, end

%% aplicacion de ICA

%numeros de IC a considerar
nic=5;

%corre la libreria fastica reduciendo la dimensión de xica a 5 (lasEig)
%y tomando nic componentes independientes
[IC_fastica] = fastica(xica(1:135,:), 'lastEig', 5, 'numOfIC', nic);

clear A_actual && A_norm && invIC && data_train && clas_train

%obtengo la invIC para poder calcular A
invIC=transpose(IC_fastica)*inv(IC_fastica*transpose(IC_fastica));

%saco la matriz A media
nm=10;
```

```

matrizA=zeros(nm,nic);
A_media=zeros(1,nic);

for i=1:nm
    matrizA(i,:)=xica(i,:)*invIC;
end
for i=1:nic
    A_media(i)=mean(matrizA(:,i));
end

%Se extrae los pesos de A para cada latido nuevo luego de centrar la
señal
%dsp se dividen los pesos por A media para tener las características
A_actual=zeros(200,5);
A_norm=zeros(200,5);

for i=1:length(muestras(:,1))
    lat_cen=muestras(i,:)-mean(muestras(i,:));

    A_actual(i,:)=muestras(i,:)*invIC;

    A_norm(i,1:nic)=A_actual(i,:)./A_media;
end

%creo las matrices de entrenamiento de SVM
data_train=[A_norm Raa]; % Racp(10:209)' Ranp(10:209)';
clas_train=ones(length(data_train(:,1)),1);

%% entrenamiento SVM

%se utilizan los siguientes parametros
%tipo de svm (-s) - SVM de una clase(2)
%tipo de nucleo (-t) - RBF
%nu=0.1, gamma=0.3, epsilon=0.001
cmd = ['-s 2 -t 2 -n 0.1 -g 0.3 -e 0.001'];

%se entrena la red y se extrae el modelo
modelo = svmtrain(clas_train,data_train, cmd);

%extraemos los vectores de soporte para la clasificacion
sv=full(modelo.SVs);
sv_coef = modelo.sv_coef;
rho=modelo.rho;

```