

Modelos explicables de detección de arritmias cardiacas

**Francisco Javier
Albarrán González**

Máster de Ciencia de Datos
TFM - Área 5 – Sector Salud

Tutor/a de TF

Carlos Luis Sánchez Bocanegra

**Profesor/a responsable de
la asignatura**

Albert Solé Ribalta

2 de enero de 2024



Esta obra está sujeta a una
licencia de Reconocimiento-
NoComercial-SinObraDerivada
[3.0 España de Creative
Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

Ficha del Trabajo Final

Título del trabajo:	Modelos explicables de detección de arritmias cardiacas
Nombre del autor/a:	Francisco Javier Albarrán González
Nombre del Tutor/a de TF:	Carlos Luis Sánchez Bocanegra
Nombre del/de la PRA:	Carlos Luis Sánchez Bocanegra Carlos José Jaramillo Gómez
Fecha de entrega:	01/2024
Titulación o programa:	Máster Universitario en Ciencia de Datos
Área del Trabajo Final:	Medicina – Análisis de electrocardiogramas para detectar arritmias
Idioma del trabajo:	Castellano
Palabras clave	Electrocardiogramas, arritmias, interpretabilidad de modelos

Resumen del Trabajo

Los electrocardiogramas (ECG) son fundamentales en la medicina actual, con al menos 300 millones realizados anualmente a nivel global. Estos registros ofrecen una valiosa oportunidad para detectar posibles anomalías cardíacas, mejorar diagnósticos y adelantarse a la aparición de síntomas mediante un análisis minucioso de patrones complejos.

La inteligencia artificial (IA) está revolucionando la práctica médica al permitirnos procesar enormes cantidades de datos y desarrollar modelos predictivos precisos para interpretar los ECG. No obstante, enfrentamos desafíos notables en su implementación, como la validación de resultados por expertos médicos, la neutralidad de la información, la protección de la privacidad del paciente y la necesidad de una interpretación clara de los resultados de los algoritmos.

El objetivo principal no es delegar la toma de decisiones médicas a la tecnología, sino aprovechar la IA como una herramienta adicional para mejorar la práctica clínica. Es esencial extraer sabiduría de estos registros cardíacos para enriquecer el campo de la medicina. La transparencia en los diagnósticos generados por la IA es clave para generar confianza tanto en profesionales médicos como en pacientes, garantizando un uso ético y

adecuado de esta tecnología en la atención sanitaria.

Analizaremos variados enfoques y algoritmos de minería de datos con el objetivo de encontrar el modelo más preciso posible, sin ser la prioridad, pero tratando de dar una explicación comprensible que pueda validar el profesional médico, basada en atributos, de su área de conocimiento, y no en cálculos o probabilidades, de difícil explicación.

Abstract

Electrocardiograms (ECGs) are fundamental in today's medicine, with at least 300 million conducted globally each year. These records provide a valuable opportunity to detect potential cardiac anomalies, improve diagnoses, and anticipate the onset of symptoms through a meticulous analysis of complex patterns.

Artificial intelligence (AI) is revolutionizing medical practice by enabling us to process vast amounts of data and develop precise predictive models to interpret ECGs. However, we face notable challenges in implementation, such as result validation by medical experts, information neutrality, patient privacy protection, and the need for a clear interpretation of algorithm results.

The primary goal is not to delegate medical decision-making to technology but to leverage AI as an additional tool to enhance clinical practice. It is essential to derive knowledge from these cardiac records to enrich the field of medicine. Transparency in AI-generated diagnoses is key to building trust among both medical professionals and patients, ensuring ethical and appropriate use of this technology in healthcare.

We will analyze various data mining approaches and algorithms with the aim of finding the most accurate model possible. While finding the most accurate model is important, our top priority is to provide a comprehensible explanation that medical professionals can easily understand and validate based on attributes within their area of expertise, rather than relying solely on calculations or probabilities that may be challenging to explain.

Índice

1.	Introducción	1
1.1.	Contexto y justificación del Trabajo	1
1.2.	Objetivos del Trabajo	5
1.3.	Motivación personal	7
1.4.	Impacto en sostenibilidad, ético-social y de diversidad	8
1.5.	Enfoque y método seguido	8
1.6.	Planificación del trabajo	10
1.7.	Sumario y estructura de productos obtenidos	11
1.8.	Interpretabilidad. Aproximación a LIME	14
1.9.	Breve descripción de otros capítulos de la memoria	14
2.	Estado del arte	16
2.1.	Situación de la IA-ECG	16
2.2.	Métodos y tecnologías	19
3.	Modelos	23
3.1.	No supervisados	23
3.2.	Supervisados	24
3.3.	LIME - Local Interpretable Model-Agnostic Explanations	29
4.	Resultados	33
4.1.	Support_Vector_Machine	39
4.2.	Random_Forest	41
4.3.	XGBoosting	43
4.4.	IC_Max_voting	45
4.5.	IC_Stacking	47
4.6.	Red neuronal sencilla/clasificadora	49
5.	Conclusiones y trabajos futuros	51
5.1.	Sobre datos y modelos	51
5.2.	Sobre la explicabilidad	54
5.3.	Retos	56
5.4.	Trabajos futuros	58

6.	Agradecimientos	60
7.	Glosario	61
8.	Bibliografía	67
9.	Anexo 1 - Filtrado de electrocardiogramas	69
10.	Anexo 2 – Referencias de interés	70
11.	Anexo 3 – Librerías Python	71

Imágenes e ilustraciones

ILUSTRACIÓN 1- EVOLUCIÓN DE LA GENERACIÓN DE DATOS	1
ILUSTRACIÓN 2- GRÁFICAS ROC DEL ESTUDIO POR HOSPITAL	2
ILUSTRACIÓN 3-... Y ASÍ HASTA 341 DATASETS	4
ILUSTRACIÓN 4- EJEMPLO DE ELECTROCARDIOGRAMA CON 12 SEÑALES O DERIVACIONES	5
ILUSTRACIÓN 5- RITMOS CARDIACOS CONSIDERADOS EN ESTE ESTUDIO Y SU FRECUENCIA DE APARICIÓN	7
ILUSTRACIÓN 6- METODOLOGÍA CRISP-DM	9
ILUSTRACIÓN 7- ESTRUCTURA DE CUADERNOS JUPYTER - PYTHON	12
ILUSTRACIÓN 8- ESTRUCTURA DE CARPETAS DE LA ENTREGA	13
ILUSTRACIÓN 9- EXTRACTO DE "ARTIFICIAL INTELLIGENCE-ENHANCED ELECTROCARDIOGRAPHY IN CARDIOVASCULAR DISEASE MANAGEMENT" [7]	17
ILUSTRACIÓN 10- IMAGEN TOMADA DE "INTEGRATION OF NOVEL MONITORING DEVICES WITH MACHINE LEARNING TECHNOLOGY FOR SCALABLE CARDIOVASCULAR MANAGEMENT" [9]	18
ILUSTRACIÓN 11- IMAGEN DE "PERFORMANCE OF A CONVOLUTIONAL NEURAL NETWORK AND EXPLAINABILITY TECHNIQUE FOR 12-LEAD ELECTROCARDIOGRAM INTERPRETATION" [11]	20
ILUSTRACIÓN 12- ESQUEMA COMPLETO DE LA RED NEURONAL UTILIZADA PARA EL MODELO QUE UTILIZA SOLO ECGS EN "ARTIFICIAL INTELLIGENCE-ENABLED FULLY AUTOMATED DETECTION OF CARDIAC AMYLOIDOSIS USING ELECTROCARDIOGRAMS AND ECHOCARDIOGRAMS"[1]	22
ILUSTRACIÓN 13- FRAGMENTO DE ÁRBOL DE DECISIÓN DE NUESTRO DATASET-DERIVACIÓN V3	25
ILUSTRACIÓN 14- DIFERENCIAS ENTRE FUNCIONES DE ACTIVACIÓN	27
ILUSTRACIÓN 15- IMAGEN TOMADA EL LIBRO "MINERÍA DE DATOS, MODELOS Y ALGORITMOS"8 DE LA UOC PARA ILUSTRAR UN MODELO CON CASCADING Y STACKING	28
ILUSTRACIÓN 16- EJEMPLIFICACIÓN DE FUNCIONAMIENTO DE LIME	29
ILUSTRACIÓN 17- ALGORITMO LIME EXTRAÍDO DEL PAPER "WHY SHOULD I TRUST YOU?"[6]	30
ILUSTRACIÓN 18- DERIVACIÓN II DE UN ELECTROCARDIOGRAMA DEL DATASET ORIGINAL DE PHYSIONET	30
ILUSTRACIÓN 19- PREDICCIÓN DE XGBOOST (ARRIBA); PREDICCIÓN LIME, BAJO EL CÓDIGO	31
ILUSTRACIÓN 20- ESQUEMA GENERAL DE ONDAS DE UN LATIDO CARDIACO	32
ILUSTRACIÓN 21- MATRICES DE CORRELACIÓN ENTRE MISMOS ATRIBUTOS DE DIFERENTES DERIVACIONES	35
ILUSTRACIÓN 22- UMAP PARA 12DERIVACIONES	36
ILUSTRACIÓN 23- COMPARATIVA DE VARIABILIDAD DE RESULTADOS	38
ILUSTRACIÓN 24- INTERPRETACIÓN DE SALIDA LIME	38
ILUSTRACIÓN 25- MATRIZ DE CONFUSIÓN NORMALIZADA PARA SVM	39
ILUSTRACIÓN 26- CURVA ROC MULTICLASE Y AUC PARA SVM	40
ILUSTRACIÓN 27- PREDICCIONES DE SVM PARA UN EJEMPLO Y EXPLICACIÓN LIME	40
ILUSTRACIÓN 28- MATRIZ DE CONFUSIÓN NORMALIZADA PARA RANDOM FOREST	41
ILUSTRACIÓN 29- CURVA ROC MULTICLASE Y AUC PARA RANDOM FOREST	42
ILUSTRACIÓN 30- PREDICCIONES DE RANDOM FOREST PARA UN EJEMPLO Y EXPLICACIÓN LIME	42
ILUSTRACIÓN 31- MATRIZ DE CONFUSIÓN NORMALIZADA PARA XGBOOST	43
ILUSTRACIÓN 32- CURVA ROC MULTICLASE Y AUC PARA XGBOOST	44
ILUSTRACIÓN 33- PREDICCIONES DE XGBOOST PARA UN EJEMPLO Y EXPLICACIÓN LIME	44
ILUSTRACIÓN 34- MATRIZ DE CONFUSIÓN NORMALIZADA PARA MAXVOTING	45

ILUSTRACIÓN 35- CURVA ROC MULTICLASE Y AUC PARA MAXVOTING _____	46
ILUSTRACIÓN 36- PREDICCIONES DE MAXVOTING PARA UN EJEMPLO Y EXPLICACIÓN LIME _____	46
ILUSTRACIÓN 37- MATRIZ DE CONFUSIÓN NORMALIZADA PARA IC-STACKING _____	47
ILUSTRACIÓN 38- CURVA ROC MULTICLASE Y AUC PARA IC-STACKING _____	48
ILUSTRACIÓN 39- PREDICCIONES DE IC-STACKING PARA UN EJEMPLO Y EXPLICACIÓN LIME _____	48
ILUSTRACIÓN 40- MATRIZ DE CONFUSIÓN NORMALIZADA PARA RED NEURONAL CLASSIFICADORA _____	49
ILUSTRACIÓN 41- CURVA ROC MULTICLASE Y AUC PARA CLASIFICADOR RED NEURONAL _____	50
ILUSTRACIÓN 42- CORRELACIONES EXISTENTES INTER-DERIVACIONES PARA DURACIÓN MEDIA DE LATIDOS Y NÚMERO DE LATIDOS _____	51
ILUSTRACIÓN 43- COMPARATIVA DE MATRICES DE CONFUSIÓN _____	52
ILUSTRACIÓN 44- COMPARATIVA DE RESULTADOS LIME PARA ECG JS00010 _____	54
ILUSTRACIÓN 45- ATRIBUTOS Y RELEVANCIA SEGÚN LIME PARA JS00010 _____	55
ILUSTRACIÓN 46- INTERÉS QUE DESPIERTA LA IA-ECG EN INTERNET _____	59
ILUSTRACIÓN 47- GRÁFICAS COMPARATIVAS DE SENSIBILIDAD-VALOR PREDICTIVO POSITIVO DEL TRABAJO DE M.VOLLMER _____	69

Índice de tablas

TABLA 1- REDES NEURONALES MÁS UTILIZADAS EN IA-ECG	21
TABLA 2- ATRIBUTOS DE TRABAJO EXTRAÍDOS DE LA SEÑAL FILTRADA	34
TABLA 3- DATOS DE PRECISIÓN/ACCURACY DE LOS MODELOS ANALIZADOS	37
TABLA 4- COMPARATIVA DE AUC MEDIA DE CADA MODELO	37
TABLA 5- MÉTRICAS DE SVM PARA CADA UNA DE LAS CLASES OBJETIVO	39
TABLA 6- MÉTRICAS DE RANDOM FOREST PARA CADA UNA DE LAS CLASES OBJETIVO	41
TABLA 7- MÉTRICAS DE XGBOOST PARA CADA UNA DE LAS CLASES OBJETIVO	43
TABLA 8- MÉTRICAS DE MÁXIMA VOTACIÓN PARA CADA UNA DE LAS CLASES OBJETIVO	45
TABLA 9- MÉTRICAS DE IC-STACKING PARA CADA UNA DE LAS CLASES OBJETIVO	47
TABLA 10- AUC POR MODELO Y CLASE – DATOS OBTENIDOS PARA LA DERIVACIÓN II	52
TABLA 11- ACCURACY POR DERIVACIÓN Y CURVAS ROC	53

1. Introducción

1.1. Contexto y justificación del Trabajo

En los tiempos actuales es tentador fusionar las Inteligencias Artificiales con casi cualquier problemática diaria, social, política, creativa, de investigación... Son grandes las expectativas que hemos depositado en esta tecnología, tan versátil y plástica, que incluso es capaz de simular el funcionamiento del cerebro mismo, capaz de aprender de la nada y capaz de obtener conclusiones de un conjunto de estímulos exteriores (los datos).

Por otro lado, el uso de la Inteligencia Artificial impulsa, para la consecución del objetivo a resolver, el nuevo paradigma actual: La búsqueda de datos.

Y es que lo uno sin lo otro, no es nada.

Si en el siglo pasado los esfuerzos informáticos se destinaban al desarrollo de máquinas más rápidas y potentes, y algoritmos “deterministas” (que debían tener todas las opciones controladas), en los últimos años con el desarrollo de Internet, el 4G y 5G, el Internet de las Cosas (IoT), con la posibilidad de capturar infinidad de datos antes inimaginables, los esfuerzos se dirigen a la captura y tratamiento de estos datos y el desarrollo de modelos estadísticos, que es lo que hemos dado en llamar Inteligencia Artificial.

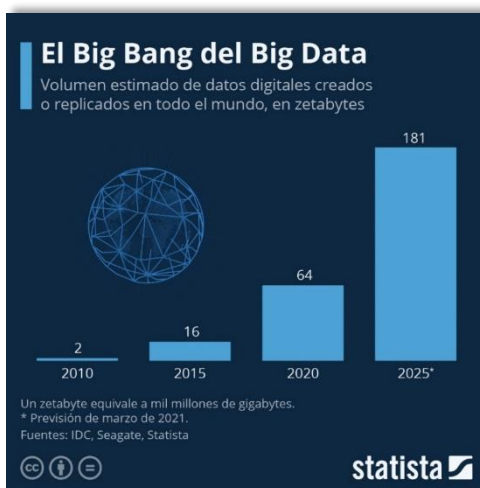


Ilustración 1- Evolución de la generación de datos

<https://es.statista.com/grafico/26031/volumen-estimado-de-datos-digitales-creados-o-replicados-en-todo-el-mundo/>

El sector médico no está al margen de esta búsqueda. Son muchas las patologías de difícil diagnóstico (problema a resolver) y son muchos los hospitales e investigadores trabajando y generando datos para “enfrentarlos” a diferentes algoritmos que arrojen luz sobre las cuestiones planteadas y les ayuden, mediante modelos matemáticos (IA), a predecir

enfermedades, mejorar los diagnósticos, etc... en definitiva, a **obtener conocimiento** y aplicarlo a su práctica médica diaria.

En los últimos años, con el advenimiento de la Inteligencia Artificial, han proliferado los estudios en diferentes especialidades médicas, con diferentes problemas y enfoques, para acercarse a este mundo del Big Data y explorar hasta dónde se puede llegar explotando los datos como nunca.

Sin embargo, no todas las especialidades presentan el mismo grado de utilización de la Inteligencia Artificial, ya sea por problemas con la financiación, por la ausencia de datos, por el escaso interés social que pueda despertar o por la baja prevalencia de las patologías a estudiar, como es el caso de la *amiloidosis* como recogen Goto, S. et al. en 2021 en el artículo “Artificial intelligence-enabled fully automated detection of cardiac amyloidosis using electrocardiograms and echocardiograms” [1]. Goto describe como a pesar de ese hándicap ya ha obtenido buenos resultados en la detección precoz de este grupo de enfermedades mediante redes neuronales alimentadas con un conjunto de *electrocardiogramas* y *ecocardiogramas* de 21.681 pacientes distribuidos en dos localizaciones (Massachusetts General Hospital –MGH- y University of California San Francisco –UCSF-). El objetivo del estudio consistía en reducir la tasa de falsos positivos, y obtuvieron resultados prometedores como se observa en las gráficas siguientes, extraídas de dicho artículo. Las gráficas muestran datos por institución, donde BWH recoge datos de prueba, no utilizados en el entrenamiento del modelo, del Brigham and Women’s Hospital de Boston.

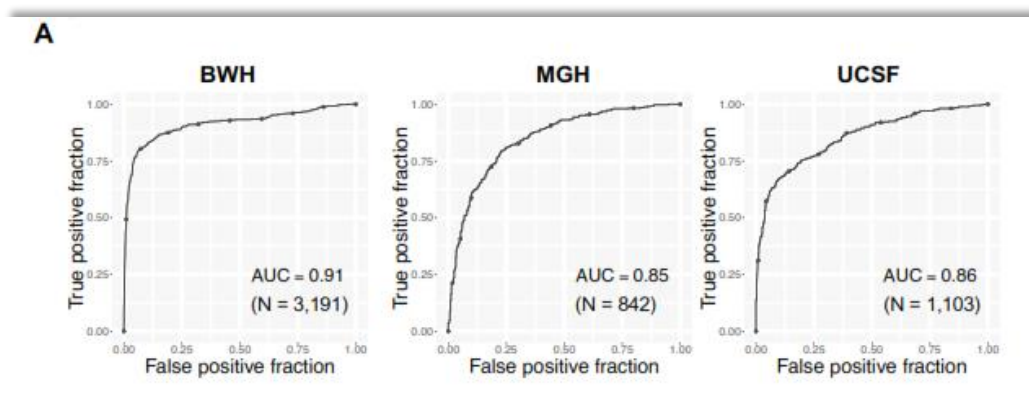


Ilustración 2- Gráficas ROC del estudio por hospital

Una medida de la bondad del modelo es el área que hay bajo la curva de manera que cuanto mayor sea, mejor será el modelo. Para BWH estamos hablando del 91% de precisión. Para lograr estos resultados se sirvieron de una red neuronal a la cual entrenaron con los datos de los electrocardiogramas, a los cuales también nos referiremos como ECG. Estas redes neuronales usadas constituyen un tipo de modelo supervisado donde, para entrenar el modelo, los datos requieren estar etiquetados con el diagnóstico que se persigue detectar. El tipo de red utilizado fue una red neuronal convolucional (CNN), ampliamente utilizadas en el procesamiento de

imágenes para la detección de patrones. Desarrollaron dos modelos, uno con ecocardiogramas, que no nos aplica, pero que nos viene a indicarnos que pueden existir tantas vías de estudio y fuentes de datos como se desee, y otro con electrocardiogramas, que es el caso que nos ocupa.

Otro hándicap relacionado con el anterior, a la hora de proceder a la implementación de modelos en el ámbito de salud es la falta de datos. Puede que sea difícil y complicada su obtención. O puede que no se haya desarrollado una metodología estandarizada, homogénea y adecuada que implique tanto a la forma de recogerlos y almacenarlos, como a su registro. Otros factores a considerar pueden ser la obtención del consentimiento de los pacientes, el establecimiento de políticas y normativas de anonimización de los datos o la variabilidad de los pacientes para evitar sesgos, haciendo que las muestras sean lo suficientemente representativas como para arrojar información y conocimiento veraz y próximo a la realidad, una vez hayan sido procesado por los algoritmos y modelos.

En el caso de los electrocardiogramas (técnica de uso muy común y no invasiva) éstos los pueden realizar distintos perfiles profesionales, pero igualmente hay que tratar todos esos componentes mencionados para poder recogerlos y tratarlos con garantías.

De la cantidad de datos se hace eco el artículo “Automated and Interpretable Patient ECG Profiles for Disease Detection, Tracking, and Discovery”^[2] aludiendo a la necesidad de grandes cantidades de datos para preservar la interpretabilidad clínica, permitir el descubrimiento del conocimiento y mantener la autonomía del médico y paciente a la hora de tomar decisiones. Este artículo, además, pone de manifiesto que la mayoría de los estudios y modelos que se implementan persiguen la **optimización del rendimiento predictivo en lugar de la propia interpretabilidad**. Y esta falta o dificultad para interpretar los resultados, podría ser el talón de Aquiles de la mayoría de los proyectos, por muy buenos resultados que obtengan. Especialmente cuando tratan temas sensibles como la salud (o la vida) de los pacientes. En estos casos creer como verdad absoluta el resultado que arroje una predicción de un modelo “*caja negra*”, se podría empezar a considerar un acto de fe, no propio de la práctica científica.

Pero como veremos más adelante, en el campo de la IA-ECG, también hay líneas de investigación encaminadas a facilitar la interpretación de las predicciones.

Hemos visto problemas. Hablemos de **los datos**.

Tan solo en la web <https://physionet.org/>^[3] descubrimos más de 300 conjuntos de datos con distintos enfoques para especialidades médicas y distintos problemas a resolver, muchos de ellos de gran calidad y de uso público.

```
#Get a list of all the PhysioNet databases available.
wfdb.get_dbs()

[[ 'aami-ec13', 'ANSI/AAMI EC13 Test Waveforms'],
  [ 'accelerometry-walk-climb-drive',
    'Labeled raw accelerometry data captured during walking, stair climbing and driving'],
  [ 'actes-cycloergometer-exercise',
    'Cardiorespiratory measurement from graded cycloergometer exercise testing'],
  [ 'adfecgdb', 'Abdominal and Direct Fetal ECG Database'],
  [ 'afdb', 'MIT-BIH Atrial Fibrillation Database'],
  [ 'afpdb', 'PAF Prediction Challenge Database'],
  [ 'aftdb', 'AF Termination Challenge Database'],
  [ 'ahadb', 'AHA Database Sample Excluded Record'],
  [ 'annotation-opioid-use-notes',
    'Annotation dataset of problematic opioid use and related contexts from MIMIC-III Critical Care Database discharge summaries'],
  [ 'antimicrobial-resistance-uti',
    'AMR-UTI: Antimicrobial Resistance in Urinary Tract Infections'],
  [ 'apnea-ecg', 'Apnea-ECG Database'],
  [ 'auditory-eeg', 'Auditory evoked potential EEG-Biometric dataset'],
  [ 'autonomic-aging-cardiovascular',
    'Autonomic Aging: A dataset to quantify changes of cardiovascular autonomic function during healthy aging'],
  [ 'bhx-brain-bounding-box',
    'Brain Hemorrhage Extended (BHX): Bounding box extrapolation from thick to thin slice CT images'],
```

Ilustración 3-... Y así hasta 341 datasets

Usamos la librería `wfdb`^[4] (<https://wfdb.readthedocs.io/en/latest/>) para el manejo de los dataset. Mediante el comando `get_dbs()` obtenemos el listado de datasets disponibles en este dominio web.

Estos datasets pueden ser explotados para entrenar distintos algoritmos. En este trabajo exploraremos los datasets de *PhysioNet* para encontrar aquel que tenga etiquetas de tipos de arritmias, que ofrezca garantías de anonimidad y consentimiento de los pacientes, y que haya contado, en su creación, con una metodología aceptable de recogida y gestión de los datos. Prepararemos esos datos para acondicionarlos a distintas técnicas de minería de datos, examinando distintos enfoques y algoritmos y analizaremos los resultados, a la búsqueda del modelo más preciso y con una mayor facilidad de interpretación y explicabilidad, característica que muchos modelos de Inteligencia Artificial adolecen.

También existen vías de investigación con datos masivos provenientes de los dispositivos wearables que muchos de nosotros utilizamos. Los smartwatches son una fuente inagotable de datos que pueden usarse para este mismo fin. Pero no existe una metodología ni una forma única y clara de extraer la información que acaba recopilándose a través de apps móviles con objetivos comerciales sin intención investigadora. Es por esto que no explotaremos esta vía, optando por entornos de obtención de los datos más controlados y de mayor calidad.

Trataremos de encontrar algún método para diagnosticar algunos tipos de *arritmia*. Utilizaremos tanto **métodos supervisados** (con registros etiquetados que nos dirán si el ECG presenta algún tipo de arritmia, para “enseñar” al sistema a buscar patrones y diferenciar) como **no supervisados**. La preferencia es evitar los modelos de *caja negra*, aunque buscaremos alguna forma de poder explicar todas las predicciones.

El presente trabajo no pretende obtener mejores resultados que un cardiólogo experimentado, pero sí una aproximación a distintos modelos y técnicas de minería de datos para ver cuál obtiene mejores resultados (mayor precisión y menor variabilidad). Buscamos modelos cuyos resultados puedan ser explicados con palabras y no con funciones matemáticas/estadísticas de difícil comprensión para la sociedad en general o herramientas que nos ayuden a interpretarlos.

Lo que se puede explicar está sujeto a la comprensión, evita dogmas, y es más fácil de implantar.

Lo que se puede explicar, arroja luz.

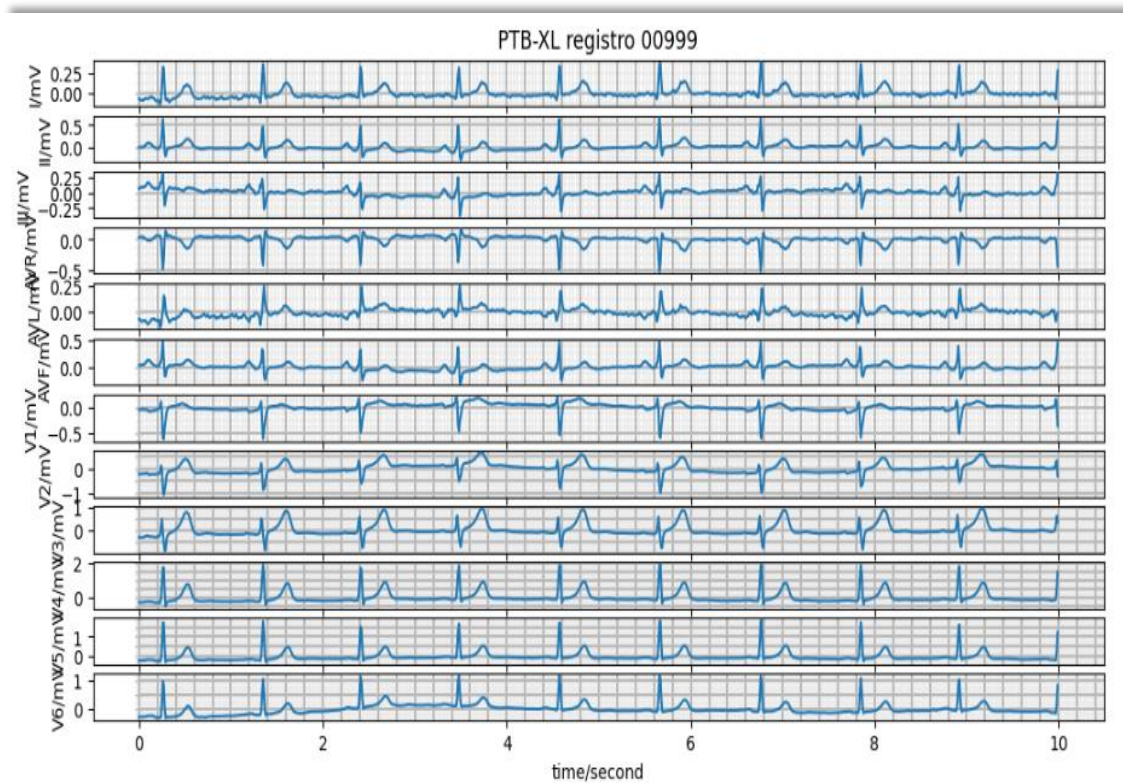


Ilustración 4- Ejemplo de electrocardiograma con 12 señales o derivaciones

1.2. Objetivos del Trabajo

Ya hemos hecho *spoiler* de nuestro principal objetivo: Hallar o construir un modelo lo suficientemente preciso como para poder contar con él, como ayuda en la práctica médica, y que pueda proporcionar explicación, de alguna manera, de los atributos que influyen en las predicciones.

¿Podemos conseguir un modelo predictivo con el cual, introduciendo un electrocardiograma y unos pocos datos más, nos detecte e identifique si existe algún tipo de arritmia, con un grado de acierto similar al de un médico, y de manera que los resultados puedan ser entendidos sin ser un erudito? Este podría ser nuestro primer y más ambicioso objetivo.

Será difícil. Aunque los datos que elijamos tengan ciertas garantías de calidad, necesitaríamos conocer y controlar todo el proceso de recogida y manipulación de la información, realizar estudios estadísticos, contrastar los resultados con médicos y ponerlo en producción, en la vida real. Y todo ello para validar nuestro modelo.

Por ello nos planteamos varios objetivos alternativos:

- 1) Hallar un modelo preciso en la detección de arritmias. En este objetivo relajamos la cuestión de la explicación de los resultados. ¿Podemos hallar un modelo predictivo que identifique tipos de arritmias en un ECG con un alto grado de precisión?
- 2) Clasificar electrocardiogramas. ¿Podemos separar, con un error mínimo, un electrocardiograma de una persona con arritmias, o alguna otra patología cardiaca, del resto de individuos? Tal vez podamos representar en un gráfico los electrocardiogramas de manera que queden agrupados aquellos que presentan patrones similares, asimilables a comportamientos cardiacos achacables a enfermedades cardiacas.
- 3) Construir una calculadora de predicción de arritmias. Con una serie de atributos de un electrocardiograma, que pueden o no venir de un electro real, determinar si existe algún tipo de arritmia y en qué probabilidad. Es decir, sin partir de un electro, con los datos, probar y experimentar combinaciones para el estudio médico de cómo se interrelacionan y en qué medida esos atributos determinan intrínsecamente un tipo de arritmia.
- 4) Elaborar un método de procesamiento automático y masivo de electrocardiogramas, desde su lectura en alguna fuente estandarizada, hasta su análisis completo y detección de patrones. Podemos aplicar todos los electros a un modelo o crear uno propio (en el fondo es el proceso principal del trabajo que presentamos).
- 5) Investigar modelos de clasificación, optimizar sus parámetros y aplicarlos al contexto médico de ECG.
- 6) Desplegar, como no puede ser de otro modo, los conocimientos del Máster, en especial de lo relacionado con la minería de datos y concretamente, su aplicación a problemas de clasificación. Podremos aplicar también otros conocimientos como *webscrapping*, Big Data, procesamiento en paralelo, construcción de pipelines, etc...

No estudiaremos todos los tipos de arritmia, lo cual daría para varios libros. Hemos seleccionado algunos tipos de arritmia muy representados y otros con escasa representatividad en el dataset original, para ver cómo gestionan los distintos algoritmos estos datos poco cuantiosos y hasta qué punto son capaces de identificarlas e influir en los resultados. En la ilustración 5 se puede observar la frecuencia de cada clase que vamos a considerar.

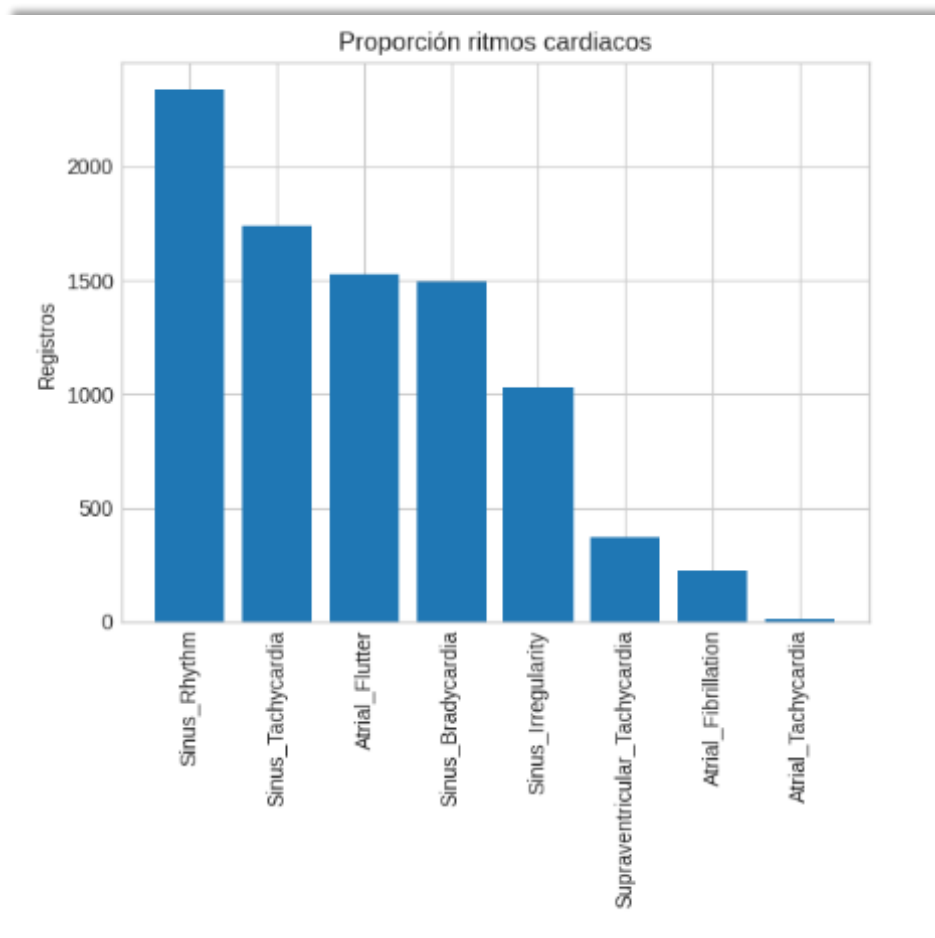


Ilustración 5- Ritmos cardiacos considerados en este estudio y su frecuencia de aparición

1.3. Motivación personal

La elección de este tema es muy personal. En junio de 2022 falleció mi padre por un infarto de miocardio, mientras quien redacta este trabajo estaba estudiando un master de Ciencia de Datos en la UOC. Entre mis asignaturas estaban “Minería de Datos” y “Modelos avanzados de Minería de Datos”. Mi padre me brindó una oportunidad en la vida con la educación que en su día me ofreció.

Creo que contribuir con lo aprendido a ayudar a los médicos a que puedan detectar más rápidamente problemas que lleven a fallos cardiacos es algo que puede alcanzarse... aunque sea poner un granito de arena, por si aporta, en esta labor que desempeñan encomiablemente un buen número de equipos e investigadores médicos y cuya labor se reconoce realmente poco, en la mayoría de los casos.

1.4. Impacto en sostenibilidad, ético-social y de diversidad

Desde un punto de vista de la sostenibilidad, el aporte del presente trabajo es evidente. Con la detección de arritmias podemos prevenir y aplicar tratamientos de manera preventiva, con menor coste, a los que habría que aplicar en casos más graves de cardiopatías. Además del ahorro en costes, esto redundará en mejor calidad de vida de los pacientes.

La componente de explicabilidad que se pretende dar a los resultados de los modelos aporta un plus en el ámbito ético-social. Por un lado, al poder explicar los resultados contribuir a la mejora de los modelos, tanto en cuanto se pueden encontrar “fallos” en la toma de decisiones/predicciones, tras la revisión de los resultados desde el prisma médico y no tras el prisma analítico/estadístico. A su vez, esto contribuye a la detección y corrección de posibles sesgos, ya que los modelos son tratados con datos de distinta proveniencia y sobre los cuales se han podido tomar decisiones que influyen en los resultados. De este modo se va depurando y garantizando un tratamiento ético de los datos y un aporte social incremental en la medida que pacientes y médicos pueden “influir” en la construcción de modelos.

Por último, dado que en ningún paso de este trabajo se van a aplicar distinciones por factores de procedencia, sexo/género, color de piel, con distintas capacidades, y mucho menos ideológicos o religiosos, todos los modelos utilizados se pueden aplicar a cualquier ECG con independencia total de la persona de la que se obtienen. Será el doctor quien, con el resultado, y conociendo el contexto del paciente (si fuma, si hace deporte, si tiene algún factor que pueda manifestarse en el electro, etc...) emita un diagnóstico clínico, objetivo y preciso para la patología detectada.

1.5. Enfoque y método seguido

El proyecto se realizará bajo el marco de trabajo [Scrum](#). Por ello, se libraron entregables semanales o cada dos semanas, dependiendo de la dificultad de cada tarea.

Cada inicio de sprint, se decidían con el tutor, en funciones de Product Owner, según la metodología SCRUM, las tareas que se incluirán en cada sprint. En cada final de sprint se explicaban los resultados obtenidos. Además, se reserva una parte de la sesión a comentar posibles mejoras en la forma de trabajar y los obstáculos que se han encontrado, en lo que vienen siendo, sesiones retrospectivas.

Para gestionar las diferentes tareas, se ha utilizado la herramienta [Trello](#), que permite trabajar con una pizarra [Kanban](#) para la descripción de los diferentes entregables y su grado de avance.

Se han ido registrando en bitácora las ideas, dudas, investigaciones, lecturas, etc... día a día. Posteriormente se comparte en la herramienta Trello, en la etiqueta Kanban adecuada, todo aquello que pueda ser susceptible de desarrollar, explorar o ampliar.

Al margen del desarrollo del proyecto con metodologías ágiles, dada la naturaleza del trabajo, emplearemos el ciclo de vida de proyectos CRISP-DM, la más habitual en proyectos de Ciencia de Datos. Este marco de trabajo es el más utilizado en este tipo de proyectos que persiguen la **obtención de valor y conocimiento** de entre un elevado conjunto de datos y se representa habitualmente, con distintos enfoques, con el esquema adjunto.

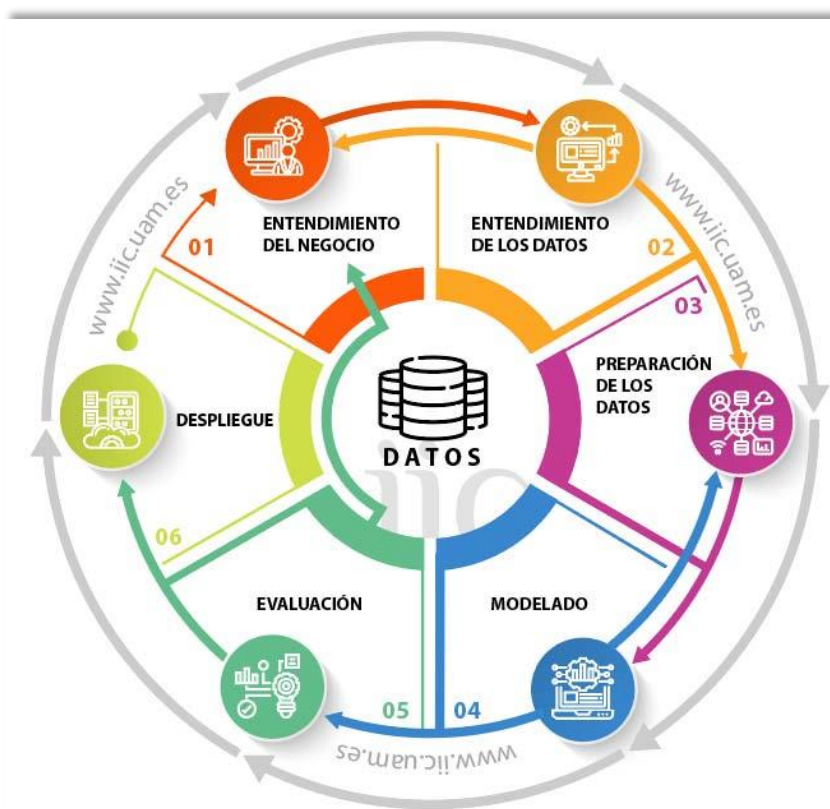


Ilustración 6- Metodología CRISP-DM

<https://www.iic.uam.es/innovacion/metodologia-crisp-dm-ciencia-de-datos/>

Esta metodología consiste, a muy alto nivel, en iterar trabajos de preparación de datos, desarrollo de modelos y evaluación de resultados hasta obtener el modelo que se ajusta a los resultados esperados sin sobre-entrenarlo, es decir, hasta que sea capaz de sacar sus conclusiones evitando que arroje resultados erróneos cuando el ECG de entrada sea sustancialmente diferente a los ECG con los que ha sido entrenado. Nos va a permitir construir una narrativa de los descubrimientos y un avance en los conocimientos a medida que avanzamos en el proyecto.

1.6. Planificación del trabajo

Como ya se ha mencionado, se ha mantenido un cuaderno de bitácora con las actividades, investigaciones, anotaciones e ideas a explorar a lo largo de todo el semestre.

El proyecto se plantea en base a sprints, propios de la metodología Agile, de carácter semanal, periodo en el que se desarrollará lo que se acuerde entre alumno y tutor. Estos desarrollos podrán consistir en bocetos de memoria, código Python, resultados de spikes (experimentos para agilizar y orientar la investigación) y todo aquello que pueda representar un avance o mejora en el trabajo y que genere valor al proyecto.

Estimamos un máximo de 14 sprints, a lo largo de 4 meses, con un esfuerzo mínimo por parte del alumno de 12 créditos (300h) que comportan la asignatura.

Las épicas, que agruparían uno o varios sprints para crear algún tipo de entregable o función completa de mayor esfuerzo al realizado en un único sprint, a grandes rasgos, podrían ser las siguientes:

Épica 1) Comprensión del problema. Contexto y definiciones preliminares. Marco de trabajo, definición de objetivos, planificación, metodología (presente documento).

Épica 2) Estado del arte. Investigaciones previas y resultados.

Épica 3) Elaborar presentación

Épica 4) EDA (análisis exploratorio de datos) y adaptación del dataset

Épica 5) Diseño de modelos supervisados

Épica 6) Diseño de modelos no supervisados

Épica 7) Memoria preliminar y revisión

Épica 8) Memoria definitiva con comentarios del tutor y mejoras

Épica 9) Preparación de la defensa

10) Defensa.

A su vez, una o varias épicas podrán componer cada una de las entregas planificadas por la UOC y recogidas en el plan docente de la asignatura y en el calendario marcado.

Módulo 1 - Definición y planificación del trabajo final (10 Oct 2023)

Módulo 2 - Estado del arte o análisis de mercado del proyecto (24 Oct 2023)

Módulo 3 - Diseño e implementación del trabajo (19 Dec 2023)

Módulo 4: Redacción de la documentación del TFM (16 Ene 2024)

Módulo 5: Defensa del proyecto (4 Feb 2024)

Puesto que vamos a trabajar con metodología Scrum, en un proyecto con **alta incertidumbre** y donde la agilidad y el aporte continuo de valor se prioriza sobre una estructura rígida de planificación, no se han utilizado diagramas de Gantt para gestionar la planificación.

1.7. Sumario y estructura de productos obtenidos

La forma de trabajar nos permite seguir una narrativa y nos condiciona el resultado, que se materializa en forma de cuaderno de Jupyter, en lo que podría ser la foto final de este itinerario iterativo en un ciclo de mejora continua de la metodología CRISP-DM utilizada.

Por una parte, se entregarán los **ficheros con los datos utilizados** en el trabajo y con los que se han entrenado todos los modelos, obtenidos de PhysioNet (“A large scale 12-lead electrocardiogram database for arrhythmia study”[\[5\]](#)), filtrados y procesados según se relata en el anexo 1.

Se acompañan **dos cuadernos de Jupyter**, como núcleo central del trabajo, con el código Python desarrollado.

El primero con la extracción, carga y procesamiento de todos los datos de todas las derivaciones sin descartar atributos. Se aporta este trabajo para consultas y obtención de datos interderivada o estadísticos. Adicionalmente nos servirá para comparar sus resultados con los del siguiente cuaderno y ver que no hay grandes mejoras en la precisión. Estamos hablando de un +2% aproximadamente de mejora en la precisión cuando utilizamos todos los atributos. Sin embargo, el coste-beneficio para los fines de este trabajo no compensan.

Y es que se hace muy complicado poder trabajar con tantísimos atributos. Manipular este conjunto de datos completo es caer en la **maldición de la dimensionalidad**, que hay que evitar en la medida de lo posible. Más datos no implica necesariamente más precisión de los modelos. Aquí sí, y por eso se aporta el cuaderno 1 con las ejecuciones de todos los modelos con todos los atributos, pero debemos ser conscientes de que una **buena elección de atributos** hace mejores y más óptimos nuestros modelos.

El segundo cuaderno se ha construido de manera en que se pueda elegir y cambiar de manera sencilla la derivación que se quiere analizar y con cuyos datos se quiere alimentar y entrenar los distintos modelos. Por ello descartaremos el resto de datos de las otras 11 derivaciones para trabajar únicamente con los datos relativos a la derivación de trabajo. Como se ha comentado y se ampliará más adelante, utilizar todos los datos extraídos no mejora sustancialmente el resultado. Se entrega el cuaderno con la ejecución completa de código para la derivación II, que es la que ofrece mejores resultados de media (+2% aproximadamente respecto a otras derivaciones) cuando ejecutamos el cuaderno para cada derivación por separado.

La estructura de los dos cuadernos es la siguiente:

<ol style="list-style-type: none"> 1. Introducción 1.1. Carga de datos y descripción 1.2. Análisis exploratorio visual y estadístico 1.3. Correlaciones 2. Modelos No Supervisados 2.1. Reducción de dimensionalidad UMAP 2.2. DBSCAN 2.3. K-means 2.4. Dendógramas 3. Modelos Supervisados 3.1. Modelo probabilístico - Naive Bayes 3.2. k-NN 3.3. Máquinas de Soporte Vectorial - SVM 3.4. SGDClassifier - stochastic gradient descent 3.5. Árboles de decisión 3.6. Regresión Logística 3.7. UMAP Supervisado 	<ol style="list-style-type: none"> 4. Modelos complejos, híbridos o transformers 4.1. Random Forest 4.2. XGBoost 4.3. Gradient Boosting 4.4. Ensemble Learning - UMAP+SGD 4.5. Red Neuronal - clasificador tabular ad-hoc 5. Inteligencia Colectiva 5.1. Max Voting 5.2. Stacking de modelos 6. Comparativas 7. Intepretabilidad de modelos 5.1. LIME - Local Interpretable Model-Agnostic Explanations 5.2. SP-LIME - Interpretabilidad global de modelos
---	---

Ilustración 7- Estructura de cuadernos Jupyter - Python

A su vez, cada apartado de los modelos supervisados se estructura de la siguiente manera:

- **Aproximación** al modelo clasificador (con una ejecución tipo para ver cómo responde a nuestros datos).
- **Búsqueda de hiperparámetros**. Normalmente estos chunks de código estarán comentados. El consumo computacional es elevado por lo que nos hemos quedado con los mejores, que pasan al siguiente chunk con una nueva ejecución más optimizada.
- **Ejecución optimizada**. Se muestran datos de precisión media, matriz de confusión y matriz normalizada.
- **Curva ROC y AUC** para cada clase en el modelo, enfrentando sensibilidad y especificidad, o lo que es lo mismo, la tasa de verdaderos positivos frente a falsos positivos. Será nuestra métrica por defecto, suficiente para los objetivos que perseguimos, si bien contaremos con otras para obtener conclusiones.
- **Calculadora de prueba** en la que se puede seleccionar un elemento de entre los 45.000 que hay en el dataset original y obtener tanto el valor real de la etiqueta “target” con el tipo de ritmo cardiaco facilitado por médicos especialistas como la predicción el modelo... y comparar.
- Por último, dentro de cada modelo, invocaremos a nuestro **interpretador de resultados**, LIME, que describiremos a continuación, para que nos devuelva, al menos de manera local, sin comprobar la totalidad de los datos y del modelo, los atributos clave que se han tenido en cuenta para la predicción y cómo han intervenido. Nos da también una estimación propia, que no tiene por qué coincidir con la del modelo que estamos explorando.

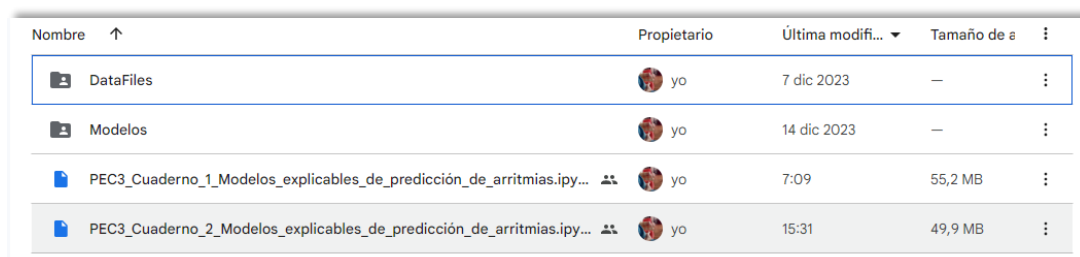
Los cuadernos finalizan con una comparativa de los modelos entrenados en cada uno de ellos, mostrando de manera centralizada alguna métrica utilizada para valorar la bondad de los

mismos y un último apartado centrado en LIME. En este apartado podemos obtener, indicando el modelo que queramos, la aportación de los atributos principales en cada clase objetivo (tipo de ritmo cardiaco), y no solo los de la predicción del modelo. También podemos ejecutar LIME para buscar explicaciones más globales del modelo que queramos analizar, pero su estudio queda fuera del alcance de este trabajo. Dejamos estas dos ideas, por si aporta, para futuros análisis.

Todos los modelos se han exportado para su posterior carga y explotación, si así se desea. Se ha implementado un chunk de volcado para cada modelo y, al final del cuaderno, otro fragmento de código comentado para recargar el que queramos, a voluntad del lector. Se entregan los **ficheros con el dump de los modelos**.

Los dos cuadernos con el código fuente (.ipynb), los ficheros csv con los 8730 electrocardiogramas tratados (carpeta “DataFiles”) y los modelos exportados (carpeta “Modelos”) son accesibles de manera pública en la siguiente URL.

<https://drive.google.com/drive/folders/1beeHDvRdGQ8wjtf4q3B0-s50d4jI4VCX?usp=sharing>



Nombre	Propietario	Última modifi...	Tamaño de a
DataFiles	yo	7 dic 2023	—
Modelos	yo	14 dic 2023	—
PEC3_Cuaderno_1_Modelos_explicables_de_predicción_de_arritmias.ipynb	yo	7:09	55,2 MB
PEC3_Cuaderno_2_Modelos_explicables_de_predicción_de_arritmias.ipynb	yo	15:31	49,9 MB

Ilustración 8- Estructura de carpetas de la entrega

Todos los entregables son accesibles sin necesidad de logarse, sin embargo, para poder visualizar correctamente los cuadernos de Jupyter es necesario tener una cuenta de Google o G-Suite de Google. Se puede acceder y visualizar en Google Colab con cuentas de la UOC sin problema.

En todo caso, los ficheros son descargables para poder ser abiertos en local con cualquier editor que este tipo de documentos como, por ejemplo, Visual Studio IDE.

Si hay algún problema con el acceso, copie la URL directamente en el navegador.

Adicionalmente se ha creado un repositorio GitHub, de acceso público, con los cuadernos Jupyter que contienen el código Python. En este repositorio no se encuentran las exportaciones de los modelos ni algunos de los ficheros csv tratados por exceder el límite del tamaño de ficheros, actualmente en 100Mb. Si bien todo este contenido no es necesario para comprobar resultados, si pudiera serlo de cara a la reproductibilidad de los mismos, por lo que se recomienda su consulta, ejecución y visionado en Google Colab.

<https://github.com/JAlbarn/ExplicabilidadModelosArritmia>

1.8. Interpretabilidad. Aproximación a LIME

Como parte relevante de presente trabajo, y dentro de la dinámica metodológica empleada, está la explicabilidad de los resultados de cada modelo, más allá de un porcentaje, un cálculo “difuso”, si se me permite la expresión, o un algoritmo que no todo el mundo conoce (o entiende).

En el paper “Why Should I Trust You?” [6] de Ribeiro M. et al. nos plantea la duda de hasta qué punto los algoritmos son válidos y si desde la revisión de los resultados, podemos inferir mejoras futuras en su cálculos. Pero para estas inferencias debemos poder identificar cuáles han sido los atributos más relevantes para la predicción, tanto si fue acertada como si no, y con ojo humano experto, poder dar la validez a dicho resultado.

Más adelante describiremos LIME. De momento indicar que, para cada modelo se ha implementado una instancia de este “meta-modelo”, que se materializa en una ejecución de ejemplo con un electrocardiograma. El objetivo es poder contrastar a nivel local, es decir, comparando elementos próximos a nuestra instancia, la predicción del modelo en estudio y la visión local de LIME. También obtendremos el peso o “relevancia” de los atributos que han influido en la decisión de cada modelo trabajado.

1.9. Breve descripción de otros capítulos de la memoria

En los siguientes apartados vamos a introducirnos en la IA-ECG, es decir, en la inteligencia artificial aplicada al estudio de los electrocardiogramas, ámbito de nuestro trabajo.

Seguiremos con los principales modelos de minería de datos, describiéndolos brevemente. Comentaremos los modelos construidos ad-hoc para el trabajo mediante “*ensemble-learning*” o “*Inteligencia Colectiva*”.

A continuación, daremos paso a un apartado con los resultados de los mejores modelos utilizados donde mostraremos la matriz de confusión normalizada obtenida, métricas, gráficas AUC de cada clase/arritmia y una predicción de ejemplo, con la interpretación del resultado mediante LIME que veremos a más adelante y que va a ser nuestra herramienta de explicabilidad, más allá del propio funcionamiento de los modelos, sus reglas y algoritmos, centrándonos en los atributos más relevantes que influyen en la predicción.

Finalizaremos la exposición con un capítulo dedicado a conclusiones y trabajos futuros, donde podremos presentar y abrir un espacio de debate sobre enfoques y el camino a seguir en el mundo de la medicina y la IA.

Tras las conclusiones nos encontraremos con un amplio glosario de términos utilizados, tanto de carácter médico como de Ciencia de Datos, la bibliografía (referenciada en la presente memoria, pero también la consultada a lo largo de todo el proceso creativo y de investigación), agradecimientos y librerías Python de tratamiento de señales utilizadas y poco conocidas fuera del ámbito de señales biológicas.

2. Estado del arte

2.1. Situación de la IA-ECG

Como ya venimos apuntando, son numerosas las aplicaciones de estas tecnologías en el ámbito científico y médico, donde una vez identificado el problema/objetivo, podríamos asegurar que la calidad de los datos, por la propia naturaleza de las investigaciones, estaría asegurada. **Equipos de precisión, profesionales cualificados, procedimientos estandarizados** hacen que los datos, materia prima de la investigación, sean cuanto menos “válidos”. Aunque también, como ya se ha señalado, hay que prestar atención a lo que llamaríamos “*metainformación*” para evitar sesgos que influirían negativamente en los resultados.

Ya hemos hablado de la detección de *amiloidosis* con electrocardiogramas y/o ecocardiogramas del artículo de Goto S. et al. [\[1\]](#) como primer ejemplo de aplicación.

Dentro de las aplicaciones donde se están destinando muchos esfuerzos y que tengan que ver con enfermedades cardíacas y con el uso de electrocardiogramas, podemos destacar la detección temprana de disfunción sistólica del ventrículo izquierdo, que es una enfermedad silenciosa que no presenta síntomas. O incluso aquellos con función ventricular normal que están en riesgo de insuficiencia cardíaca. Algunos logros son recogidos por Konstantinos C. Siontis, et al en su artículo de 2021 [\[7\]](#), y llevados a cabo en la Clínica Mayo, referente entre los entes investigadores y divulgadores, donde ya disponen de un modelo probado con CNN para este fin.

En este mismo artículo, “*Artificial intelligence-enhanced electrocardiography in cardiovascular disease management*” [\[7\]](#) también se hace referencia a otras aplicaciones como la detección de fibrilación auricular silente a partir de un ECG de ritmo sinusal, la detección de miocardiopatía hipertrófica o la mejora en la gestión de medicamentos antiarrítmicos como la dofetilida o el sotalol, permitiendo un monitoreo de los pacientes más eficaz, con menos riesgo de efectos tóxicos o incluso con ECG de una sola señal obtenida mediante smartwatches, que es una vía que comentaremos más adelante dentro del estado del arte de las cardiopatías y la inteligencia artificial, para la investigación y la recopilación de información.

Reseñable es que el uso de estas técnicas ha sido probado también con éxito para la detección del COVID-19 ayudando a salvar vidas.

Ofrece el artículo un interesante abanico de nuevos usos futuros que ofrecen amplias posibilidades y grandes expectativas de mejora en el diagnóstico y en la práctica médica.

- Atrial fibrillation
 - Beyond the CHA₂DS₂-VASc score — electrocardiogram (ECG) markers of stroke risk integrated with electronic health record-based clinical information, imaging and biomarkers
 - Rhythm-pattern recognition (implantable and wearable cardiovascular electronic devices) to predict atrial fibrillation events and artificial intelligence-enhanced ECG (AI-ECG)-guided 'pill-in-pocket' (oral anticoagulation or class Ic antiarrhythmic drugs) in patients with paroxysmal atrial fibrillation
 - Smartphone-based surveillance of the QT interval for patients in whom treatment with sotalol, dofetilide or other medications that can affect repolarization is initiated or for patients longitudinally treated with sotalol, dofetilide or other medications that can affect repolarization
- Sudden cardiac death: identifying which patients should receive an implantable cardioverter-defibrillator and integration with clinical and imaging markers
 - Beyond the left ventricular ejection fraction in patients with ischaemic or non-ischaemic cardiomyopathy
 - Risk stratification in patients with hypertrophic cardiomyopathy or inherited arrhythmia syndromes
- Predicting the exacerbation of congestive heart failure on the basis of continuous ECG ambulatory data to prevent emergency department visits and hospitalizations
- Severity and staging of heart failure, valvular heart disease and pulmonary arterial hypertension to guide clinical decision-making, prognosis and monitoring
- Angina and stress ECG analysis — identifying high-risk patients who will benefit from invasive coronary evaluation and possible intervention

Ilustración 9- Extracto de "Artificial intelligence-enhanced electrocardiography in cardiovascular disease management" [7]

Detección del consumo de diferentes drogas que pueden inducir arritmias, por la manera como alteran el ritmo cardíaco y las señales de un electrocardiograma, han sido tratado por Edi Prifti en otro artículo de 2021, mediante el uso de Deep Learning [8].

El universo de intervención de la IA-ECG es enorme. El entorno es propicio como indicábamos al inicio de este apartado y los campos de conocimiento son numerosos.

Por otro lado, se abren líneas nuevas de investigación gracias a la recogida de información a través de dispositivos wearables, tales como smartwatches, bandas, calzado, etc conectados a Internet. Es el internet de las cosas (IoT) de donde podemos extraer información prácticamente ilimitada.

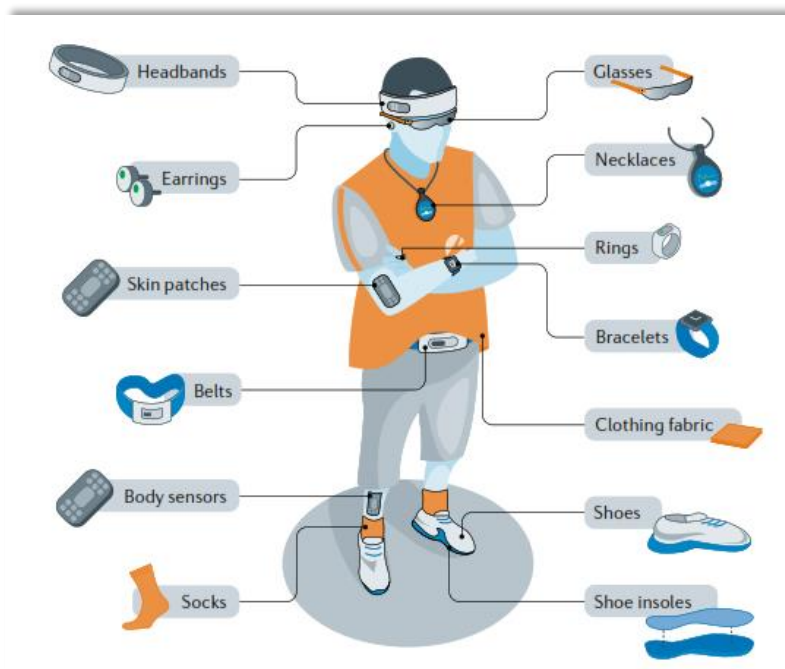


Ilustración 10- Imagen tomada de "Integration of novel monitoring devices with machine learning technology for scalable cardiovascular management"

[9]

Si bien los dispositivos IoT no tienen la capacidad y exactitud de los aparatos médicos, ni el proceso de recogida de datos sigue una metodología adecuada, hay sólidas líneas de investigación. A modo de ejemplo citaremos dos casos de uso que se recogen en el artículo de Konstantinos C. Siontis y su equipo [7]:

- Se ha conseguido una alta precisión para detectar una baja fracción de eyección del ventrículo izquierdo con ECG de una sola señal, por lo que su aplicación en el uso de teléfonos inteligentes es muy posible. No podemos decir que abra el camino, pero sí abre nuevas líneas de investigación en el desarrollo de sistemas de captura confiable de información y el desarrollo de modelos predictivos para alertar de manera temprana de esta patología.
- Alertas de fibrilación auricular mediante datos recogidos por fotopleletismografía, técnica que, sin estar basada en pulsos eléctricos como el ECG, puede utilizarse para la monitorización periférica de la frecuencia cardíaca. De hecho, ya se ha desarrollado una red neuronal para detectar la fibrilación auricular de forma pasiva, es decir, sin intervención médica y de manera automática, a partir de estas señales luminosas obtenidas en smartphones, concretamente, Apple Watch. La amplia penetración en la sociedad de este tipo de dispositivos puede suponer una mejora en las condiciones de vida de personas susceptibles o propensas a tener algún tipo de fibrilación auricular.

2.2. Métodos y tecnologías

En cuanto al estado del arte de las tecnologías utilizadas, el abanico también es amplio. El conjunto de técnicas de minería de datos, algoritmos, redes neuronales, etc... son ampliamente versátiles y plásticos como para adaptarse a cualquier tipo de problema. Ya sea de **clasificación** de enfermedades, como de **predicción**. Ya sea para **descubrir** como para **inferir** hipótesis.

La mayoría de los estudios médicos en IA-ECG utilizan **modelos supervisados**. Esto significa que a cada registro de información le acompaña una o varias etiquetas marcando el diagnóstico dado por especialistas. Con estos datos, y sus correspondientes etiquetas, se entrenan los modelos y algoritmos que obtienen patrones. Estos patrones, basados en modelos estadísticos, son los que quedarán almacenados en el modelo al cual se le proporcionan nuevos datos para que “pronostiquen” su diagnóstico. Si el modelo es suficientemente bueno, se pone en práctica con datos del mundo real, pero siempre bajo la supervisión médica.

Como se ha comentado anteriormente, **la tendencia de todos estos modelos es a mejorar u optimizar el porcentaje de aciertos, muchas veces en detrimento de la explicabilidad**, asunto importante para que estos modelos puedan implantarse con seguridad y con la confianza apropiada por parte del personal médico y de los pacientes.

En algunos casos la etiqueta “de control” con la que se entrenan los modelos, es decir, el diagnóstico para un determinado ECG, proviene de pruebas externas, como en el caso del estudio de Goto S.1 y su equipo para la detección de la amiloidosis donde se realizaban pruebas para confirmar la enfermedad mediante biopsia de tejido, exploración de medicina nuclear, resonancia magnética cardiaca o pruebas genéticas. Si todo esto no era factible, el ECG se etiquetaba en función de las propias características del propio ECG, porque cada uno de ellos debe tener su etiqueta de diagnóstico.

En el modelo desarrollado por Awni Y. Hannun et al. [\[10\]](#) para detectar arritmias en ECG, estas etiquetas incluían distintas clasificaciones con mayor o menor detalle del diagnóstico y eran proporcionadas por cardiólogos experimentados de manera individual, pero también lo que denominaron “estándar de oro” que consistía en el diagnóstico de un comité de cardiólogos; todo ello para poder entrenar con información veraz el modelo y para determinar diferentes grados de precisión. Lo llamativo de este estudio, además del enorme despliegue de personal especializado, es que los datos de los ECG utilizados solo tenían una señal en lugar de las 12 habituales (al estilo de los trabajos comentados con smartwatches) por lo que se disponía de 12 veces menos de información que si el ECG estuviera completo. Pero, por el contrario, el modelo podría aplicarse con ECG más sencillos y por lo tanto más baratos y de más fácil instalación, mantenimiento y manejo. Y por qué no, instalables en futuros wearables como los que ya hemos visto anteriormente.

Existen muchos más modelos desarrollados por diferentes equipos, con distintos enfoques y distintos problemas médicos (aunque en el fondo se podrían reducir a uno, “**predicción**”), y

todos utilizan datos etiquetados para mostrárselos a los algoritmos, para que identifiquen patrones que puedan considerarse únicos de una patología y, por tanto, asimilarlos a un diagnóstico.

En “Automated and Interpretable Patient ECG Profiles for Disease Detection, Tracking, and Discovery” [2], se restringen los datos solo a ECGs con ritmo sinodal normal, para facilitar el desarrollo del modelo, y en “Deep learning analysis of electrocardiogram for risk prediction of drug-induced arrhythmias and diagnosis of long QT syndrome” del ya mencionado Edi Prifti [8], el estudio se basa en cohortes de sujetos con distintas casuísticas, pero también, en ambos casos, los datos están etiquetados y el modelo de IA desarrollado es, por tanto, supervisado.

Señalar que, a todas luces, un modelo no supervisado solo permitiría clasificar, es decir, agrupar ECG similares de forma que no sabríamos, a priori, si los diferentes corresponden a una enfermedad u otra, o son otro tipo de pulsos cardiacos correspondientes a una persona sana. Su uso estaría más restringido a investigación y extracción de conocimiento de la información que arrojan los ECG, que a la predicción. Solo un ojo experto podrá señalar a qué se corresponde cada agrupación de electrocardiogramas.

Merece una mención especial el trabajo realizado por el equipo de J. Weston Hughes titulado “Performance of a Convolutional Neural Network and Explainability Technique for 12-Lead Electrocardiogram Interpretation” [11] donde, además de la precisión, se ha perseguido la tan ansiosa explicabilidad utilizando ECG de 12 derivaciones para entrenar una red convolucional (CNN), obteniendo un rendimiento comparable al de los cardiólogos en numerosas clases de diagnóstico de las 38 examinadas y distribuidas en 5 categorías, pero, y aquí está lo novedoso, implementando una técnica llamada LIME (Linear Interpretable Model-Agnostic Explanations) que permite resaltar fragmentos de ECG que son relevantes para el diagnóstico de la red neuronal. Es decir, no solo predice con gran precisión, sino que marca las partes del electrocardiograma que apoyan su diagnóstico/predicción para la revisión por parte del especialista, que podrá dar una explicación “con palabras” al paciente y/o verificar si lo que arroja el modelo tiene sentido o no.

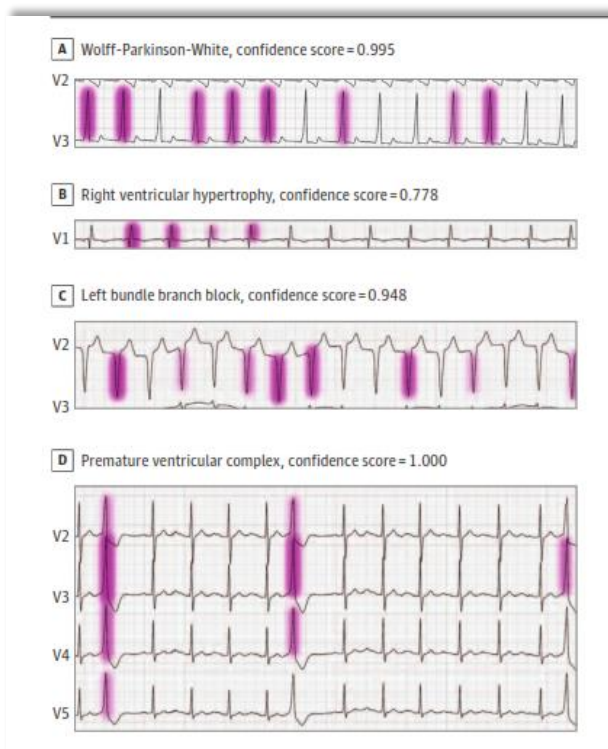


Ilustración 11- Imagen de “Performance of a Convolutional Neural Network and Explainability Technique for 12-Lead Electrocardiogram Interpretation” [11]

con varios ejemplos de diagnósticos, con el nivel de confianza, señalando las zonas con alteraciones

Y si el rey entre los modelos son los modelos supervisados, la técnica estrella, usada mayoritariamente (aunque en algunos casos complementada con otras) son las redes neuronales convolucional a las que ya nos hemos referido anteriormente. Gracias a este tipo de redes, podemos descubrir la esencia de los datos en el eje horizontal (combinando información de más de un punto de tiempo, pero dentro de una derivación) y en el vertical (combinando información de muchas derivaciones, pero en un solo punto de tiempo) o combinando la información de ambos ejes.

Sus ventajas son la facilidad de implementación o la capacidad de trabajar con imágenes para la extracción de patrones e información relevante de manera eficiente, pero, por el contrario, funciona como una caja negra que impide saber el porqué de su predicción, y esto, que no deja de ser una gran desventaja, puede cuestionar sus resultados en el mundo real, aunque la tasa de aciertos en sus diagnósticos sea apreciable.

En la siguiente tabla podemos ver los tipos de redes neuronales utilizados en los artículos mencionados y que han servido de inspiración y referencia para la confección de este trabajo.

Artículo de referencia	Tipo de Red Neuronal
Artificial intelligence-enabled fully automated detection of cardiac amyloidosis using electrocardiograms and echocardiograms [1]	2D-CNN based model
Artificial intelligence-enhanced electrocardiography in cardiovascular disease management [7]	CNN
Automated and Interpretable Patient ECG Profiles for Disease Detection [2]	CNN-HMM
Deep learning analysis of electrocardiogram for risk prediction of drug-induced arrhythmias and diagnosis of long QT syndrome [8]	CNN + t-SNE + k-means

Tabla 1- Redes neuronales más utilizadas en IA-ECC

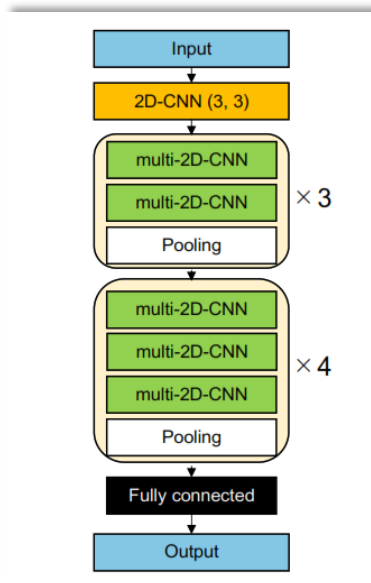


Ilustración 12- Esquema completo de la Red Neuronal utilizada para el modelo que utiliza solo ECGs en "Artificial intelligence-enabled fully automated detection of cardiac amyloidosis using electrocardiograms and echocardiograms"[\[1\]](#)

El hecho de que podamos combinar la información en distintos ejes y distintas capas como hacen las redes CNN nos lleva a un posible salto tecnológico, por explorar en próximos años, con el desarrollo reciente de los **LLM (Large Language Model)** y **las inteligencias generativas** que parten del paper titulado "Attention is all you need" [\[12\]](#) y que propone el tratamiento de información en diferentes capas para obtener información de distintas formas y con distintos

enfoques, analizando cada dato "tokenizado" con su entorno con distintos puntos de vista. Este tipo de modelos, utilizados ampliamente en el procesamiento de lenguaje natural (NLP), podrían utilizarse de manera similar con los ECG. Por hacer una analogía, los ECG serían las frases/oraciones del lenguaje natural que se procesarían mediante LLM para obtener una semántica propia del "lenguaje de los electrocardiogramas", descubriendo que "palabras", o rasgos en el ECG, podrían encajar mejor a continuación, de modo que los ECG anómalos pudieran ser identificados.

3. Modelos

A continuación, se describirán brevemente los modelos utilizados, su funcionamiento y sus aplicaciones en técnicas de Inteligencia Artificial o Machine Learning. Seguiremos el orden en que se muestran en el segundo cuaderno de Jupyter, y que es el mismo en que fueron introducidos cuando describíamos anteriormente el itinerario trazado.

3.1. No supervisados

UMAP

UMAP (Uniform Manifold Approximation and Projection) es un algoritmo de reducción de la dimensionalidad y clustering de datos para entornos de alta dimensionalidad, útil para su representación y visualización. Calcula la distancia entre todos los puntos en el espacio original y mediante proyecciones va reduciendo la dimensionalidad minimizando la función de costo que representa la pérdida de información debida a dichas transformaciones.

Es un algoritmo no supervisado, sin etiquetas que clasifiquen las instancias, pero se puede adaptar para trabajar como modelo supervisado.

DBSCAN

[DBSCAN](#) o "Density-Based Spatial Clustering of Applications with Noise" (Agrupamiento Espacial Basado en Densidad con Ruido). Es un algoritmo de agrupamiento que no requiere definir previamente el número de clusters como en K-Means. En lugar de eso, DBSCAN identifica áreas densas de puntos en un espacio de datos y agrupa los puntos en función de la densidad de estos. Los puntos con muchos vecinos cercanos se consideran puntos núcleo, agrupándose en un cluster los que tienen suficientes vecinos en una distancia cercana específica. Por otra parte, están los puntos frontera que están cerca de un grupo, pero no cumplen con los requisitos de densidad para ser considerados densos y poder agruparse en el grupo cercano. Normalmente se agrupan en el cluster cercano. Además, están los puntos considerados como ruido, que no se asigna a ningún cluster.

K-MEANS

[K-Means](#) es una técnica utilizada en análisis de datos para agrupar información similar basada en la proximidad a k puntos elegidos aleatoriamente llamados centroides. Cada punto se va asignando al centroide más cercano que se recalcula tomando el promedio de los datos asignados a él. El proceso se repite hasta completar la asignación de todos los datos o hasta que el grupo es estable.

DENDOGRAMAS

Un **dendrograma** es una representación jerárquica que muestra cómo los elementos o grupos se agrupan en base a su similitud o distancia, utilizando una estructura de árbol donde las ramas más cercanas representan elementos más similares entre sí.

Linkage completo (*complete linkage*): Este método calcula la distancia entre dos clusters basándose en las distancias máximas entre los puntos de los clusters. Es decir, considera la distancia más lejana entre cualquier par de puntos, uno de cada cluster, y utiliza esa distancia como medida para fusionar o unir clusters.

Linkage parcial (*single linkage*): En contraste, este método calcula la distancia entre dos clusters basándose en las distancias mínimas entre sus puntos. Toma la distancia más corta entre cualquier par de puntos, uno de cada cluster, y utiliza esa distancia como medida para fusionar o unir clusters.

3.2. Supervisados

MODELO PROBABILÍSTICO NAÏVE BAYES

El [modelo supervisado probabilístico Naïve Bayes](#) es un algoritmo de aprendizaje automático utilizado principalmente para clasificación. Se basa en el teorema de Bayes de probabilidad condicionada y asume independencia entre las características para hacer predicciones. Se basa en la observación de datos de entrenamiento etiquetados para estimar las probabilidades de que ciertas características pertenezcan a una clase específica. Estas probabilidades son las que se usarán después para realizar las clasificaciones de los datos sin etiqueta, ya sean los datos de test o nuevas instancias para emitir una predicción.

K-NN

El algoritmo [K-Nearest Neighbors](#) (K-NN) es un método de aprendizaje supervisado principalmente para clasificación. Funciona asignando una etiqueta a un punto (dato) basado en las etiquetas de los puntos más cercanos en un espacio de características. K representa la cantidad de vecinos considerados para la clasificación. Para predecir la clase de un nuevo punto, se buscan los K vecinos más cercanos y se asigna la clase que aparece más veces entre esos vecinos al nuevo punto.

MÁQUINA DE SOPORTE VECTORIAL O SVM

Este algoritmo se basa en la búsqueda de un hiperplano óptimo que separe los datos en las distintas clases de una manera efectiva. [SVM](#) puede manejar eficazmente datos no linealmente separables al transformarlos a un espacio dimensional superior a través de un kernel, permitiendo encontrar un hiperplano separador incluso en espacios no lineales. Es lo que se conoce como *kernel trick*.

SVM CON DESCENSO DE GRADIENTE ESTOCÁSTICO

[SGD-SVM](#) es una variante de SVM que utiliza el descenso de gradiente estocástico para entrenar el modelo de manera más rápida y eficiente. En lugar de calcular el gradiente de la función de pérdida utilizando todo el conjunto de datos en cada paso, SGD-SVM calcula el gradiente de la función de pérdida utilizando solo una muestra aleatoria (o un conjunto pequeño de muestras) en cada iteración. Ajusta los parámetros del modelo (los pesos y el sesgo) en la dirección opuesta al gradiente de la función de pérdida, lo que ayuda a minimizar la función de pérdida y a encontrar los mejores parámetros para separar las clases.

ÁRBOL DE DECISIÓN

Representan una estructura en forma de [árbol](#) donde cada nodo interno representa una característica, cada rama representa una decisión basada en esa característica, y cada hoja representa el resultado final, como una etiqueta de clasificación o un valor numérico en el caso de regresión. Estos modelos dividen el conjunto de datos en ramas utilizando reglas de decisión simples basadas en las características, lo que facilita la interpretación y el análisis de cómo se toman las decisiones en cada punto del árbol.

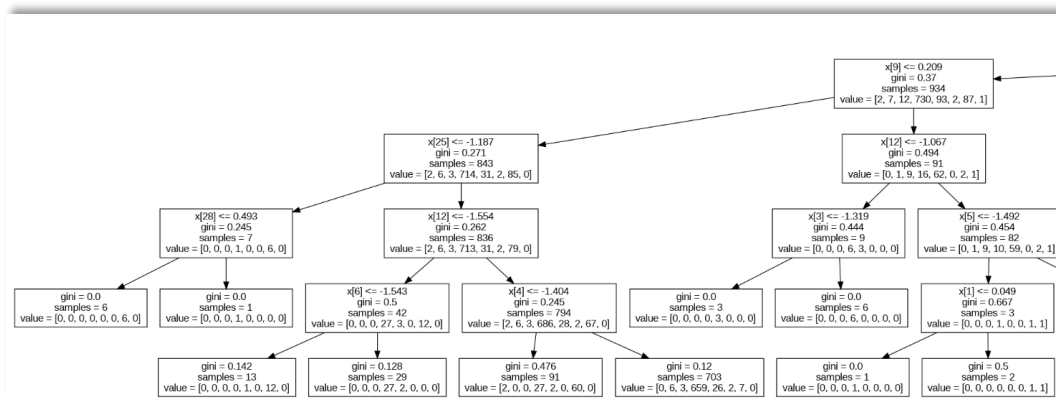


Ilustración 13- Fragmento de árbol de decisión de nuestro dataset-derivación V3

REGRESIÓN LOGÍSTICA (MULTICLASE)

En este [enfoque analítico](#), la variable dependiente es finita o categórica, ya sea A o B (regresión binaria) o una variedad de opciones finitas A, B, C o D (regresión multinomial). Trata de buscar la relación entre una variable dependiente (la variable objetivo o target) y una o más variables independientes mediante la estimación de probabilidades con una función de regresión logística. Para ello codifica los valores de la variable dependiente y, mediante el entrenamiento, se van ajustando los coeficientes de esa función, minimizando la pérdida (discrepancia entre predicciones de modelo y las etiquetas reales del dataset de entrenamiento).

UMAP-SUPERVISADO

UMAP-supervisado es una extensión de UMAP que incorpora información de etiquetas (supervisión) en el proceso de reducción de dimensionalidad. Este algoritmo utiliza dichas etiquetas para guiar la reducción de la dimensionalidad, intentando preservar las estructuras de grupos y las separaciones entre clases en el espacio de menor dimensión.

RANDOM FOREST

Este algoritmo se basa en la idea de construir múltiples árboles de decisión y combinar sus predicciones para obtener un resultado más robusto y preciso. Cada árbol ve una muestra distinta y aleatoria de los datos, tanto atributos como elementos para aumentar la diversidad de árboles y reducir la correlación entre los datos. Finalmente, para la clasificación, se combina la predicción de todos ellos eligiendo la clase que más puntuaciones o votos ha tenido. En caso de regresión, se promedian los resultados para obtener un resultado final.

XGBOOST

[XGBoost](#) (Extreme Gradient Boosting) es un algoritmo de aprendizaje automático basado en árboles, conocido por su eficiencia, velocidad y precisión en problemas de clasificación y regresión. Es una implementación mejorada y optimizada del algoritmo de aumento de gradiente.

Se construye un “bosque” de árboles secuenciales donde cada árbol se enfoca en corregir los errores de los anteriores. Utiliza un algoritmo de optimización de gradiente para minimizar la función de pérdida, por lo que va mejorando la precisión en cada iteración.

Además, implementa métodos para el procesamiento en paralelo, lo que permite una ejecución rápida incluso en grandes cantidades de conjuntos de datos.

GRADIENTBOOSTINGCLASSIFIER

Similar al anterior. Utiliza árboles de decisión débiles ensamblados secuencialmente de modo que la salida de unos se convierte en la entrada de los siguientes para ir mejorando la precisión a través del descenso del gradiente y minimizando la función de pérdida.

La diferencia con XGBoost reside en que mientras que [GradientBoostingClassifier](#) es parte de scikit-learn [13] y ofrece un enfoque más estándar, XGBoost es una biblioteca independiente que proporciona mejoras específicas, mayor flexibilidad y un rendimiento optimizado para problemas de aprendizaje supervisado basados en Gradient Boosting (construye un modelo predictivo mediante la combinación secuencial de múltiples modelos simples, generalmente árboles de decisión, enfocándose en corregir los errores de predicción de modelos anteriores para mejorar la precisión general del modelo).

ENSEMBLE LEARNING - UMAP + SVM CON DESCENSO DE GRADIENTE ESTOCÁSTICO

Entendemos por [ensemble learning](#) la combinación de las predicciones de múltiples modelos base (clasificadores o regresores) para mejorar la precisión y el rendimiento general del modelo final, aprovechando la diversidad y la combinación de múltiples hipótesis. Este método busca reducir el sobreajuste y mejorar la generalización al combinar las fortalezas de varios modelos para obtener una predicción más precisa y robusta.

En nuestro caso hemos compuesto un modelo clasificador utilizando UMAP como primer paso, para agrupar las instancias en un espacio de baja dimensionalidad, y un modelo de soporte vectorial, SVM, que recibe estas agrupaciones, de forma que pueda “separar” el espacio mediante la función de kernel óptima en distintas zonas, dentro de cada cual estaría cada clase de nuestro conjunto de datos.

RED NEURONAL

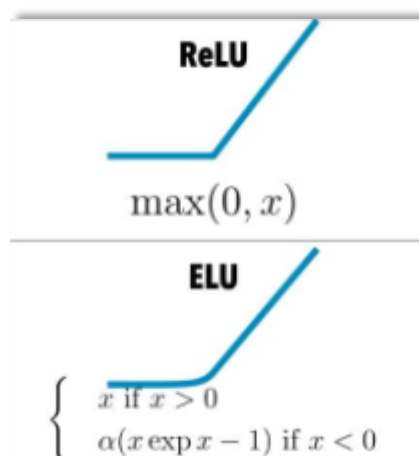


Ilustración 14- Diferencias entre funciones de activación

Una [red neuronal](#) es un modelo computacional inspirado en el funcionamiento del cerebro humano, compuesto por nodos interconectados llamados neuronas artificiales, organizados en capas, que procesan información. Estas redes aprenden a través del ajuste de pesos y conexiones entre neuronas para realizar tareas como clasificación, reconocimiento de patrones o predicción en datos, empleándose en problemas de aprendizaje automático y procesamiento de datos complejos debido a su capacidad para capturar relaciones no lineales y realizar tareas sofisticadas.

En el trabajo proponemos una red simple de tres capas y función de activación eLu, para evitar la posible inactividad de ciertas neuronas. Hemos comprobado que, para la red propuesta, obtiene mejores rendimientos que la habitual ReLu.

IC CON VOTACIÓN DE MAYORÍAS

Los modelos de aprendizaje de inteligencia colectiva se basan en la idea de que un conjunto de individuos o entidades puede trabajar en conjunto para resolver problemas, tomar decisiones o realizar tareas complejas de manera más efectiva que un solo individuo. Estos modelos se inspiran en fenómenos observados en la naturaleza, como enjambres de insectos, colonias de hormigas o bandadas de pájaros, donde la cooperación y la interacción entre individuos dan como resultado comportamientos colectivos inteligentes.

En nuestro caso específico, varios modelos individuales, todos los anteriores (excepto la red neuronal), hacen predicciones sobre un conjunto de datos y la predicción final se determina por el voto mayoritario de estos modelos.

IC STACKING DE MODELOS

El stacking o apilamiento es un método de ensamblaje que combina múltiples modelos de aprendizaje para mejorar las predicciones. A diferencia del voto mayoritario en el método Max_Voting, el stacking no solo se basa en la combinación simple de las predicciones, sino que utiliza un meta-modelo (también llamado "modelo de nivel superior" o "meta-aprendizaje") que se entrena con las salidas de los modelos base para hacer la predicción final.

De manera similar al anterior modelo, vamos a conformar la salida de los modelos sencillos anteriores como entrada para un nuevo modelo simple, una regresión logística, para obtener nuestras predicciones.

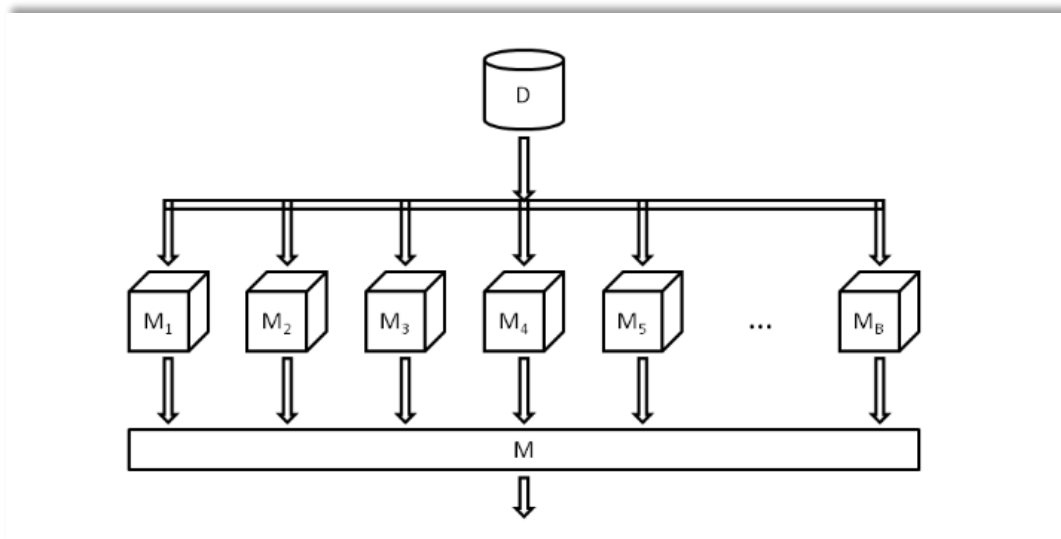


Ilustración 15- Imagen tomada el libro "Minería de Datos, modelos y algoritmos"8 de la UOC para ilustrar un modelo con cascading y stacking

3.3. LIME - Local Interpretable Model-Agnostic Explanations

Más allá de la precisión de los modelos y los estudios sesudos, está el mundo real. Los datos de entrenamiento no lo son todo y la realidad, como bien es sabido, “supera a la ficción”. Es por esto que entender el motivo por el que se obtienen algunas predicciones, es importante, sobre todo si afectan a situaciones como la salud, el trabajo o al día a día.

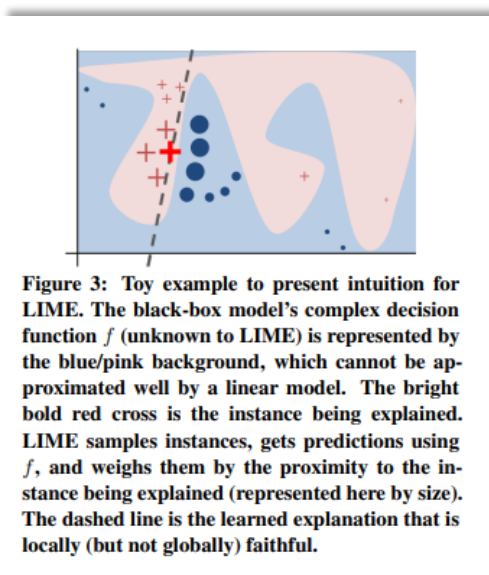


Ilustración 16- Ejemplificación de funcionamiento de LIME

LIME es capaz de explicar cómo las variables o atributos terminan afectando a la predicción. Es independiente del modelo que estemos analizando. Esto lo consigue con un enfoque local, comparando los atributos del elemento para el que queremos predecir su clase, insertado en el modelo en estudio, con aquellos próximos a él y que han sido afectados por su aparición. De esa comparación y esas variaciones puede determinar qué atributos han influido en los cambios producidos en el modelo. Utiliza un enfoque local que le posibilita discriminar de manera lineal por muy curvada que sea frontera entre clases que marca nuestro modelo (como cuando hacemos zoom a una parte de una circunferencia, que al final adopta la forma de línea recta).

Algorithm 1 LIME for Sparse Linear Explanations

Require: Classifier f , Number of samples N

Require: Instance x , and its interpretable version x'

Require: Similarity kernel Π_x , Length of explanation K

$\mathcal{Z} \leftarrow \{\}$

for $i \in \{1, 2, 3, \dots, N\}$ **do**

$z'_i \leftarrow \text{sample_around}(x')$

$\mathcal{Z} \leftarrow \mathcal{Z} \cup \langle z'_i, f(z_i), \Pi_x(z_i) \rangle$

end for

$w \leftarrow \text{K-Lasso}(\mathcal{Z}, K)$ \triangleright with z'_i as features, $f(z)$ as target **return** w

Ilustración 17- Algoritmo LIME extraído del paper "Why Should I Trust You?" [6]

Arrojará una predicción en base a ese enfoque. Y esta podrá coincidir o no con la dada por el modelo en estudio (con enfoque global). Pero nos proporcionará los atributos que han sido relevantes para la predicción de nuestro modelo y con su análisis, podremos determinar situaciones como:

- si la predicción fue acertada, aunque con atributos incorrectos
- si fue incorrecta por el uso de atributos equivocados
- si fue correcta con los atributos adecuados...

Pero en todo caso, podremos explicar y podremos dar feedback para la mejora continua de los algoritmos utilizados y mejorar su precisión y explicabilidad.

Un caso de uso de LIME, muy visual, es el descrito por J. Weston Hughes en el artículo "Performance of a Convolutional Neural Network and Explainability Technique for 12Lead Electrocardiogram Interpretation" [11] donde, con la información de los atributos relevantes, los ubica en un electrocardiograma y los marca para que el ojo experto del cardiólogo ponga su interés en esa parte del mismo como se muestran en la imagen.

Vamos a introducirnos en LIME con un ejemplo, que es como se ven mejor las cosas.

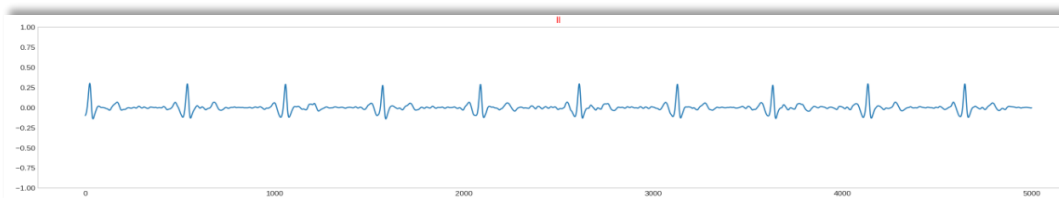


Ilustración 18- Derivación II de un electrocardiograma del dataset original de physionet

El electrocardiograma anterior está etiquetado como “Sinus Bradycardia”, lo que significa un ritmo normal, pero lento. Podría ser un deportista, pero podría ser síntoma de alguna patología, que solo el doctor, poniendo en contexto, podrá dilucidar.

Al pasar esa derivación al modelo entrenado con XGBoost, éste lo cataloga correctamente. ¿Por qué?

Veamos que nos dice LIME.

```

Diagnóstico real = 0 Sinus_Bradycardia
Name: Nombre_ritmo_cardiaco, dtype: category
Categories (1, object): ['Sinus_Bradycardia']
Predicción = Sinus_Bradycardia

#Explicación LOCAL del diagnóstico predicho por el modelo usado según LIME, por sus atributos de mayor importancia
indice_prediccion = obtener_indice_por_valor(dicc_arritmias, list(dicc_arritmias.values())[elemento])
dd=[]
dd.append(indice_prediccion)

lime = explainer.explain_instance(Pred_sin.T[0], cxgb.predict_proba, num_features= 8, labels=dd)
lime.show_in_notebook(show_table=True)
    
```

Prediction probabilities		Feature	
Sinus_Bradyc...	1.00	II Intervalo_medio_posterior	1.03
Sinus_Rhythm	0.00	II Intervalo_medio_anterior	1.03
Sinus_Irregul...	0.00	II Media_intervalos_medios	1.03
Atrial_Flutter	0.00	II Intervalo_medio_actual	1.03
Other	0.00	II Dif_media_amplitud_ST	0.08
		II Tiempo_medio_PS	0.17
		II Dif_media_amplitud_PS	-0.31
		II Ratio_medio_ampST_ampPQ	-0.19

Ilustración 19- Predicción de XGBoost (arriba); predicción LIME, bajo el código

Para LIME también es un ritmo lento, Sinus Bradycardia, marcado en azul a la izquierda. Y nos dice qué atributos han influido en la decisión de XGBoost, para su decisión (coincidente en este caso) en la columna del medio (hacia la derecha, lo que suma para ser Sinus Bradicardia, como la duración del intervalo o latido; y para que no lo sea, hacia la izquierda, como la diferencia media de amplitudes de entre la onda S y T – ver imagen siguiente-). **El sentido correcto de la influencia de cada atributo solo lo podrá dar un experto**, pero desde la distancia, en este caso, parece tener sentido... el tiempo del latido influye en esta etiqueta y las ondas S y T, últimas de cada latido, para ritmos lentos, tienen a estar atenuadas, por lo que su diferencia es menor. En la tabla derecha en azul indica el valor de esas variables para su valoración.

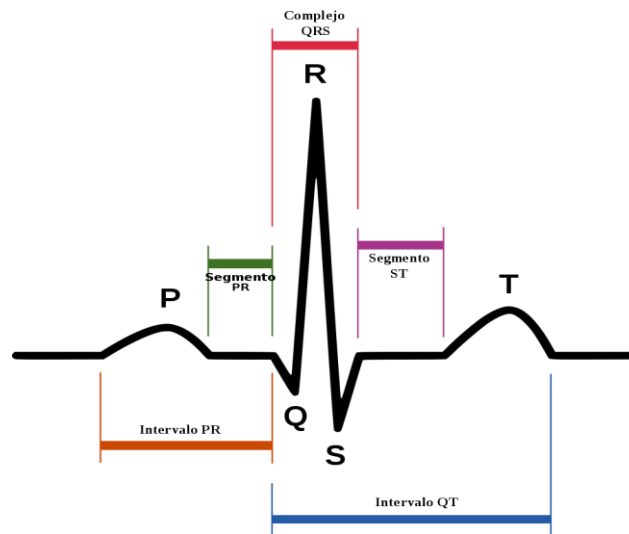


Ilustración 20- Esquema general de ondas de un latido cardiaco

Fuente: [Wikipedia](#)

Si hubiera discrepancias, ese experto tendría que ver quién tiene razón. Cuenta con los atributos relevantes para el modelo que muestra LIME como apoyo, y con todo su conocimiento, experiencia y **el contexto que nunca puede faltar**.

También podría darse el caso en que esos atributos, aun cuando la predicción hubiera sido correcta, que fueran erróneos, por lo cual se podría afinar el modelo para corregirlo.

Adicionalmente, LIME presenta un módulo que llama “*submodular pick*” para tener una visión global, analizando una muestra considerable de elementos para explicarlo de manera más generalista. En las pruebas realizadas de este módulo, si queremos que trate nuestro dataset completo para poder explicar cualquier predicción, el sistema se queda sin memoria, abortándose la ejecución. Solo se ha podido ejecutar para un número medio de elementos, para los cuales se puede pedir la explicación global. Mantendremos esta parte en el código, como línea de investigación futura.

4. Resultados

El primer elemento de relevancia a destacar es el propio DataFrame generado. Del análisis de los electrocardiogramas en formato binario, previo filtrado para la eliminación de bajas frecuencias y procesar la señal para aumentar las amplitudes de las ondas, de forma que fueran mejor detectables, se extrajeron más de 500 atributos (41 atributos por derivación). Se pasa de una onda continua, con ciertos patrones (asimilables a los latidos del corazón), a un conjunto de datos cuantificables, entendibles y, ventaja principal, procesables. La estructura de cada derivación queda delimitada a estos atributos o características, que son el elemento primordial del estudio.

<i>Columna</i>	Descripción	Tipo
<i>Id</i>	Identificador del ECG	Alfanumérico
<i>Dx</i>	Diagnóstico ECG	Numérico
<i>Edad</i>	Edad del paciente	Numérico
<i>Género</i>	Género del paciente	Alfanumérico
<i>Amplitud_media_R</i>	Amplitud media onda R	Float
<i>Intervalo_medio_anterior</i>	Duración media intervalo anterior	Float
<i>Intervalo_medio_actual</i>	Duración media intervalo actual	Float
<i>Intervalo_medio_posterior</i>	Duración media intervalo posterior	Float
<i>Media_intervalos_medios</i>	Duración media de intervalos	Float
<i>Ratio_actual_anterior</i>	Relación entre intervalo actual y anterior	Float
<i>Ratio_actual_posterior</i>	Relación entre intervalo actual y siguiente	Float
<i>Ratio_media_actual</i>	Relación media de duraciones de intervalos	Float
<i>Tiempo_medio_PR</i>	Tiempo medio entre onda P y R	Float
<i>Tiempo_medio_QR</i>	Tiempo medio entre onda Q y R	Float
<i>Tiempo_medio_RS</i>	Tiempo medio entre onda R y S	Float
<i>Tiempo_medio_RT</i>	Tiempo medio entre onda R y T	Float
<i>Tiempo_medio_PQ</i>	Tiempo medio entre onda P y Q	Float
<i>Tiempo_medio_PS</i>	Tiempo medio entre onda P y S	Float
<i>Tiempo_medio_PT</i>	Tiempo medio entre onda P y T	Float
<i>Tiempo_medio_QS</i>	Tiempo medio entre onda Q y S	Float
<i>Tiempo_medio_QT</i>	Tiempo medio entre onda Q y T	Float
<i>Tiempo_medio_ST</i>	Tiempo medio entre onda S y T	Float
<i>Ratio_medio_PT_QS</i>	Relación de tiempo entre ondas PT_QS	Float
<i>Ratio_medio_QT_QS</i>	Relación de tiempo entre ondas QT_QS	Float
<i>Dif_media_amplitud_PQ</i>	Diferencia media de amplitudes ondas P y Q	Float
<i>Dif_media_amplitud_QR</i>	Diferencia media de amplitudes ondas Q y R	Float
<i>Dif_media_amplitud_RS</i>	Diferencia media de amplitudes ondas R y S	Float
<i>Dif_media_amplitud_ST</i>	Diferencia media de amplitudes ondas S y T	Float
<i>Dif_media_amplitud_PS</i>	Diferencia media de amplitudes ondas P y S	Float
<i>Dif_media_amplitud_PT</i>	Diferencia media de amplitudes ondas P y T	Float

<i>Dif_media_amplitud_QS</i>	Diferencia media de amplitudes ondas Q y S	Float
<i>Dif_media_amplitud_QT</i>	Diferencia media de amplitudes ondas Q y T	Float
<i>Ratio_medio_ampST_ampQS</i>	Relación entre diferencias de amplitudes ST/QS	Float
<i>Ratio_medio_ampRS_ampQR</i>	Relación entre diferencias de amplitudes RS/QR	Float
<i>Ratio_medio_ampPQ_ampQS</i>	Relación entre diferencias de amplitudes PQ/QS	Float
<i>Ratio_medio_ampPQ_ampQT</i>	Relación entre diferencias de amplitudes PQ/QT	Float
<i>Ratio_medio_ampPQ_ampPS</i>	Relación entre diferencias de amplitudes PQ/PS	Float
<i>Ratio_medio_ampPQ_ampQR</i>	Relación entre diferencias de amplitudes PQ/QR	Float
<i>Ratio_medio_ampPQ_ampRS</i>	Relación entre diferencias de amplitudes PQ/RS	Float
<i>Ratio_medio_ampRS_ampQS</i>	Relación entre diferencias de amplitudes PS/QS	Float
<i>Ratio_medio_ampRS_ampQT</i>	Relación entre diferencias de amplitudes RS/QT	Float
<i>Ratio_medio_ampST_ampPQ</i>	Relación entre diferencias de amplitudes ST/PQ	Float
<i>Ratio_medio_ampST_ampQT</i>	Relación entre diferencias de amplitudes ST/QT	Float

Tabla 2- Atributos de trabajo extraídos de la señal filtrada

Es importante poner en valor este conjunto de datos por la complejidad de la información inicial y el volumen inmenso de datos que se han tenido que tratar.

40.000 ECG x 12 derivaciones x 12 latidos de media (**no conocido a priori**) x 39 atributos de onda (19 de amplitudes, 12 temporales y 8 específicos de la onda R)
+
40.000 ECG x 12 derivaciones x 5.000 coordenadas temporales cada uno

Nuestro dataset final contiene los datos extraídos de 8732 electrocardiogramas con 515 atributos para el cuaderno Jupyter 1 de la entrega (“*PEC3_Cuaderno_1_Modelos_explicables_de_predicción_de_arritmias.ipynb*”) o con 41 atributos, los de la tabla, si tomamos una sola derivación como elemento de estudio (cuaderno 2 “*PEC3_Cuaderno_2_Modelos_explicables_de_predicción_de_arritmias.ipynb*”).

Dedicaremos el 70% para entrenamiento y el resto, para test. Para validar y hacer pruebas podremos usar cualquier electrocardiograma del dataset original de PhysioNet, para lo cual se ha habilitado en el código una “calculadora” que aplica las transformaciones a la instancia elegida, extrae los atributos de trabajo y se lo pasa al modelo en cuestión para comparar su salida con la etiqueta proporcionada en origen.

En el proceso de descubrimiento, se obtiene un resultado que resulta, cuanto menos, sorprendente. A la hora de estudiar las correlaciones entre los mismos datos, de un mismo evento (los latidos del corazón durante 10 segundos que duran los ECG), pero de distintas derivaciones, que se esperaría que fueran elevadas, se observa (imagen siguiente) que sucede todo lo contrario, con algunas excepciones que no analizamos, para centrarnos en la Ciencia de Datos, propiamente.

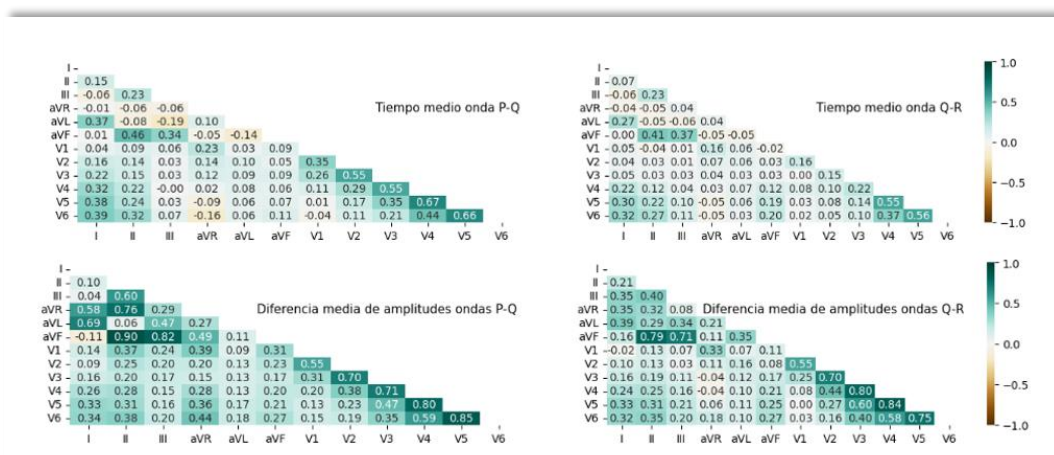


Ilustración 21- Matrices de correlación entre mismos atributos de diferentes derivaciones

En el análisis exploratorio visual (EDA) realizado se encuentran algunos hallazgos que sí encajan con la lógica, como que, a mayor edad, más frecuencia de algún tipo de arritmia. No se ha profundizado mucho en este apartado de análisis por el elevado número de atributos, que desde el desconocimiento de la práctica médica sería complicado afinar, y para dar más espacio a los modelos. Y es aquí donde empezamos un recorrido desde los modelos no supervisados (aunque conocemos de antemano los valores objetivo), para comprobar si podríamos representar los distintos grupos o clusters en una gráfica donde estuvieran los ECG diferenciados y agrupados en distintas zonas del espacio hasta los modelos propiamente supervisados en los que profundizaremos a continuación.

Pero antes, el resultado de aplicar modelos no supervisados. Ni k-means, ni dendogramas, ni métodos de clustering por densidad, como DBSCAN o UMAP, han dado separaciones claras por etiquetas, “conocidas” por nosotros, pero no por los algoritmos. De haberlo conseguido, hubiera supuesto una manera “fácil” de explicar si un electrocardiograma presenta una arritmia u otra en función de donde se representa en ese espacio.

Se han utilizado distintos tipos de distancia y distintos valores de los parámetros, pero no se han conseguido separar los clusters según nuestras etiquetas. No obstante, mediante UMAP sí se aprecia cierta tendencia que merece mencionarse. En la imagen siguiente, donde cada color representa un tipo de arritmia, podemos observar cómo algunas están razonablemente diferenciadas del resto, acotándose en ciertas zonas del espacio 2D utilizado. Sin embargo, otras están extremadamente mezcladas y confusas (zona central), como para poder clasificar correctamente de qué tipo es cada uno de los puntos (que representan un electrocardiograma).

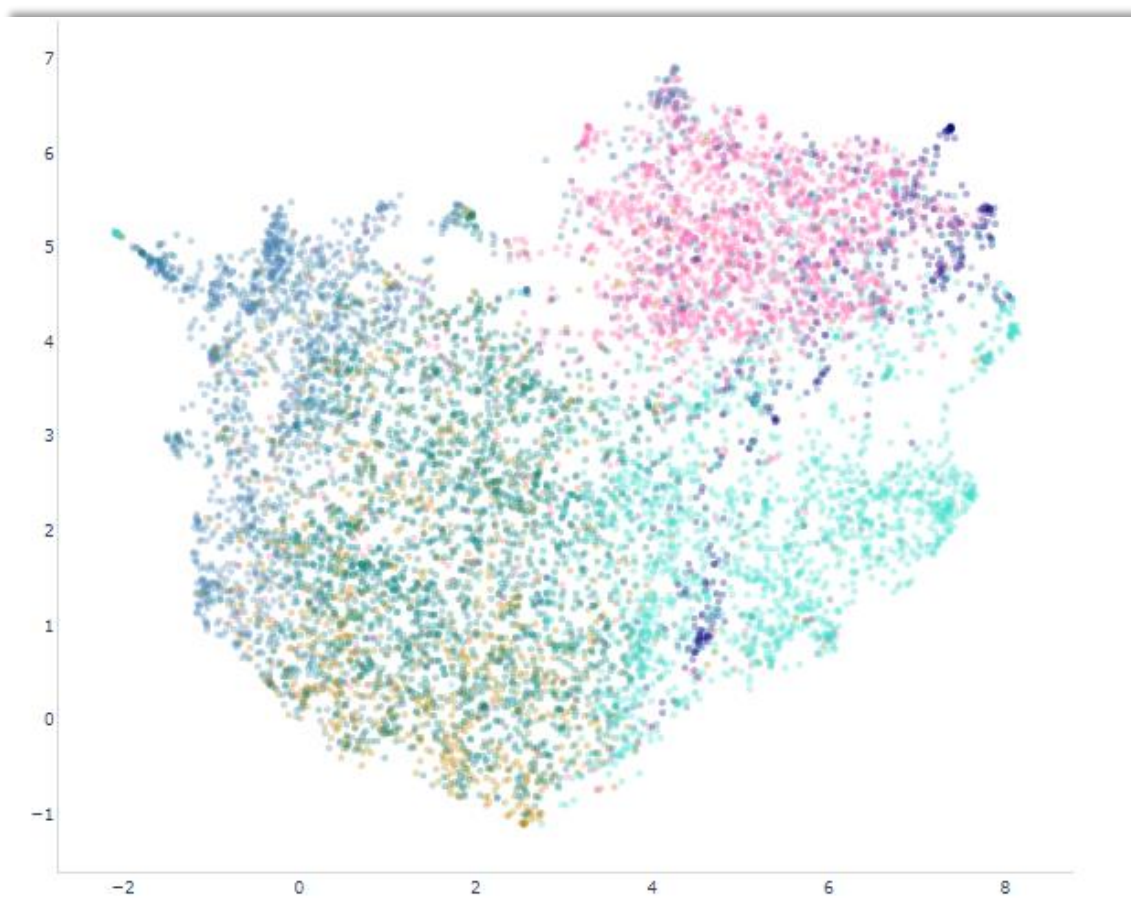


Ilustración 22- UMAP para 12 derivaciones

Vamos a exponer a continuación los resultados de los mejores modelos supervisados con una sola derivación, ubicados en cuaderno Jupyter “PEC3_Cuaderno_2_Modelos_explicables_de_predicción_de_arritmias.ipynb”. Los resultados y comentarios son extrapolables al cuaderno 1 que trata los datos de todas las derivaciones en conjunto.

No vamos a explicar los modelos utilizados, que ya están descritos sucintamente en el capítulo anterior, deteniéndonos en su lugar en las métricas y en los atributos más relevantes de los modelos elegidos, según LIME. Es evidente que cada modelo construye sus predicciones en base a distintos algoritmos, de difícil comprensión en muchos casos para un paciente o un médico, en base a funciones y cálculos estadísticos, y es justamente este factor el que queremos limitar, aportando explicaciones basadas en los atributos; si bien, que no sean “entendibles” no quita su validez, a la luz de los resultados.

Para determinar los mejores modelos cuyos resultados presentaremos a continuación, vamos a contrastar algunas de las métricas obtenidas, como la precisión y, para reducir los falsos negativos o falsos positivos, el área medio bajo la curva ROC o AUC. De los mejores mostraremos el detalle de la curva ROC por cada clase o etiqueta objetivo.

```

for modelo,pred in zip(lista,predicciones):
    print(modelo,"Accuracy:",round(accuracy_score(y_test,pred) * 100 , 2),"%")

Gaussian_Naive_Bayes Accuracy: 60.08 %
K_Nearest_Neighbors Accuracy: 71.45 %
Support_Vector_Machine Accuracy: 79.85 %
StoChastic_Gradient_Descent Accuracy: 72.33 %
Decision_Tree_Classifier Accuracy: 75.69 %
Logistic_Regression Accuracy: 75.73 %
Random_Forest Accuracy: 83.44 %
XGBoosting Accuracy: 83.74 %
Gradient_Boosting_Classifier Accuracy: 82.71 %
Ensemble_Learning_UMAP_SGD Accuracy: 67.82 %
IC_Max_voting Accuracy: 81.34 %
IC_Stacking Accuracy: 84.43 %

```

Tabla 3- Datos de precisión/accuracy de los modelos analizados

```

for modelo,probs in zip(lista,probabilidades):
    print(modelo,"AUROC:",round(roc_auc_score(y_test,probs[:], multi_class = 'ovo') *
100 , 2),"%") #medias one-versus-one

Gaussian_Naive_Bayes AUROC: 80.26 %
K_Nearest_Neighbors AUROC: 83.5 %
Support_Vector_Machine AUROC: 91.38 %
StoChastic_Gradient_Descent AUROC: 83.02 %
Decision_Tree_Classifier AUROC: 81.96 %
Logistic_Regression AUROC: 86.09 %
Random_Forest AUROC: 90.14 %
XGBoosting AUROC: 92.77 %
Gradient_Boosting_Classifier AUROC: 88.14 %
Ensemble_Learning_UMAP_SGD AUROC: 75.57 %
IC_Max_voting AUROC: 91.38 %
IC_Stacking AUROC: 89.24 %

```

Tabla 4- Comparativa de AUC media de cada modelo

Podemos comparar también la variabilidad de los resultados de cada modelo, considerando como mejores, si atendiéramos a la siguiente ilustración, los que presentan un boxplot más compacto, con el valor medio, en triángulo verde, más elevado y, a ser posible, lo más cercano a la mediana (línea naranja). En todos los modelos sucede que la media es inferior a la mediana por el efecto que producen las dos clases escasamente representadas y que comentábamos anteriormente, debido a que la precisión para estas clases es peor. Los modelos no han contado con suficientes instancias para poder predecir con garantías estas dos clases. Si nos fijamos solo en estos datos, XGBoosting sería el modelo a elegir.

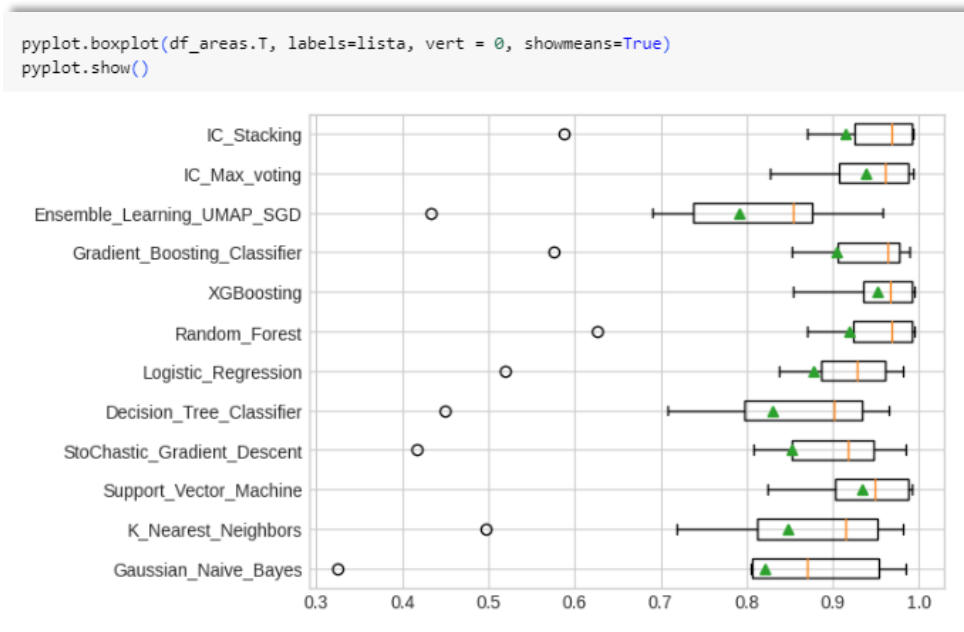


Ilustración 23- Comparativa de variabilidad de resultados

En todo caso nos quedaremos con:

- Support_Vector_Machine
- Random_Forest
- XGBoosting
- IC_Max_voting
- IC_Stacking

Mostraremos también los resultados de la red neuronal sencilla que construimos, a modo de ejemplo, para ver cómo lo ha hecho y si pudiera ser elegible como un modelo preciso, aunque no profundizaremos ya que el resultado ha sido pobre.

En cada modelo ejecutaremos LIME y compararemos las predicciones, a la vez que podremos observar los atributos más relevantes para el modelo y que más han influido, al menos de manera

local, en la predicción. Los datos se deben interpretar como se recoge en la imagen adyacente.

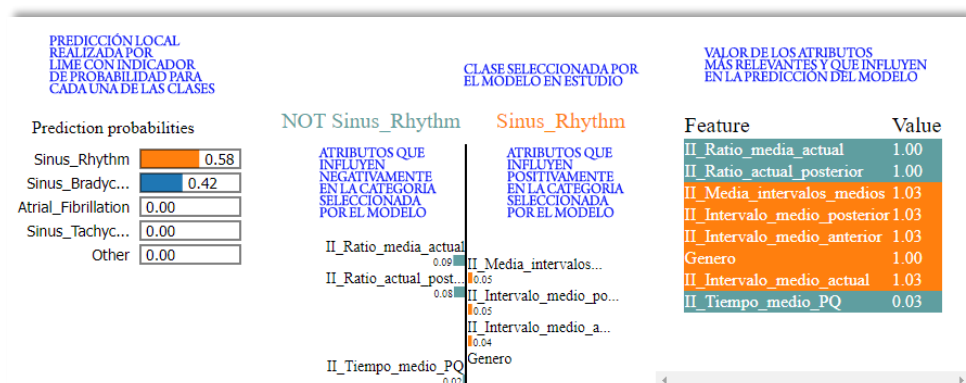


Ilustración 24- Interpretación de salida LIME

4.1. Support_Vector_Machine

Configuración de parámetros

```

csvm = make_pipeline(StandardScaler(),
                    SVC(C=10, gamma=0.01, probability = True))
csvm.fit(X_train, y_train)
    
```

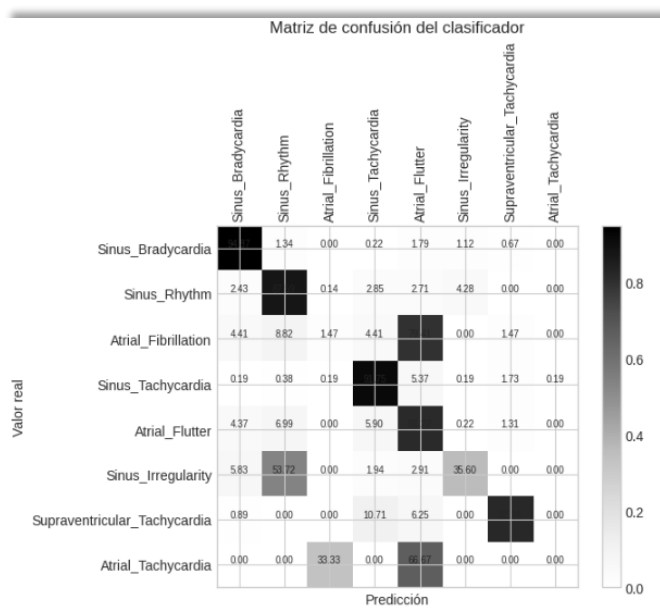


Ilustración 25- Matriz de confusión normalizada para SVM

Precisión con datos de entrenamiento: 0.8403141361256544

Precisión con datos de test : 0.7984732824427481

	precision	recall	f1-score	support
<i>Sinus_Bradycardia</i>	88%	95%	91%	448
<i>Sinus_Rhythm</i>	74%	88%	80%	701
<i>Atrial_Fibrillation</i>	25%	1%	3%	68
<i>Sinus_Tachycardia</i>	87%	92%	90%	521
<i>Atrial_Flutter</i>	75%	81%	78%	458
<i>Sinus_Irregularity</i>	75%	36%	48%	309
<i>Supraventricular_Tachycardia</i>	83%	82%	83%	112
<i>Atrial_Tachycardia</i>	0%	0%	0%	3

Tabla 5- Métricas de SVM para cada una de las clases objetivo

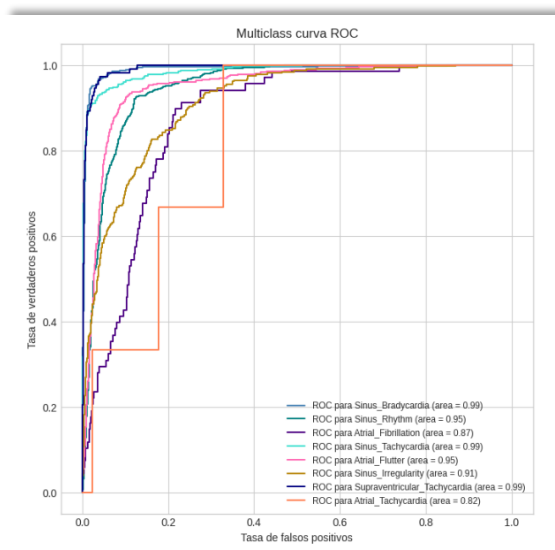


Ilustración 26- Curva ROC multiclase y AUC para SVM

```
[ ] prueba = 'JS00010'
uno = calculadora(prueba, pintar = False)

print('\nDiagnóstico real = ', uno['Nombre_ritmo_cardiaco'])

Pred_sin = uno.copy()
Pred_sin = Pred_sin.drop(columns=['Nombre_ritmo_cardiaco'])

elemento = int(csvm.predict(Pred_sin))
print('Predicción = ', list(dicc_arritmias.values())[elemento])

Leído y Procesando 9 ecg-arrhythmia/1.0.0/WFDBRecords/01/010/ JS00010
Diagnóstico real = 0 Sinus_Bradycardia
Name: Nombre_ritmo_cardiaco, dtype: category
Categories (1, object): ['Sinus_Bradycardia']
Predicción = Sinus_Bradycardia

[ ] #Explicación LOCAL del diagnóstico predicho por el modelo usado según LIME, por sus atributos de mayor importancia
indice_prediccion = obtener_indice_por_valor(dicc_arritmias, list(dicc_arritmias.values())[elemento])
dd=[]
dd.append(indice_prediccion)

lime = explainer.explain_instance(Pred_sin.T[0], csvm.predict_proba, num_features= 8, labels=dd)
lime.show_in_notebook(show_table=True)
```

Prediction probabilities		Feature		Value
Sinus_Bradyc...	0.57	II_Intervalo_medio_a...	II_Intervalo_medio_actual	1.03
Sinus_Rhythm	0.31	II_Media_intervalos...	II_Media_intervalos_medios	1.03
Sinus_Irregul...	0.11	II_Intervalo_medio_a...	II_Intervalo_medio_anterior	1.03
Atrial_Flutter	0.01	II_Intervalo_medio_po...	II_Intervalo_medio_posterior	1.03
Other	0.00	II_Dif_media_amplitud...	II_Dif_media_amplitud_QT	0.19
		II_Ratio_actual_post...	II_Ratio_actual_posterior	1.00
		II_Dif_media_amplitud...	II_Dif_media_amplitud_PS	-0.31
		II_Tiempo_medio_PQ	II_Tiempo_medio_PQ	0.03

Ilustración 27- Predicciones de SVM para un ejemplo y explicación LIME

4.2. Random_Forest

Configuración de parámetros

```
crf = make_pipeline(StandardScaler(),
                    RandomForestClassifier(random_state=seed,
                                        criterion='gini', n_estimators =1000,
                                        max_depth=20, bootstrap=False, n_jobs=8))
```

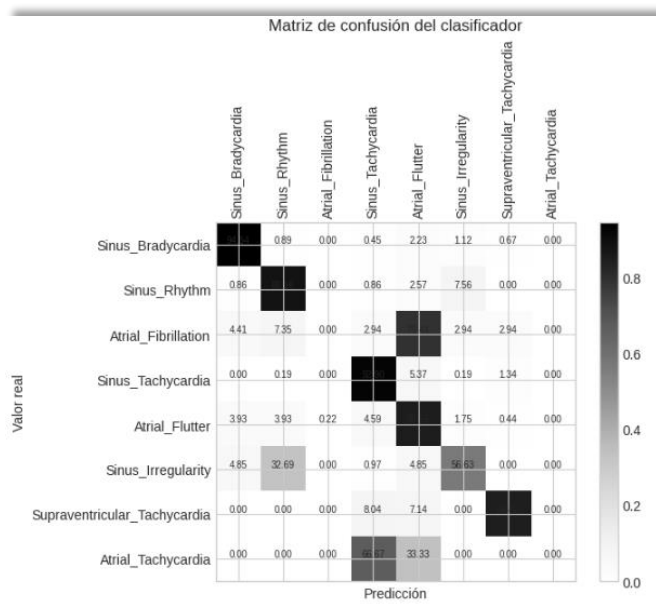


Ilustración 28- Matriz de confusión normalizada para Random Forest

Precisión con datos de entrenamiento: 0.9988547120418848

Precisión con datos de test : 0.834351145038168

	precision	recall	f1-score	support
<i>Sinus_Bradycardia</i>	91%	95%	93%	448
<i>Sinus_Rhythm</i>	83%	88%	85%	701
<i>Atrial_Fibrillation</i>	0%	0%	0%	68
<i>Sinus_Tachycardia</i>	91%	93%	92%	521
<i>Atrial_Flutter</i>	74%	85%	79%	458
<i>Sinus_Irregularity</i>	72%	57%	63%	309
<i>Supraventricular_Tachycardia</i>	87%	85%	86%	112
<i>Atrial_Tachycardia</i>	0%	0%	0%	3

Tabla 6- Métricas de Random Forest para cada una de las clases objetivo

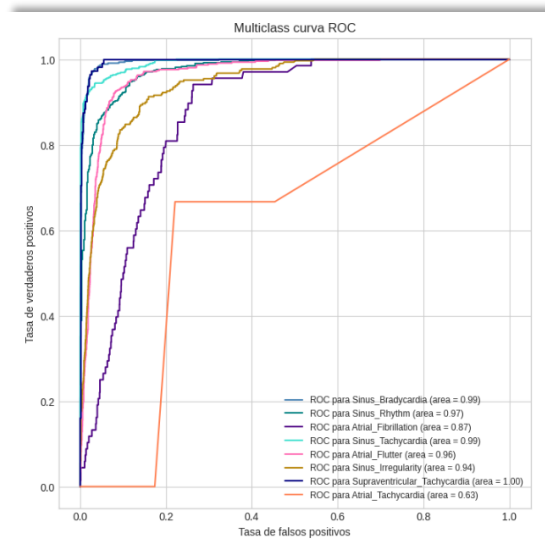


Ilustración 29- Curva ROC multiclase y AUC para Random Forest

```

prueba = 'JS00010'
uno = calculadora(prueba, pintar = False)

print('\nDiagnóstico real = ', uno['Nombre_ritmo_cardiaco'])

Pred_sin = uno.copy()
Pred_sin = Pred_sin.drop(columns=['Nombre_ritmo_cardiaco'])

elemento = int(cr.f.predict(Pred_sin))
print('Predicción = ', list(dicc_arritmias.values())[elemento])

```

Leído y Procesando 9 ecg-arrhythmia/1.0.0/WFDBRecords/01/010/ JS00010
 Diagnóstico real = 0 Sinus_Bradycardia
 Name: Nombre_ritmo_cardiaco, dtype: category
 Categories (1, object): ['Sinus_Bradycardia']
 Predicción = Sinus_Bradycardia

```

#Explicación LOCAL del diagnóstico predicho por el modelo usado según LIME, por sus atributos de mayor importancia
indice_prediccion = obtener_indice_por_valor(dicc_arritmias, list(dicc_arritmias.values())[elemento])
dd=[]
dd.append(indice_prediccion)

lime = explainer.explain_instance(Pred_sin.T[0], cr.f.predict_proba, num_features= 8, labels=dd)
lime.show_in_notebook(show_table=True)

```

Prediction probabilities		Feature		Value
Sinus_Bradyc...	1.00	II_Media_intervalos...	II_Media_intervalos_medios	1.03
Sinus_Rhythm	0.00	II_Intervalo_medio_po...	II_Intervalo_medio_posterior	1.03
Atrial_Fibrillation	0.00	II_Intervalo_medio_a...	II_Intervalo_medio_actual	1.03
Sinus_Tachyc...	0.00	II_Intervalo_medio_a...	II_Intervalo_medio_anterior	1.03
Other	0.00	II_Ratio_medio_actua...	II_Ratio_medio_actual	1.00
		II_Ratio_medio_actua...	II_Ratio_medio_actual_posterior	1.00
		II_Ratio_medio_actua...	II_Tiempo_medio_QT	0.26
		II_Tiempo_medio_QT	II_Ratio_medio_ampST_ampPQ	-0.19
		II_Tiempo_medio_QT		
		II_Ratio_medio_ampS...		

Ilustración 30- Predicciones de Random Forest para un ejemplo y explicación LIME

4.3. XGBoosting

Configuración de parámetros

```
cxgb = make_pipeline(StandardScaler(),
                    xgb.XGBClassifier(
                        objective='multi:softprob',
                        num_class=8,
                        max_depth=20,
                        learning_rate= 0.01,
                        n_estimators=1000))
```

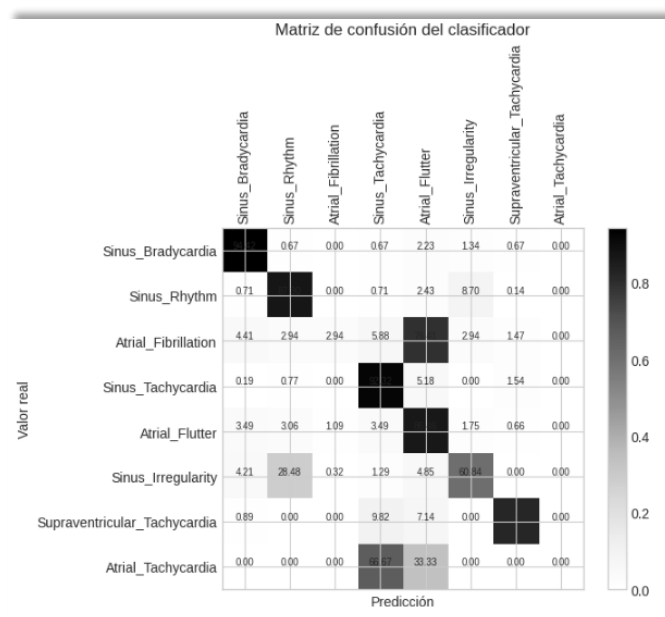


Ilustración 31- Matriz de confusión normalizada para XGBoost

Precisión con datos de entrenamiento: 1.0
 Precisión con datos de test : 0.8374045801526717

	precision	recall	f1-score	support
<i>Sinus_Bradycardia</i>	92%	94%	93%	448
<i>Sinus_Rhythm</i>	85%	87%	86%	701
<i>Atrial_Fibrillation</i>	25%	3%	5%	68
<i>Sinus_Tachycardia</i>	91%	92%	92%	521
<i>Atrial_Flutter</i>	75%	86%	80%	458
<i>Sinus_Irregularity</i>	71%	61%	66%	309
<i>Supraventricular_Tachycardia</i>	85%	82%	84%	112
<i>Atrial_Tachycardia</i>	0%	0%	0%	3

Tabla 7- Métricas de XGBoost para cada una de las clases objetivo

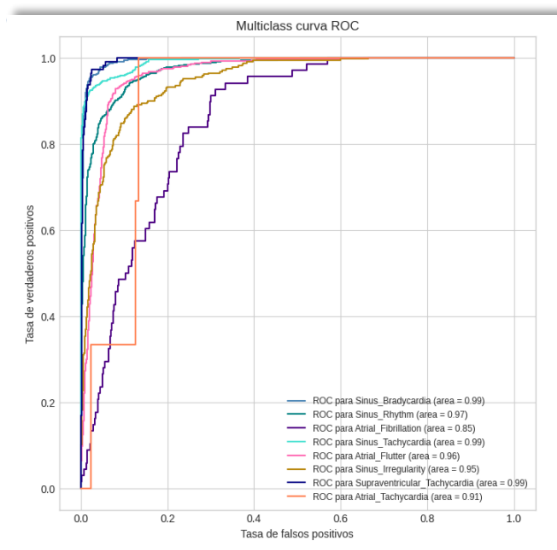


Ilustración 32- Curva ROC multiclase y AUC para XGBoost

```

prueba = 'JS00010'
uno = calculadora(prueba, pintar = True)

print('\nDiagnóstico real = ', uno['Nombre_ritmo_cardiaco'])

Pred_sin = uno.copy()
Pred_sin = Pred_sin.drop(columns=['Nombre_ritmo_cardiaco'])

elemento = int(cxgb.predict(Pred_sin))
print('Predicción = ', list(dicc_arritmias.values())[elemento])
    
```

Leído y Procesando 9 ecg-arrhythmia/1.0.0/WFDBRecords/01/010/ JS00010

Diagnóstico real = 0 Sinus_Bradycardia
 Name: Nombre_ritmo_cardiaco, dtype: category
 Categories (1, object): ['Sinus_Bradycardia']
 Predicción = Sinus_Bradycardia

```

[ ] #Explicación LOCAL del diagnóstico predicho por el modelo usado según LIME, por sus atributos de mayor importancia
indice_prediccion = obtener_indice_por_valor(dicc_arritmias, list(dicc_arritmias.values())[elemento])
dd=[]
dd.append(indice_prediccion)

lime = explainer.explain_instance(Pred_sin.T[0], cxgb.predict_proba, num_features= 8, labels=dd)
lime.show_in_notebook(show_table=True)
    
```

Prediction probabilities		Feature	
Sinus_Bradyc...	1.00	II_Intervalo_medio_po...	1.03
Sinus_Rhythm	0.00	II_Intervalo_medio_anterior	1.03
Sinus_Irregul...	0.00	II_Intervalo_medio_actual	1.03
Atrial_Flutter	0.00	II_Media_intervalos_medios	0.17
Other	0.00	II_Tiempo_medio_PS	0.08
		II_Dif_media_amplitud_ST	-0.31
		Genero	1.00

Ilustración 33- Predicciones de XGBoost para un ejemplo y explicación LIME

4.4. IC_Max_voting

Configuración de parámetros

```
classifiers = [
    ("gnb", gnb),
    ("cknn", cknn),
    ("csvm", csvm),
    ("csvm_sgd", csvm_sgd),
    ("cdt", cdt),
    ("crl", crl),
    ("crf", crf),
    ("cxgb", cxgb),
    ("cgbc", cgbc),
    ("t1", t1)
]
```

cmaxv = VotingClassifier(estimators=classifiers, voting='soft') #soft, en función de la suma de probabilidades predichas en los estimadores ya calibrados vs hard, votación pura y dura

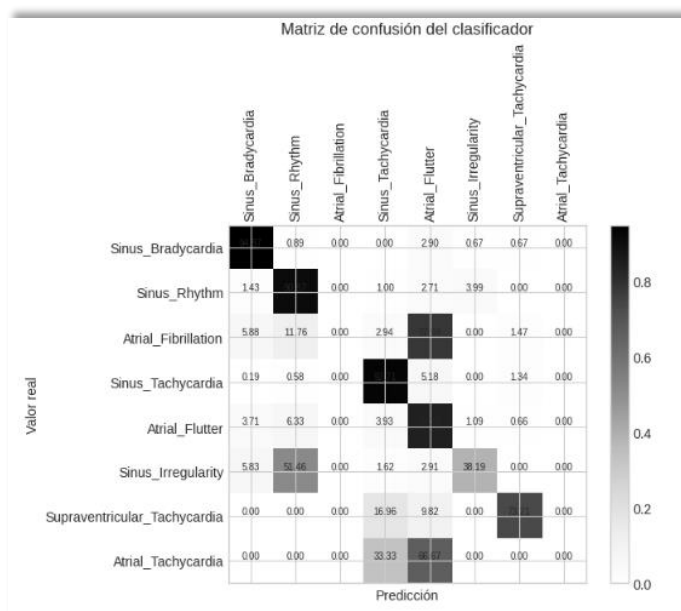


Ilustración 34- Matriz de confusión normalizada para MaxVoting

Precisión con datos de entrenamiento: 0.9286649214659686

Precisión con datos de test : 0.8133587786259542

	precision	recall	f1-score	support
Sinus_Bradycardia	89%	95%	92%	448
Sinus_Rhythm	76%	91%	83%	701
Atrial_Fibrillation	0%	0%	0%	68
Sinus_Tachycardia	90%	93%	91%	521
Atrial_Flutter	74%	84%	79%	458
Sinus_Irregularity	77%	38%	51%	309
Supraventricular_Tachycardia	85%	73%	79%	112
Atrial_Tachycardia	0%	0%	0%	3

Tabla 8- Métricas de Máxima votación para cada una de las clases objetivo

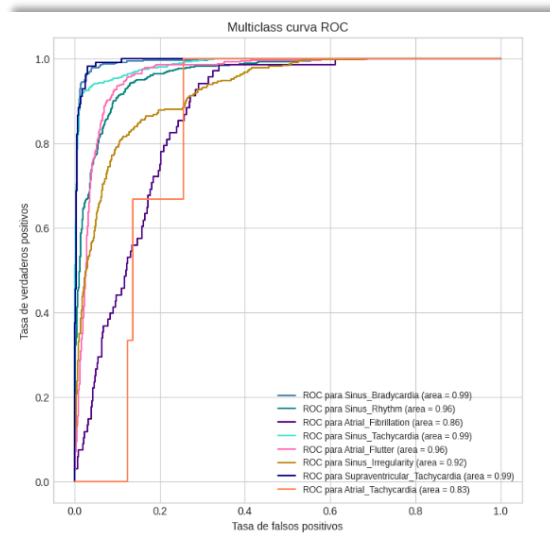


Ilustración 35- Curva ROC multiclase y AUC para MaxVoting

```

prueba = 'JS00010'
uno = calculadora(prueba, pintar = False)

print('\nDiagnóstico real = ', uno['Nombre_ritmo_cardiaco'])

Pred_sin = uno.copy()
Pred_sin = Pred_sin.drop(columns=['Nombre_ritmo_cardiaco'])

elemento = int(cmaxv.predict(Pred_sin))
print('Predicción = ', list(dicc_arritmias.values())[elemento])

```

Leído y Procesando 9 ecg-arrhythmia/1.0.0/WFDBRecords/01/010/ JS00010
 Diagnóstico real = 0 Sinus_Bradycardia
 Name: Nombre_ritmo_cardiaco, dtype: category
 Categories (1, object): ['sinus_Bradycardia']
 Predicción = Sinus_Bradycardia

```

[ ] #Explicación LOCAL del diagnóstico predicho por el modelo usado según LIME, por sus atributos de mayor importancia
indice_prediccion = obtener_indice_por_valor(dicc_arritmias, list(dicc_arritmias.values())[elemento])
dd = []
dd.append(indice_prediccion)

lime = explainer.explain_instance(Pred_sin.T[0], cmaxv.predict_proba, num_features= 8, labels=dd)
lime.show_in_notebook(show_table=True)

```

Prediction probabilities	OT Sinus_Bradycardia	Feature	Value
Sinus_Bradyc... 0.66	II_Intervalo_medio_a... 0.03	II_Intervalo_medio actual	1.03
Sinus_Rhythm 0.28	II_Media_intervalos... 0.04	II_Media_intervalos_medios	1.03
Sinus_Irregul... 0.05	II_Intervalo_medio_po... 0.04	II_Intervalo_medio_posterior	1.03
Atrial_Fluter 0.01	II_Intervalo_medio_a... 0.02	II_Intervalo_medio anterior	1.03
Other 0.01	II_Dif_media_amplit... 0.02	II_Dif_media amplitud ST	0.08
	II_Dif_media_amplit... 0.02	II_Dif_media amplitud QT	0.19
	II_Tiempo_medio_PS 0.01	II_Tiempo_medio PS	0.17
	Genero 0.00	Genero	1.00

Ilustración 36- Predicciones de MaxVoting para un ejemplo y explicación LIME

4.5. IC_Stacking

Configuración de parámetros

```
#configuramos nuestro modelo de modelos

level0 = []
level0.append(("gnb", gnb ))
level0.append(("cknn", cknn))
level0.append(("csvm", csvm))
level0.append(("csvm_sgd", csvm_sgd))
level0.append(("cdt", cdt))
level0.append(("crl", crl))
level0.append(("crf", crf))
level0.append(("cxgb", cxgb))
level0.append(("cgbc", cgbc))
level0.append(("t1", t1))

#como nivel 1 "selector" elegiremos un modelo sencillo.
level1 = LogisticRegression()

#generamos instancia y lo probamos
ic = StackingClassifier(estimators=level0,final_estimator=level1,cv=4)
```

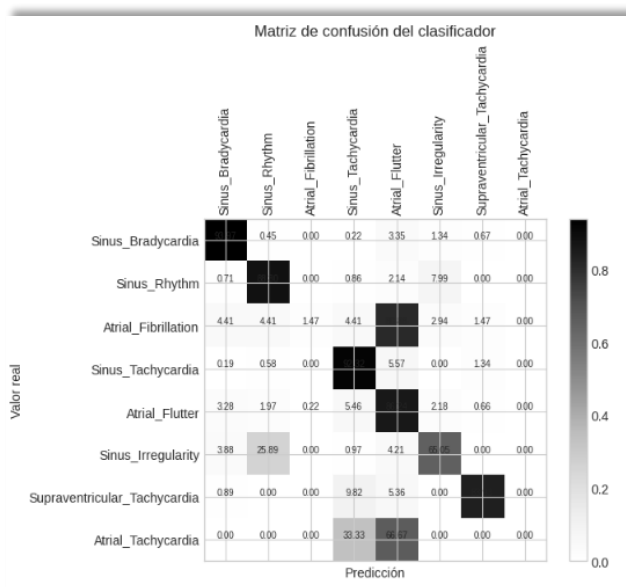


Ilustración 37- Matriz de confusión normalizada para IC-Stacking

Precisión con datos de entrenamiento: 0.96842277486911
 Precisión con datos de test : 0.8442748091603054

	precision	recall	f1-score	support
Sinus_Bradycardia	92%	94%	93%	448
Sinus_Rhythm	86%	88%	87%	701
Atrial_Fibrillation	50%	1%	3%	68
Sinus_Tachycardia	91%	92%	91%	521
Atrial_Flutter	75%	86%	80%	458
Sinus_Irregularity	73%	65%	69%	309
Supraventricular_Tachycardia	87%	84%	85%	112
Atrial_Tachycardia	0%	0%	0%	3

Tabla 9- Métricas de IC-Stacking para cada una de las clases objetivo

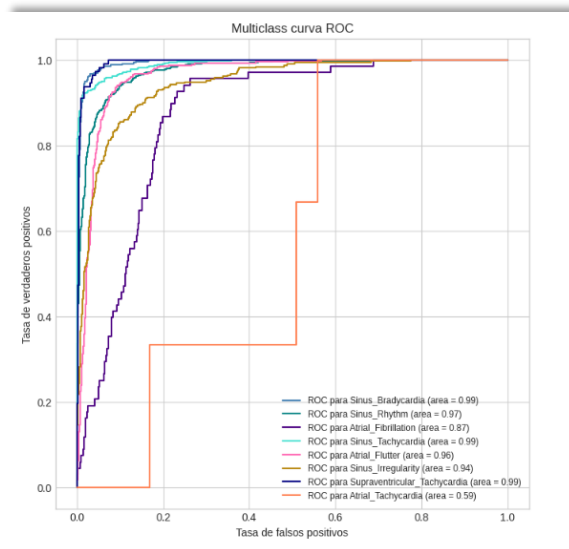


Ilustración 38- Curva ROC multiclase y AUC para IC-Stacking

```

prueba = 'J500010'
uno = calculadora(prueba, pintar = True)
print('\nDiagnóstico real = ',uno['Nombre_ritmo_cardiaco'])

Pred_sin = uno.copy()
Pred_sin=Pred_sin.drop(columns=['Nombre_ritmo_cardiaco'])

elemento = int(ic.predict(Pred_sin))
print('Predicción = ', list(dicc_arritmias.values())[elemento])

```

Leído y Procesando 9 ecg-arrhythmia/1.0.0/WFDBRecords/01/010/ J500010

Diagnóstico real = 0 Sinus_Bradycardia
 Name: Nombre_ritmo_cardiaco, dtype: category
 Categories (1, object): ['Sinus_Bradycardia']
 Predicción = Sinus_Bradycardia

```

#Explicación LOCAL del diagnóstico predicho por el modelo usado según LIME, por sus atributos de mayor importancia
indice_prediccion = obtener_indice_por_valor(dicc_arritmias, list(dicc_arritmias.values())[elemento])
dd=[]
dd.append(indice_prediccion)

lime = explainer.explain_instance(Pred_sin.T[0], ic.predict_proba, num_features= 8, labels=dd)
lime.show_in_notebook(show_table=True)

```

Prediction probabilities		Feature	
Sinus_Bradyc...	0.87	II_Intervalo_medio actual	1.03
Sinus_Irregul...	0.05	II_Media_intervalos_medios	1.03
Atrial_Flutter	0.04	II_Intervalo_medio_posterior	1.00
Sinus_Rhythm	0.04	II_Ratio_actual_posterior	1.00
Other	0.00	II_Ratio_media actual	1.00
		II_Intervalo_medio anterior	1.03
		II_Dif_media amplitud QT	0.19
		II_Tiempo_medio QR	0.04

Ilustración 39-Predicciones de IC-Stacking para un ejemplo y explicación LIME

4.6. Red neuronal sencilla/clasificadora

Configuración de parámetros

```

data_config = DataConfig(
    target=['target'],
    continuous_cols=num_col_names,
    categorical_cols=cat_col_names,
    continuous_feature_transform="quantile_normal",
    normalize_continuous_features=True
)
trainer_config = TrainerConfig(
    auto_lr_find=True, # activamos la búsqueda automática del learning rate
    # óptimo
    batch_size=32,
    max_epochs=100,
    gpus=1,
)
optimizer_config = OptimizerConfig()
model_config = CategoryEmbeddingModelConfig(
    task="classification",
    layers="4096-4096-512", # Estructura de nodos de la red neuronal
    activation="ELU", # Función de activación
    learning_rate = 1e-3,
    metrics=["accuracy"],
)
    
```

Test metric	DataLoader 0
test_accuracy	0.7678503394126892
test_loss	0.7198674082756042

Precisión con datos de entrenamiento: 0.8342875838377228

Precisión con datos de test : 0.7678503245513555

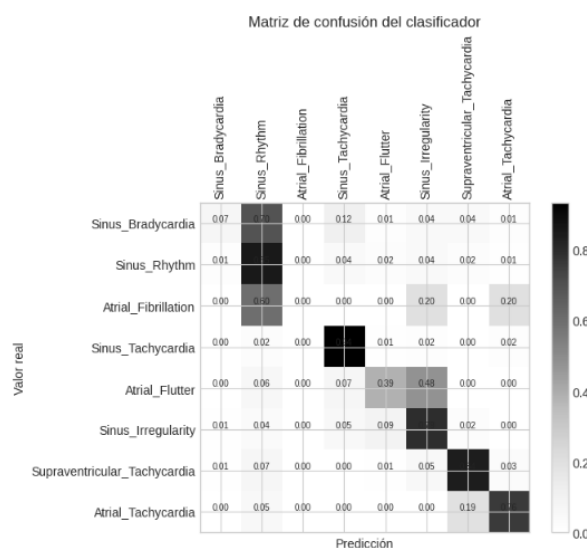


Ilustración 40- Matriz de confusión normalizada para Red Neuronal Clasificadora

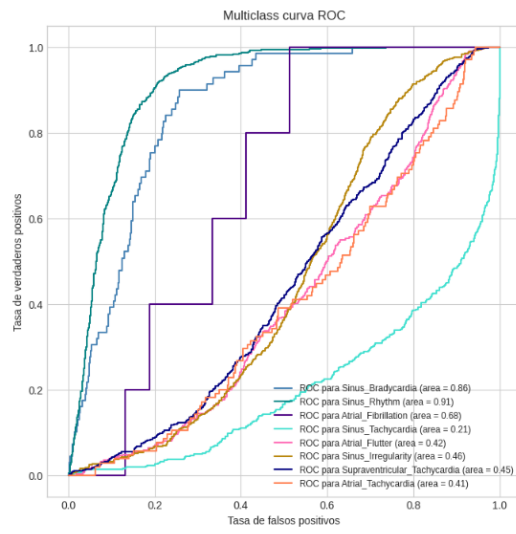


Ilustración 41- Curva ROC multiclase y AUC para Clasificador Red Neuronal

5. Conclusiones y trabajos futuros

5.1. Sobre datos y modelos

Del estudio realizado, basándonos en la experiencia y en los datos obtenidos podemos sacar algunas conclusiones que deberían ser corroboradas por algún especialista médico, pero que podrían deberse a que el dataset utilizado esté ya tratado y adecuado para su tratamiento en estudios de analítica de datos, para el ámbito académico, habiendo perdido parte de su validez médica.

Por una parte, nos encontramos con la sorpresa de la **no correlación entre los mismos datos en diferentes derivaciones, a pesar de proceder del mismo evento**, aunque registrado en distintos lugares del torso humano (*Ilustración 21*). Los datos, para todas las derivaciones, han sido tratados de la misma manera, con las mismas funciones, métodos y procedimientos, pero salvo con la duración de los latidos, algo evidente, apenas hay atributos correlacionados entre sí.

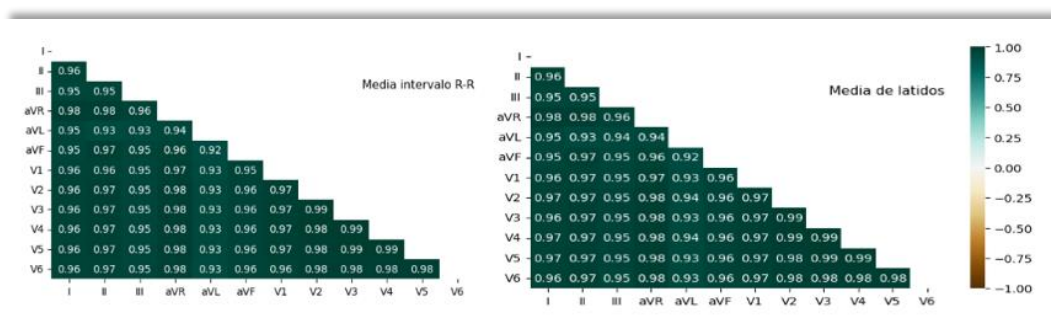


Ilustración 42- Correlaciones existentes inter-derivaciones para duración media de latidos y número de latidos

Por otra, tenemos las altas tasas de precisión de todos los modelos practicados, que nos hacen sospechar de que algo pudiera no estar bien del todo. ¿El código? ¿Los datos? Sería de gran utilidad poder validar los resultados en algún dataset del mundo real, lo cual excede del presente trabajo. Para “Atrial Tachycardia” el porcentaje disminuye sustancialmente, achacándolo al bajo número de instancias con esta etiqueta, pero igualmente nos resulta llamativo su alta precisión en el resto de clases. **Un menor número de instancias complica su clasificación.** Eliminar estas clases seguramente podría mejorar los resultados a la hora de identificar el resto de clase, pero lo hemos querido mantener, sin hacer *oversampling*, para comprobar cómo afectan y en qué medida los modelos reconocen los patrones con pocas instancias.

	SINUS BRADYCARDIA	SINUS RHYTHM	ATRIAL FIBRILLATION	SINUS TACHYCARDIA	ATRIAL FLUTTER	SINUS IRREGULARITY	SUPRAVENTRICULAR TACHYCARDIA	ATRIAL TACHYCARDIA
<u>Gaussian Naive Bayes</u>	95,16	85,77	82,9	95,36	88,06	83,3	97,45	8,69
<u>K Nearest Neighbors</u>	97,46	89,42	70,89	96,18	91,77	85,69	92,77	49,73
<u>Support Vector Machine</u>	99,01	93,82	88	98,79	94,73	91,39	98,86	90,36
<u>Stochastic Gradient Descent</u>	96,62	89,54	80,55	95,02	91,93	85,71	98,1	76,13
<u>Decision Tree Classifier</u>	94,58	92,92	80,43	97,16	91,37	84,91	96,08	42,2
<u>Logistic Regression</u>	96,37	93,36	86,16	96,31	92,12	89,96	98,09	58,07
<u>Random Forest</u>	99,52	97,01	86,9	99,28	95,73	94,51	99,42	82,05
<u>XGBoosting</u>	99,46	97,14	83,83	99,36	95,7	95,03	99,18	87,29
<u>Gradient Boosting Classifier</u>	98,49	96,6	83,56	98,76	94,29	93,25	95,88	81,96
<u>Ensemble Learning UMAP SGD</u>	93,54	83,41	68,58	90,94	83,87	81,7	82,25	72,68
<u>IC Max voting</u>	99,13	95,61	87,62	98,93	95,42	93,48	99,14	88,63
<u>IC Stacking</u>	99,36	97,27	88,06	99,17	96,02	94,87	98,34	83,87

Tabla 10- AUC por modelo y clase – datos obtenidos para la derivación II

Otra conclusión con la que nos quedamos es que **más datos no implican mejores resultados para algún determinado modelo**, ya sea por la redundancia; por la calidad del dato, más difícil de controlar en datasets de alta dimensionalidad; por la irrelevancia de algunos; o por algún tipo de dependencia por descubrir entre los datos. Es significativo cómo matriz de confusión normalizada para nuestra red neuronal fracasa estrepitosamente cuando trabaja con todos los atributos y de manera aceptable para algunas clases objetivo, las más representadas en el conjunto de datos, como se muestra en la imagen siguiente:

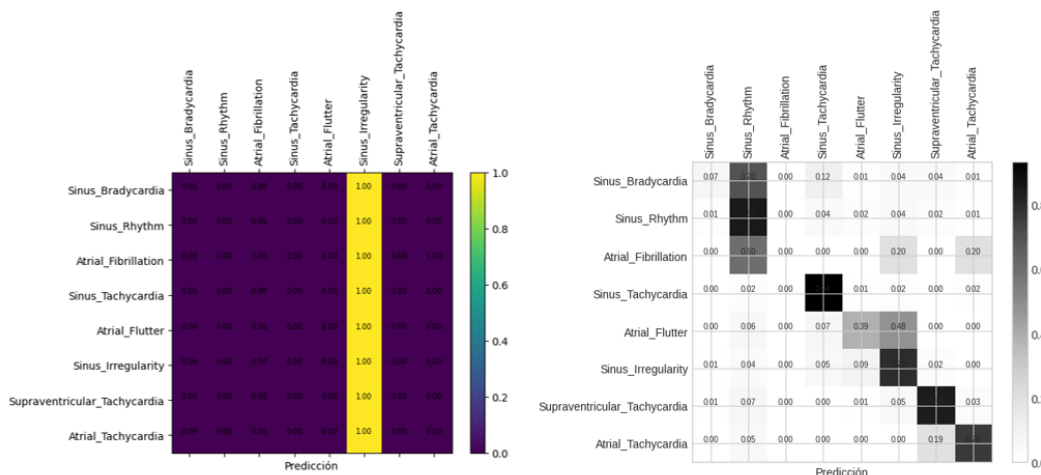


Ilustración 43- Comparativa de matrices de confusión

A la izquierda RN con todos los atributos; A la derecha, RN con atributos de una derivación

Para el resto de modelos se ha comprobado que los resultados, cuando se utilizan los más de 500 atributos desplegados en conjunto, no son mejores sustancialmente que cuando utilizamos los datos de una sola derivación (41 atributos). Y esto choca con la intuición y choca con el primer punto de este apartado en el que se indica que, según las observaciones y las correlaciones intra-

derivación son mínimas. La mejora es del orden de un 2% y no para todos los modelos ni todas las derivaciones.

Similar a lo anterior es el hecho de que, **los resultados por derivación, comparados entre sí, tampoco difieren mucho**. Sería indiferente, en términos generales, que usáramos los datos de una derivación u otra, ya que la precisión diferiría muy poco. Aunque si hubiera que quedarse con alguna derivación como la que mejores datos arroja, ésta sería la II. Se aportan tres resultados de distintas derivaciones, de un mismo modelo, como evidencia de lo indicado.

MODELO IC-max_voting	V2	Avf	II
Precisión con datos de entrenamiento:	0.9325916230366492	0.9121400523560209	0.9275196335078534
Precisión con datos de test :	0.8232824427480916	0.8030534351145038	0.8076335877862595

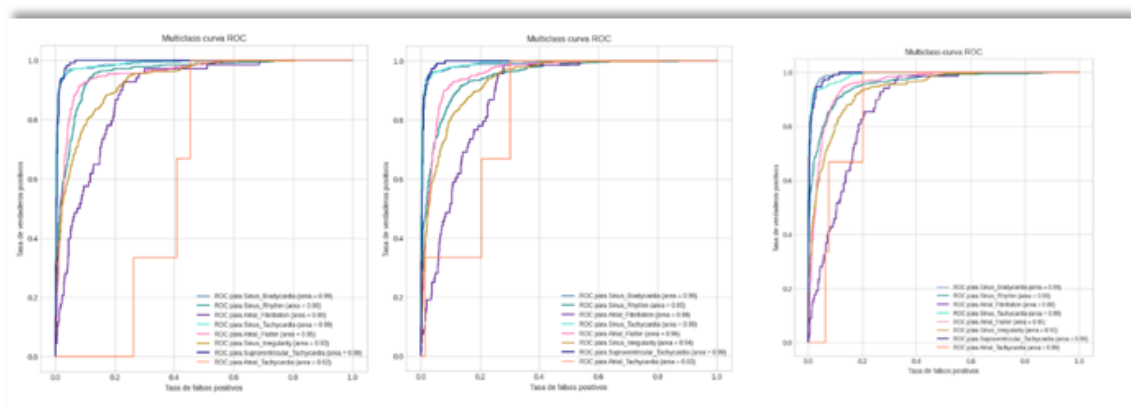


Tabla 11- Accuracy por derivación y curvas ROC

Desde un punto de vista más centrado en la propia Ciencia de Datos, señalar que **los algoritmos “colectivos”, como los basados en árboles múltiples (random forest o boosting), stacking o Inteligencia colectiva, presentan mejores tasas de acierto**, por lo general, y con una menor variabilidad en los resultados. Esto se puede apreciar en las tablas 3 y 4 y la ilustración 22 con las boxplot de cada modelo.

SVM, como grata sorpresa, también presenta buenos promedios, a pesar de ser aplicado a un conjunto de datos muy mezclado, como se aprecia de las gráficas UMAP, y por lo tanto, de difícil separación.

Los mejores modelos, como ya se ha indicado, para nuestro conjunto de datos, son:

- Support_Vector_Machine
- Random_Forest
- XGBoosting
- IC_Max_voting
- IC_Stacking

5.2. Sobre la explicabilidad

El objetivo principal del presente trabajo no consistía en encontrar el mejor modelo que detectara n tipos de arritmia. Hablábamos de explicabilidad, de poder interpretar los resultados más allá de la comprensión de los algoritmos en sí, basados en cálculos estadísticos y sujetos a cierta aleatoriedad.

Merece por tanto un apartado especial dentro de las Conclusiones.

Es bien sabido que, de entre los modelos expuestos, tan solo los Árboles de Decisión son susceptibles de ser explicados con sencillez, basados en divisiones de cada “hoja” en base a ciertos valores de algún atributo, resultando en unas reglas claras y de fácil aplicación a la hora de aplicar a nuevas instancias.

Para el resto de modelos, hemos utilizado LIME. Un modelo independiente del modelo que se quiera examinar y que nos devuelve la predicción “local” de una instancia y los atributos relevantes de la predicción del principal. Estas son las conclusiones extraídas:

- 1) Si el modelo principal está relativamente bien ajustado, la predicción de LIME coincide. Esto se puede ver en las ilustraciones 27, 30, 33, 36 y 39 de los modelos expuestos. Cuando el modelo principal no es preciso, ambas predicciones tienen a ser dispares y solo el ojo experto humano podrá analizar y ver cuál de ellos tiene razón.
- 2) La probabilidad del resultado dado por LIME, en las ilustraciones anteriormente mencionadas, varía. Lo que sugiere la posibilidad de combinar en un **sistema modelos+LIME** y emitir el resultado más preciso para ambos.

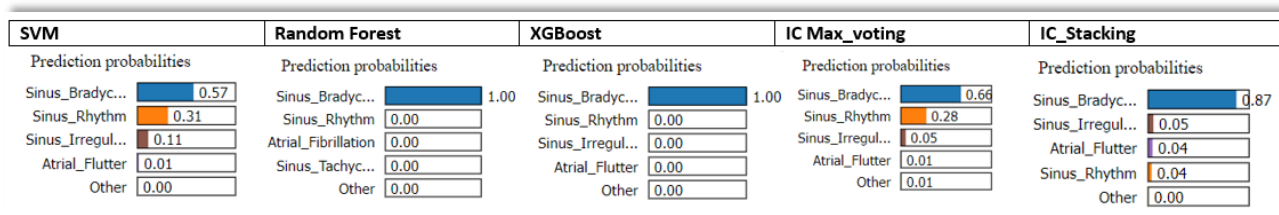


Ilustración 44- Comparativa de resultados LIME para ECG JS00010

- 3) No todos los atributos tienen la misma relevancia para cada modelo. En nuestro trabajo se aprecia la aparición de ciertos atributos en varios modelos (los relativos a la duración del latido), pero no en el mismo orden, ni con la misma ponderación, tomando relevancia otros de tipo amplitud. En el ejemplo de LIME ofrecido anteriormente para su comprensión ya se indicó que es el cardiólogo, con el contexto y, posiblemente, datos adicionales no incorporados en este trabajo (peso, altura, fumador...) quien debe evaluar si tiene sentido o no para el paciente, pero a priori todo hace pensar que, de entre todos los atributos, sí hay algunos más importantes y que son prioritarios para la mayoría de modelos. Asimismo, la depuración de atributos podría ser un enfoque para

mejorar el modelo principal y, a su vez, contrastarlo con LIME y esos expertos, para poner el foco en nuevos atributos que puedan cobrar relevancia.

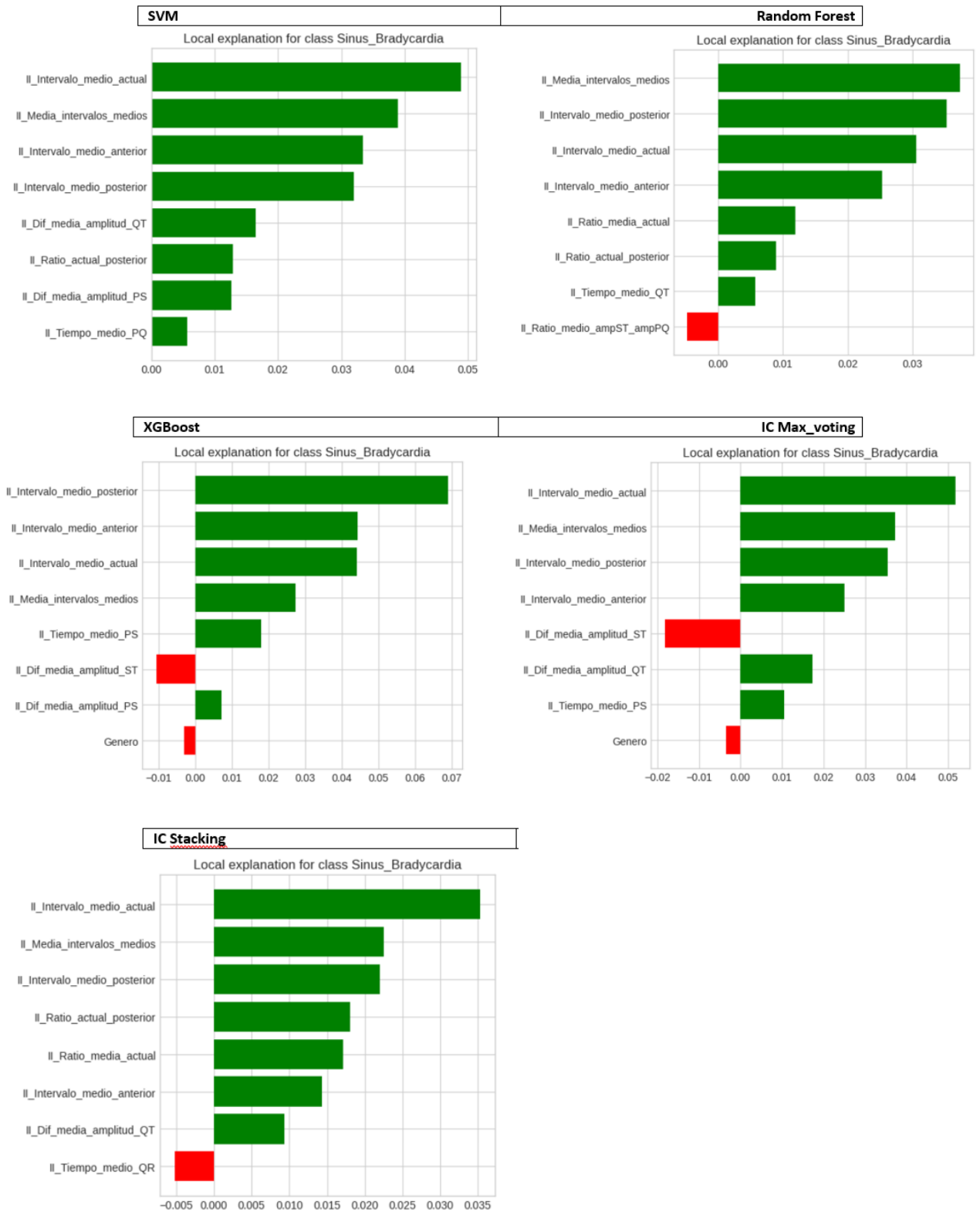


Ilustración 45- Atributos y relevancia según LIME para JS00010

5.3.Retos

Todas las tecnologías nuevas implican siempre problemas y retos. La aplicación de IA a la práctica médica, y concretamente, a la predicción de enfermedades cardiacas como las arritmias a través de los datos de un electrocardiograma, no es la excepción. Pero todos ellos son esperanzadores y como retos, nos permiten avanzar. Así lo recoge la RAE en la 5ª definición de la palabra reto:

“Objetivo o empeño difícil de llevar a cabo, y que constituye por ello un estímulo y un desafío para quien lo afronta”

Nos enfrentaremos, pues, a situaciones que no son fáciles de superar, pero sí son un estímulo gratificante.

Hemos hecho anteriormente referencia a problemas relacionados con los propios datos y las generalizaciones que derivan de los modelos y algoritmos, y también de los que hemos llamado “metainformación” y que va más allá del significado de los datos aislados y a los que aplicamos una visión social.

En el primer grupo de retos podríamos encontrarnos con los relacionados con la propia calidad de los datos y su posibilidad de corrupción. Los datos deben ser, por una parte, extraídos siguiendo técnicas que permitan corroborar su independencia, para su estudio estadístico posterior, con la precisión adecuada para el fin que se persiga. Pero también hay que garantizar su almacenaje correcto, sin pérdida o modificación de información. Incluso, se deberían establecer estándares robustos en cuanto la propia transmisión de los datos ya que, la variación de algunos simples bytes de manera aleatoria, podría afectar al resultado de los modelos. La variación en un byte en un electrocardiograma, inapreciable para el ojo humano, podría derivar en conclusiones erróneas. Hoy en día el almacenamiento en dispositivos y la transmisión de datos está más que asegurada, pero no por ello debemos obviarlo.

Debemos también utilizar datos correctos. Podemos disponer de decenas, centenas de atributos, pero no todos van a aportar información útil para la extracción de conclusiones. No debemos caer en la “maldición de la dimensionalidad” del machine learning y, puesto que podemos contar con un sinfín de características o atributos, pensar que todos aportan. Es preciso un análisis detallado del significado de cada uno de ellos, estudiar correlaciones y, en la medida de lo posible, simplificar. También de cara a la tan ansiada interpretabilidad, reto muy relevante a tener en cuenta. No solo cuenta la precisión, sino también, como veremos más adelante, que puedan ser explicadas las predicciones de la máquina.

Del mismo modo los datos deben ser robustos y tener la consistencia adecuada. Existe en el ámbito de la Inteligencia Artificial el dicho **“garbage in, garbage out”** (o una versión libre en castellano, “uno recoge lo que siembra”) que considero que es suficientemente explícito como

para tener que profundizar en ello. Tan solo remarcar que lo que obtengamos dependerá, en una gran medida, de lo que introduzcamos al modelo y de los datos y atributos que usemos (de su relevancia, de su tratamiento previo, de su precisión, de su calidad...).

Y tenemos también la cuestión del volumen de datos. No tanto el número de atributos y la dimensión, ya mencionado, sino de la cantidad de registros. En nuestro caso, de electrocardiograma y/o pacientes. Es evidente que, con conjuntos de datos mayores, podremos entrenar más y mejor a los distintos modelos (ya sean redes neuronales, árboles de decisión, o cualesquiera de las técnicas que usemos). De hecho, podríamos caer en una espiral virtuosa de forma que nuevos datos vayan siendo introducidos en un modelo aceptable y aceptado, de forma que éste se vaya ajustando y mejorando.

En el segundo grupo de desafíos, podríamos tener la creación de un marco legal (y ético) para respaldar la toma de decisiones clínicas basadas en Inteligencia Artificial. Este problema ya es mencionado por Konstantinos C. Siontis [\[7\]](#) en su artículo sobre el uso de los electrocardiogramas y la Inteligencia Artificial que ya hemos mencionado con anterioridad.

También encontramos en este conjunto de consideraciones relevantes asuntos como la anonimización de los datos, que tratándose de datos médicos son de carácter privado y se podría violar la intimidad y la confidencialidad entre paciente y doctor. La gestión de los sesgos se constituye en un pilar fundamental, tanto en la recopilación de los datos (para evitar discriminar a parte de la población por sexo, nivel social, color de piel, nacionalidad), como su tratamiento (proceso en el que los científicos de datos pueden influir en los resultados a través de las decisiones tomadas en función de sus puntos de vista).

Y por último estarían las cuestiones éticas, no menos relevantes, sobre todo cuando las tecnologías están en una fase incipiente de su desarrollo y no hay marcos sociales ni jurídicos que acoten sus impactos. Y lo cierto es que la tecnología siempre va por delante de las regulaciones y aceptaciones sociales, haciendo que estas tengan un carácter reactivo, y en el intervalo de tiempo hasta la aceptación y regulación se pueden cometer abusos que modifiquen incluso la sociedad o condicionen el propio hecho de construcción de la ética alrededor de la misma.

También es cierto que la tecnología modifica nuestros hábitos y condiciona nuestro día a día. Muchos comportamientos actuales eran impensables hace tan solo 10 años. Y esto puede estar sucediendo hoy con la práctica médica. ¿En qué medida un cardiólogo podría “delegar” el diagnóstico en una máquina que viene con unas premisas de precisión superiores a la suya? ¿Puede el diagnóstico de la máquina condicionar al del especialista? ¿Cómo de tranquilo estará el paciente sabiendo que su diagnóstico proviene de un algoritmo? Son algunas preguntas de difícil respuesta pero que ya se están analizando. Konstantinos C. Siontis [\[7\]](#) cita el proyecto EAGLE en el cual se trata de averiguar si los resultados de la Inteligencia Artificial afectan a los diagnósticos y, por lo tanto, a la práctica médica.

En nuestro trabajo hemos intentado dejar de lado, al menos de primeras, aquellos algoritmos que actúan como “caja negra” y si la precisión era aceptable, dar una **explicación automática pero basada en atributos**. Entendemos que, a igualdad de precisión, tendrán una mejor comprensión por parte de la sociedad y, por ende, una mayor penetración en la función médica, aquellos **modelos que sean explicables**.

Y es que, por muy certeros diagnósticos que faciliten los modelos, al margen de la interpretabilidad de los mismos, **el éxito de estas tecnologías está supeditado a la mejora de la práctica médica, sin dudas ni polémicas**, y a su aceptación plena por parte del personal médico y de los pacientes, puesto que está en juego la vida de la gente.

5.4. Trabajos futuros

La vida real es mucho más diversa y compleja, con gran variedad de individuos con sus circunstancias y sus condicionantes, que cualquier estudio controlado de laboratorio. La Inteligencia Artificial aplicada a los electrocardiogramas nos acerca a ese mundo real, puesto que se trata de una técnica no invasiva de uso conocido y fácil de realizar. Lo complicado, al margen de los retos ya descritos, es la interpretación de esos electrocardiogramas, pero hoy en día los modelos implementados por estas tecnologías, apoyados en esa infinidad de información, hacen posible la extracción de patrones para multitud de patologías cardiacas.

Queda mucho camino para asegurar la calidad de los datos, garantizar la representatividad de las muestras, seguir ampliando y mejorando los conjuntos de datos, pero el camino ya se ha iniciado. Existen, como se ha mencionado, enfoques muy variados, ideas razonables para explorar, diferentes datos en los que apoyarse para complementar la información de los ECG, diferentes técnicas con grandes posibilidades, y un abanico de problemas o patologías, con sus problemas, por solventar.

En este mismo trabajo ya se han planteado posibles ampliaciones y enfoques. Algunas ideas:

- Explorar LIME global.
- De confirmarse que una derivación ofrece resultados muy aceptables, casi idénticos a los que se desprenden de usar 12, se podrán incorporar “calculadoras” como las desarrolladas en los cuadernos Jupyter en distintos wearables, dando un impulso a la IoT-ECG, y confirmando su utilidad como herramienta médica.
- Se pueden combinar los mejores modelos de los cuadernos presentados en uno nuevo de Inteligencia Colectiva para aprovechar los mejores de cada uno de ellos... y de cada derivada. Es decir, podemos tomar el mejor modelo del cuaderno que utilizar todos los datos y combinarlo con el mejor de la derivada I, con el mejor de la II, etc... y es que las combinaciones son múltiples.
- O conociendo a los atributos relevantes que nos da LIME, tratar de comprender aquellos que hacen que alguna clase se clasifique mal para “afinar” el modelo

- Obtener nuevos electrocardiogramas, de otras fuentes, con las clases menos representadas en este estudio, “Atrial tachycardia” y “Atrial fibrillation”, y procesarlas con la calculadora creada, para enriquecer los modelos y mejorar los resultados.
- Metacombinación de modelos+LIME en sistema de Inteligencia Colectiva para aprovechar las bondades de ambos modelos en cuanto a precisión, probabilidad del resultado y atributos principales.
- Introducción de nuevos atributos en el dataset en detrimento de los menos representativos para LIME en un ciclo de mejora continua del modelo.

Pero, sea como sea, no podemos caer en la panacea de creernos los diagnósticos de la máquina, tanto en cuanto no se solventen los desafíos del desarrollo de estas tecnologías, de la misma manera que tenemos que ser cautos con la manera como estos **diagnósticos derivados de la IA-ECG pueden influenciar a médicos y pacientes**, sobre todo si los datos presentan sesgos que no se hayan detectado o eliminado.

Y es que estas tecnologías, como casi todas las que suponen algún tipo de disrupción, en algún momento son cuestionadas, pero debemos ser conscientes de que solo el uso que les demos es lo que determinará si es una buena herramienta, para salvar vidas, o para discriminar y dividir la sociedad... y es esta misma sociedad, su conjunto, la que decidirá si acaban por imponerse estas “ayudas médicas” y cómo se terminarán implantando.

El futuro de este campo de la investigación médica es prometedor. Los trabajos expuestos hasta el momento así lo atestiguan, con datos esperanzadores y precisiones superiores a las de los expertos.

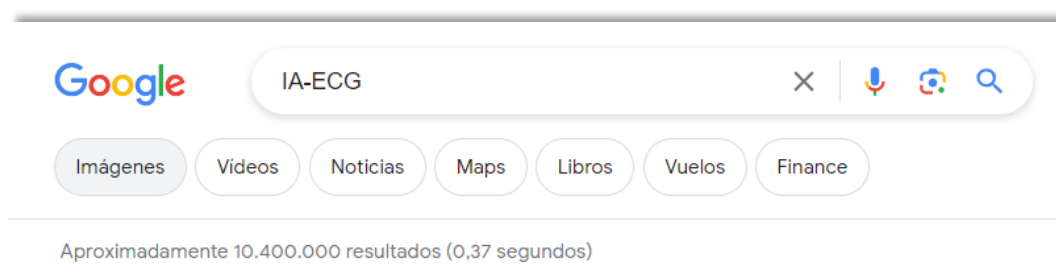


Ilustración 46- Interés que despierta la IA-ECG en Internet

Una simple búsqueda en Google de “IA-ECG” nos arroja más de 10 millones de resultados. Son muchos los equipos trabajando en ello. Y son muchos los ámbitos médicos donde estas tecnologías pueden contribuir de manera muy positiva para prevenir problemas cardiacos y salvar vidas.

El futuro promete y si los resultados se pueden explicar, más aún...

6. Agradecimientos

Para la realización de este trabajo he contado con la inestimable ayuda de mi tutor, **Carlos Luis Sánchez Bocanegra**, dispuesto a reunirse a cualquier hora y atento a todas las cuestiones del trabajo.

También quiero agradecer al profesor **Albert Solé Ribalta** por esa cobertura necesaria cuando ha hecho falta.

Y un agradecimiento especial al **Doctor Carlos José Jaramillo Gómez** que, con gran amabilidad y paciencia, ha puesto a mi disposición su experiencia y conocimientos de cardiólogo y sin la cual no habría podido pasar de la representación de electrocardiogramas. Gran doctor, gran profesor y mejor persona.

Muchas gracias a los tres.

No quiero olvidarme de mi familia. Son ellos quienes han tenido que aguantar mis largas sentadas de estudio y programación, y soportarme, en todos los significados de la palabra, cuando las cosas no salían como esperaba.

Gracias.

7. Glosario

Aleteo Auricular/Atrial. Ritmo anormal del corazón caracterizado por contracciones rápidas y regulares de las aurículas.

Amiloide. Proteínas que se pliegan anormalmente y se agrupan en estructuras insolubles. Estas proteínas pueden acumularse en diferentes órganos o tejidos del cuerpo, causando diversas enfermedades, como la amiloidosis, donde estas acumulaciones afectan la función normal de los órganos.

Amiloidosis. Trastorno en el que proteínas anormales llamadas *amiloides* se acumulan en órganos y tejidos del cuerpo, afectando su función normal.

Arritmia. Alteración en el ritmo cardíaco, donde el corazón late demasiado rápido, demasiado lento o de manera irregular.

AUC. "Area Under the Curve" (Área Bajo la Curva). Término comúnmente usado en estadística y aprendizaje automático para evaluar la precisión de un modelo predictivo en un gráfico de curva ROC (Receiver Operating Characteristic). Representa la medida de la capacidad del modelo para distinguir entre clases.

Big Data. Conjuntos de datos extremadamente grandes y complejos que tradicionalmente superan las capacidades de manejo de herramientas convencionales de procesamiento de datos. Incluye aspectos como la recopilación, almacenamiento, procesamiento y análisis de información a gran escala para extraer conocimientos, patrones o tendencias que podrían ser difíciles de obtener con métodos tradicionales.

Bradicardia Sinusal. Condición médica caracterizada por un ritmo cardíaco más lento de lo normal, generalmente menos de 60 latidos por minuto, donde el nodo sinusal, el "marcapasos" natural del corazón, genera impulsos a un ritmo más lento de lo habitual.

Caja negra (modelo de). Enfoque en el que se considera un sistema, proceso o modelo desde fuera, sin necesariamente conocer o entender en detalle su funcionamiento interno. Se observan las entradas y salidas del sistema sin conocer la lógica interna o los mecanismos específicos que ocurren dentro de él.

Chunk. En informática y procesamiento de datos, un "chunk" se refiere a una porción o bloque de datos o de código que ha sido dividido para su manejo más eficiente.

Clustering. Técnica de aprendizaje no supervisado en la que se agrupan datos similares entre sí en conjuntos o clústeres, basándose en características comunes, con el objetivo de encontrar patrones intrínsecos o estructuras ocultas en los datos.

CRISP-DM. (Cross-Industry Standard Process for Data Mining) Modelo estándar utilizado en minería de datos y análisis predictivo. Define un proceso en seis etapas: comprensión del negocio, comprensión de los datos, preparación de los datos, modelado, evaluación y despliegue, para guiar a los profesionales en proyectos de minería de datos.

Cuaderno Jupyter. Aplicación web interactiva que permite crear y compartir documentos que contienen código en vivo, visualizaciones, texto explicativo y más. Es ampliamente utilizado en ciencia de datos y programación para explorar y analizar datos, realizar experimentos y crear informes interactivos.

Derivación. Cada uno de los distintos ángulos de registro de la actividad eléctrica del corazón. Estas derivaciones representan las perspectivas desde las cuales se capturan las señales eléctricas del corazón en el ECG para evaluar la actividad cardiaca desde diferentes ángulos y posiciones en el cuerpo.

Dofetilida. Medicamento antiarrítmico que se utiliza para tratar ciertos tipos de arritmias cardíacas, específicamente para restaurar y mantener un ritmo cardíaco normal en pacientes con fibrilación auricular paroxística. Ayuda a regular los latidos del corazón al influir en el flujo de iones en las células del músculo cardíaco.

Ecocardiogramas. Prueba no invasiva que utiliza ondas de ultrasonido para crear imágenes en tiempo real del corazón. Se utiliza para evaluar la estructura, función, tamaño y movimiento de las diferentes partes del corazón, incluyendo las válvulas y las cámaras cardíacas, lo que permite diagnosticar afecciones cardíacas y evaluar su gravedad.

Electrocardiogramas (ECG). Prueba médica que registra la actividad eléctrica del corazón a través de electrodos colocados en la piel del paciente. Este examen muestra la representación gráfica de los impulsos eléctricos del corazón, lo que permite evaluar el ritmo, la velocidad y la regularidad del latido cardíaco, además de detectar posibles anomalías cardíacas.

Ensemble-learning. Técnica en el aprendizaje automático donde se combinan múltiples modelos de aprendizaje para mejorar la precisión y robustez general del sistema.

Especificidad. Medida que indica la capacidad de una prueba para identificar correctamente a los verdaderos negativos. Es la proporción de los casos negativos que son correctamente identificados como negativos por la prueba o modelo. En resumen, representa la capacidad de evitar falsos positivos.

Falso negativo. Resultado de una prueba diagnóstica o de un modelo predictivo que indica incorrectamente la ausencia de una condición o característica cuando en realidad está presente. En otras palabras, es cuando la prueba o el modelo no detecta algo que está presente en realidad, arrojando un resultado negativo cuando debería haber sido positivo.

Falso positivo. Un falso positivo ocurre cuando una prueba diagnóstica o un modelo predictivo indica incorrectamente la presencia de una condición o característica cuando en realidad está ausente. Es decir, es un resultado positivo que se produce erróneamente al identificar algo como presente cuando en realidad no lo está.

Fibrilación auricular/atrial. Arritmia cardíaca común y potencialmente grave caracterizada por latidos cardíacos irregulares y rápidos en las aurículas, las dos cavidades superiores del corazón. En la fibrilación auricular, las aurículas no se contraen de manera coordinada, lo que puede resultar en un flujo sanguíneo irregular y aumentar el riesgo de coágulos sanguíneos, accidentes cerebrovasculares y otros problemas cardíacos.

Fibrilación auricular silente. Casos de fibrilación auricular que ocurren sin síntomas aparentes o sin que la persona afectada sea consciente de la arritmia. Aunque la fibrilación auricular puede ser asintomática en algunos individuos, sigue presentando un riesgo de complicaciones graves, como accidentes cerebrovasculares, por lo que su detección y manejo adecuados son importantes.

Filtrado Butterworth. Tipo de procesamiento que permite el paso de frecuencias dentro de una banda determinada mientras atenúa las frecuencias fuera de esa banda de manera gradual y sin generar distorsión en la fase de la señal.

Fotopleletismografía. Técnica no invasiva que mide los cambios en el volumen sanguíneo en tejidos periféricos mediante la detección de la absorción de luz. Usualmente se emplea para medir el pulso y la frecuencia cardíaca, y se realiza colocando un sensor en la piel, generalmente en el dedo o en la muñeca.

Fracción de eyección del ventrículo izquierdo. Medida que indica la eficiencia con la que el ventrículo izquierdo del corazón bombea sangre hacia el cuerpo.

F1-score. Métrica de evaluación en machine learning que combina la precisión y la exhaustividad de un modelo en una sola medida numérica, útil para determinar su rendimiento en la clasificación de datos. El F1-score es la media armónica de precisión y exhaustividad.

$$F_1 = 2 \times \frac{\text{Precisión} \times \text{Exhaustividad}}{\text{Precisión} + \text{Exhaustividad}}$$

Donde:

- $\text{Precisión} = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Positivos}}$
- $\text{Exhaustividad} = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Negativos}}$

Hiperparámetro. Parámetro ajustable que permite controlar el proceso de entrenamiento de un modelo.

IA-ECG. Inteligencia artificial (IA) aplicada al análisis de electrocardiogramas (ECG). La IA en el campo de la medicina, particularmente en el análisis de ECG, ha ganado relevancia debido a su capacidad para detectar patrones, anomalías y enfermedades cardíacas de manera más eficiente y precisa que los métodos tradicionales.

IoT. Red de dispositivos interconectados a través de internet que recopilan y comparten datos para automatizar procesos y mejorar la eficiencia en diversos ámbitos, desde el hogar hasta la industria o la medicina.

Irregularidad sinusal. Condición cardíaca que se caracteriza por cambios en el ritmo normal del nodo sinusal. Esto puede resultar en variaciones en la frecuencia cardíaca, lo que provoca un latido cardíaco irregular y poco uniforme.

Kanban. Método visual para gestionar el flujo de trabajo, especialmente utilizado en la gestión de proyectos y procesos. Consiste en tableros visuales con tarjetas o post-its que representan tareas o elementos de trabajo, mostrando su progreso a través de diferentes etapas o columnas (por ejemplo, "Por hacer", "En progreso", "Hecho"). Este enfoque ayuda a visualizar y controlar el flujo de trabajo, identificar cuellos de botella y optimizar la eficiencia.

Maldición de la dimensionalidad. Concepto del aprendizaje automático que se refiere a los problemas que surgen cuando se trabaja con conjuntos de datos de alta dimensionalidad, es decir, con muchas características, que pueden dificultar el análisis, la interpretación y la precisión de los modelos.

Metodologías ágiles. Enfoque para el desarrollo de software que prioriza la flexibilidad, la colaboración y la adaptabilidad a lo largo del proceso de desarrollo. Se basa en principios como la entrega iterativa e incremental, la retroalimentación continua, la colaboración estrecha entre equipos multidisciplinarios y la capacidad de responder a los cambios.

Modelo de Inteligencia Colectiva. Enfoque basado en la idea de que la colaboración y la diversidad de modelos pueden conducir a decisiones más acertadas, innovación y resolución de problemas.

Nodo sinusal. Estructura del corazón que actúa como un marcapasos natural. Se encuentra en la aurícula derecha y es responsable de generar los impulsos eléctricos que regulan el ritmo cardíaco, iniciando cada ciclo de latido cardíaco.

Oversampling. Técnica utilizada para ajustar la distribución de clases de un conjunto de datos, aumentando la presencia de las clases menos representadas con copias iguales, con algunas transformaciones, hasta igualar o aproximar su número de instancias al resto de clases.

PhysioNet. Repositorio en línea de datos fisiológicos y herramientas relacionadas utilizadas por la comunidad científica para compartir, acceder y analizar datos biomédicos, como señales fisiológicas, registros de electrocardiogramas, datos de monitorización, entre otros, con el objetivo de fomentar la investigación y el desarrollo de métodos en la ciencia médica y la ingeniería biomédica.

Python. Lenguaje de programación de alto nivel.

Recall. También conocido como recuperación o exhaustividad, es una métrica en estadísticas y aprendizaje automático que mide la proporción de instancias positivas que un modelo identifica correctamente sobre el total de instancias positivas reales.

$$\text{Recall} = \frac{\text{Verdaderos Positivos}}{\text{Verdaderos Positivos} + \text{Falsos Negativos}}$$

Red neuronal convolucional (CNN). Tipo de red neuronal artificial eficaz en el procesamiento y análisis de datos como imágenes o señales temporales. Están compuestas por capas para extraer y aprender características significativas de los datos de entrada.

Reducción de la dimensionalidad. Proceso de disminuir el número de variables o dimensiones en un conjunto de datos, preservando la información más relevante para simplificar y facilitar su análisis o visualización.

Ritmo sinusal. Ritmo cardíaco regular y coordinado entre las aurículas y los ventrículos.

ROC. "Receiver Operating Characteristic". Representación gráfica que muestra el desempeño de un modelo de clasificación, representando la tasa de verdaderos positivos frente a la tasa de falsos positivos en diferentes puntos de corte

Scrum. Marco de trabajo ágil utilizado en el desarrollo de software y en otros proyectos, que se basa en iteraciones cortas llamadas "sprints". Se enfoca en la colaboración, la flexibilidad para adaptarse a cambios y la entrega continua de productos de valor.

Sensibilidad. Medida que evalúa la capacidad de un test o modelo para identificar correctamente los casos positivos entre todos los casos verdaderamente positivos.

Spike. Trabajo de investigación, exploración o experimentación breve y limitado en el tiempo, realizado por un equipo para obtener información, resolver problemas técnicos o investigar soluciones antes de comprometerse con una tarea más extensa. Los "spikes" ayudan a reducir incertidumbres y riesgos antes de abordar el desarrollo de una funcionalidad o característica específica en un proyecto.

Taquicardia auricular/atrial. Ritmo cardíaco anormal caracterizado por la generación rápida y desorganizada de impulsos eléctricos en las aurículas del corazón.

Taquicardia sinusal. Ritmo cardíaco rápido, pero regular.

Taquicardia supraventricular. Término que describe varios tipos de ritmos cardíacos anormales y rápidos que se originan por encima de los ventrículos. Incluye la taquicardia auricular, la taquicardia nodal AV y otros tipos específicos de taquicardias paroxísticas.

Webscrapping. Técnica de extracción automática de datos de páginas web, permitiendo recopilar información de manera estructurada para su análisis o uso posterior.

4G. Cuarta generación de estándares de comunicaciones móviles. Ofrece velocidades de conexión a internet más rápidas y mejoras significativas en la capacidad de transmisión de datos en comparación con las generaciones anteriores (3G, 2G, etc.).

5G. Quinta generación de estándares de comunicaciones móviles. Ofrece avances significativos en la velocidad, la capacidad y la latencia de la red, permitiendo velocidades de conexión a internet considerablemente más rápidas que en generaciones anteriores. El 5G tiene el potencial de revolucionar las comunicaciones al admitir una mayor cantidad de dispositivos conectados, impulsar la Internet de las Cosas (IoT), mejorar la transmisión de datos para aplicaciones exigentes, como la realidad virtual y aumentada, y habilitar tecnologías emergentes como vehículos autónomos y ciudades inteligentes.

8. Bibliografía

- [1] Goto S, Mahara K, Beussink-Nelson L, et al. Artificial intelligence-enabled fully automated detection of cardiac amyloidosis using electrocardiograms and echocardiograms. *Nat Commun.* 2021;12(1):2726. <https://doi.org/10.1038/s41467-021-22877-8>
- [2] Tison GH, Zhang J, Delling FN, Deo RC. Automated and interpretable patient ECG profiles for disease detection, tracking, and discovery. *Circ Cardiovasc Qual.* 2019;12:e005289. <https://doi.org/10.1161/CIRCOUTCOMES.118.005289>
- [3] Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., ... & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation [Online]*. 101 (23), pp. e215–e220.
- [4] Moody, G., Pollard, T., & Moody, B. (2022). WFDB Software Package (version 10.7.0). PhysioNet. <https://doi.org/10.13026/gjvw-1m31>
- [5] Zheng, J., Guo, H., & Chu, H. (2022). A large scale 12-lead electrocardiogram database for arrhythmia study (version 1.0.0). PhysioNet. <https://doi.org/10.13026/wgex-er52>
- [6] Ribeiro MT, Singh S, Guestrin C. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. En: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; 13-17 de agosto de 2016; San Francisco, CA, EE. UU. 2016. pp. 1135-1144. <https://doi.org/10.48550/arXiv.1602.04938>
- [7] Siontis KC, Noseworthy PA, Attia ZI, Friedman PA. Artificial intelligence-enhanced electrocardiography in cardiovascular disease management. *Nat Rev Cardiol.* 2021;18(7):465–78. <https://doi.org/10.1038/s41569-020-00503-2>
- [8] Prifti E, Fall A, Davogustto G, Pulini A, Denjoy I, Funck-Brentano C, Khan Y, Durand-Salmon A, Badilini F, Wells QS, Leenhardt A, Zucker JD, Roden DM, Extramiana F, Salem JE. Deep learning analysis of electrocardiogram for risk prediction of drug-induced arrhythmias and diagnosis of long QT syndrome. *Eur Heart J.* 2021;42(38):3948-3961. <https://doi.org/10.1093/eurheartj/ehab588>

- [9] Krittanawong C, Rogers AJ, Johnson KW, et al. Integration of novel monitoring devices with machine learning technology for scalable cardiovascular management. *Nat Rev Cardiol.* 2021;18:75-91. <https://doi.org/10.1038/s41569-020-00445-9>
- [10] Hannun AY, Rajpurkar P, Haghpanahi M, et al. Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. *Nat Med.* 2019;25:65-69. <https://doi.org/10.1038/s41591-018-0268-3>
- [11] Hughes JW, Olgin JE, Avram R, Abreau SA, Sittler T, Radia K, Hsia H, Walters T, Lee B, Gonzalez JE, Tison GH. Performance of a Convolutional Neural Network and Explainability Technique for 12-Lead Electrocardiogram Interpretation. *JAMA Cardiol.* 2021;6(11):1285-1295. <https://doi.org/10.1001/jamacardio.2021.2746>
- [12] Vaswani A, Shazeer NM, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I. "Attention is All you Need." *Neural Information Processing Systems.* 2017. <https://doi.org/10.48550/arXiv.1706.03762>
- [13] Pedregosa F, Varoquaux, Ga"el, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine learning in Python. *Journal of machine learning research.* 2011;12(Oct):2825–30.
- [14] Pan JP, Tompkins WJ. A Real-Time QRS Detection Algorithm. *IEEE Transactions on Biomedical Engineering.* 1985; BME-32(3):230-236. <https://doi.org/10.1109/TBME.1985.325532>
- [15] Elgendi M, Jonkman M, De Boer F. Frequency bands effects on QRS detection. In *Biosignals 2010 – Proceedings of the 3rd International Conference on Bio-inspired Systems and Signal Processing*, volume 1. 2010; 428–431. <https://doi.org/10.5220/0002742704280431>
- [16] Porr B, Howell L, Stournaras I, Nir Y. A collection of ECG heartbeat detection algorithms implemented in Python. [Internet]. Disponible en: <https://doi.org/10.5281/zenodo.3353396>
- [17] Vollmer M, Giraldo Guzmán JA. Efficiency of Different Heartbeat Detection Methods by Using Alternative Noise Reduction Algorithms. In *2022 Computing in Cardiology (CinC); 2022; Tampere, Finlandia.* 2022. pp. 1-4. <https://doi.org/10.22489/CinC.2022.034>

9. Anexo 1 - Filtrado de electrocardiogramas

Para conformar el conjunto de datos de trabajo, cada electrocardiograma se ha sometido a un procesamiento de filtrado y extracción de atributos.

El procesado ha consistido en eliminar ruidos, bajas frecuencias y picos de señal mediante el algoritmo de Pan Tompkins [14] que consiste en un filtrado pasa-banda para atenuar las bajas frecuencia de las ondas P Y T del ECG, oscilaciones e interferencias de la red y trata la señal para acentuar las frecuencias altas correspondientes a la onda QRS (ilustración 20).

Seguidamente, ya estamos en disposición de extraer las características de cada derivación, las mismas para cada una de ellas, y que estudiaremos tanto en conjunto como por separado para ver posibles relaciones entre derivaciones como el impacto que tienen los datos extraídos en los modelos de clasificación. Para esta labor utilizaremos la implementación de Elgendi, Mohamed & Jonkman et al. [15] proporcionada en la librería *py-ecg-detectors* [16] y que consiste en la detección de las ondas mediante filtros de media móvil. También enfatiza más las ondas débiles, como la P o la T, con lo que su identificación dentro de cada latido es más factible.

Los métodos y técnicas utilizadas para el filtrado y preprocesado de la señal son sólidos y reconocidos, y hecho se citan en el artículo de M.Vollmer et al. "Efficiency of Different Heartbeat Detection Methods by Using Alternative Noise Reduction Algorithms" [17] como eficientes estando el de Pan Tompkins entre los de mejor calificación en términos de sensibilidad y Tasas de verdaderos positivos.

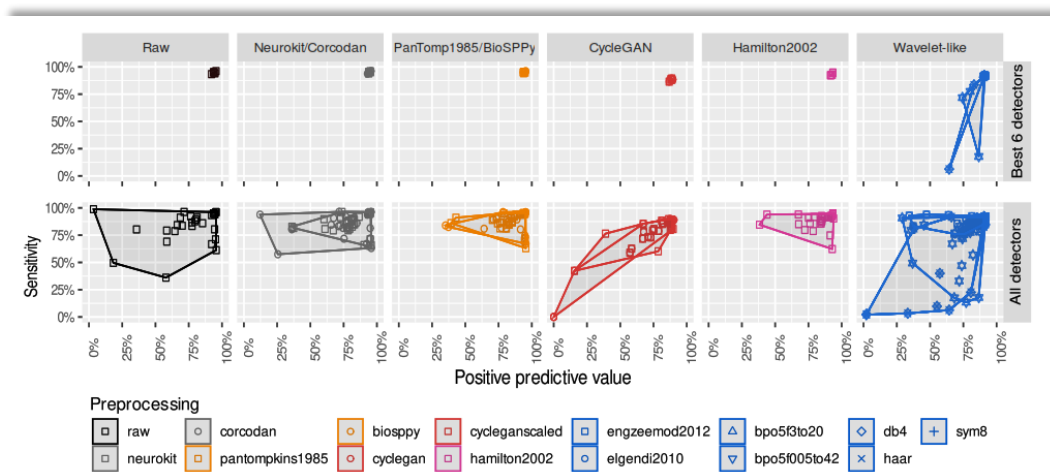


Ilustración 47- Gráficas comparativas de sensibilidad-valor predictivo positivo del trabajo de M.Vollmer

El resto de técnicas apuntadas en el artículo anterior, también con buenos resultados, consisten en establecer distintos tipos de filtrado por frecuencias (*filtrados Butterworth* de diferente orden) o añadir pasos en el procesamiento como eliminar la línea base con estimaciones lineales o aplicar distintos cortes de rechazo, entre otros. Nos decantamos por Pan Tomkins [14] y Elgendi [15] principalmente por los menores tiempos de ejecución y por una menor tasa de nulos entre los datos que generaban para nuestro dataset.

10. Anexo 2 – Referencias de interés

Chapman P, Clinton J, Kerber R, editors. CRIPS-DM 1.0. The CRISP-DM consortium. 2000.

Chollet F, editor. Deep learning with Python. Manning Publications Co; 2021.

Gironés Roig J, Casas Roma J, Minguillón J, Caihuelas Quiles R (eds). Minería de datos: modelos y algoritmos. Editorial UOC; 2017. ISBN 978-84-9116-903-1

Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., ... & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation* [Online]. 101 (23), pp. e215–e220.

Hearty J, editor. Advanced Machine Learning with Python. Packt Publishing; 2016

Lekadir K, Quaglio G, et al. Artificial intelligence in healthcare: Applications, risks, and ethical and societal impacts. ISBN: 978-92-846-9456-3. doi:10.2861/568473. Disponible en: <https://op.europa.eu/en/publication-detail/-/publication/958117aa-0c91-11ed-b11c-01aa75ed71a1/language-en>

Raschka S, editor. Python machine learning: unlock deeper insights into machine learning with this vital guide to cutting-edge predictive analytics. Packt Publishing; 2015.

Zheng, J., Chu, H., Struppa, D. et al. Optimal Multi-Stage Arrhythmia Classification Approach. *Sci Rep* 10, 2898 (2020). <https://doi.org/10.1038/s41598-020-59821-7>

Wagner P, Strodthoff N, Boussejot RD, et al. PTB-XL, a large publicly available electrocardiography dataset. *Sci Data*. 2020;7:154. doi:10.1038/s41597-020-0495-6

11. Anexo 3 – Librerías Python

Se describen a continuación las librerías Python utilizadas del ámbito médico.

wfdb → visualización, análisis y grabación de señales fisiológicas

Moody, G., Pollard, T., & Moody, B. (2022). WFDB Software Package (version 10.7.0). PhysioNet. <https://doi.org/10.13026/gjvw-1m31>

#####

neurokit2 → paquete de procesamiento de señales biológicas

Makowski, D., Pham, T., Lau, Z. J., Brammer, J. C., Lespinasse, F., Pham, H., Schölzel, C., & Chen, S. A. (2021). NeuroKit2: A Python toolbox for neurophysiological signal processing. Behavior Research Methods, 53(4), 1689-1696. <https://doi.org/10.3758/s13428-020-01516-y>

#####

biobss → paquete de procesamiento de señales de wearables

Taçci Ç, Karakuş İ, Çavuşoğlu D, Akyön FC. A package for processing signals recorded using wearable sensors, such as Electrocardiogram (ECG), Photoplethysmogram (PPG), Electrodermal activity (EDA) and 3-axis acceleration (ACC). [Internet]. Disponible en: <https://github.com/obss/BIOBSS>.

#####

py-ecg-detectors → implementación de algoritmos de detección de ondas

Porr B, Howell L, Stournaras I, Nir Y. A collection of ECG heartbeat detection algorithms implemented in Python. [Internet]. Disponible en: <https://doi.org/10.5281/zenodo.3353396>

