

# Clasificación de Condiciones Cardiacas Utilizando Métodos de Machine Learning

Jared Andrés Silva Villa, Paola Félix Torres<sup>□</sup>, Herbert Eduardo Euroza Hernández<sup>□</sup>, and Alma Paulina González Sandoval<sup>□</sup>

Instituto Tecnológico y de Estudios Superiores de Monterrey  
Guadalajara Jal., México



**Keywords:** Modelos predictivos · Electrocardiogramas · Machine Learning · Salud

**Abstract.** Este estudio se centra en la clasificación de señales de electrocardiogramas (ECG) mediante técnicas de modelados primitivos para mejorar la precisión diagnóstica en la detección de diversas condiciones cardiacas. La metodología involucró una estrategia de doble modelo, en el que el primero modelo clasifica ECGs del tipo normales o anormales, utilizado dos conjuntos de datos `ptbdNormal.csv` y `ptbdAnormal.csv`, mientras que el segundo modelo, aplicado a casos anormales, distingue entre cuatro clases específicas de anormalidades. Este enfoque busca mitigar los desafíos del desequilibrio de datos y el sesgo. La evaluación del regimiento indicó que Random Forest fue el modelo más efectivo, demostrada robustez y precisión a la hora de clasificar señales. El estudio destaca la importancia de preprocesamiento de datos, la optimización de hiperparámetros en el desarrollo de modelos predictivos confiables. Los resultados sugieren que combinar modelos puede mejorar significativamente el rendimiento de la clasificación, ofreciendo beneficios potenciales para la detección temprana de enfermedades cardíacas.

## 1 Introducción

Las enfermedades cardiovasculares (ECV) representan la principal causa de muerte a nivel mundial, siendo responsables de aproximadamente 16.7 millones de fallecimientos anuales. Con el envejecimiento de la población y el aumento de factores de riesgo como los estilos de vida poco saludables, se espera que la incidencia de ECV siga incrementándose en las próximas décadas. En este contexto, el electrocardiograma (ECG) ha emergido como una de las herramientas más utilizadas para la detección temprana de anomalías cardíacas, debido a su naturaleza no invasiva, accesibilidad y bajo costo. [6]

El ECG permite identificar diversas afecciones cardíacas analizando la actividad eléctrica del corazón. Sin embargo, la correcta interpretación del ECG sigue siendo un reto, ya que la presencia de ruido y artefactos puede afectar su precisión. Además, la habilidad clínica de los médicos para interpretar estas señales es frecuentemente limitada, lo que ha motivado el desarrollo de herramientas automáticas que mejoren el diagnóstico. [6]

En los últimos años, el uso de técnicas de machine learning ha revolucionado el análisis del ECG, permitiendo la clasificación automática de latidos cardíacos y la detección precisa de anomalías. Estas técnicas emplean algoritmos avanzados de procesamiento de señales y extracción de características para identificar patrones en las señales del ECG que, de otro modo, serían difíciles de reconocer manualmente. Modelos de aprendizaje profundo, como las redes neuronales y los sistemas de clasificación supervisada, han mostrado ser especialmente eficaces en el análisis de señales complejas, proporcionando resultados prometedores en términos de precisión, sensibilidad y especificidad. [1]

Este trabajo explora un enfoque basado en machine learning para la clasificación automática de latidos cardíacos, con el objetivo de mejorar el diagnóstico de las ECV y optimizar la toma de decisiones clínicas. Asimismo, se analiza el impacto de la incorporación de técnicas de procesamiento de señales y aprendizaje automático en la calidad y eficiencia de los sistemas de diagnóstico.

## 2 Metodología

Para completar el objetivo establecido y los materiales en mano, detallamos la secuencia de trabajo seguida para la realización de este proyecto. Primeramente se hace un análisis exploratorio de los datos, verificando la cantidad de clases presentes por cada tipo de señal cardíaca. Un modelo de clasificación debe ser entrenado con un número igual de observaciones por clase, ya que de ser caso contrario, el modelo aprenderá más sobre la clase que tenga más observaciones y aprender muy poco sobre las que tiene muy pocas. De este modo se genera un modelo que es bueno para predecir solo un tipo de clase.

Con los datos ya cargados, se obtiene el conteo por clase de los datasets y se verifica si se necesita hacer un balance de las clases presentes. En caso de que sea necesario se aplican técnicas de muestreo para que todas las clases tengan el mismo número de observaciones.

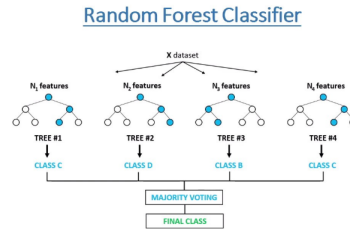
Una vez se cuenta con las clases balanceadas, se instancian varios modelos de clasificación para comprobar cuales son los más efectivos para el conjunto de datos a utilizar, y se utiliza validación cruzada para confirmar que los modelos sean buenos al generalizar los datos, intentando prevenir el sobreajuste de los modelos seleccionados. Se pretende prevenir el sobreajuste por que al estar sobre ajustados los modelos se vuelven muy buenos al predecir los datos proporcionados pero reduce su rendimiento al darle datos nuevos. Al generalizar los

modelos, estos rinden de manera apropiada al darle nuevos datos que clasificar, que es últimamente lo que necesita para un problema de este tipo. [2]

Después de evaluar los modelos seleccionados, se escoge uno para poder llevarlo a la etapa de mejora. En este caso se utilizará el modelo de Random Forest. Este modelo se basa en la idea de ensamble donde múltiples modelos simples llegan a una solución de mejor calidad a diferencia de hacer las predicciones en solitario. Random Forest, o RF, es un modelo de ensamble de árboles de decisión, consistiendo de miles de árboles de decisión que forman este “bosque” que el nombre del modelo indica. En ensamble, las categorizaciones erróneas son pocas en comparación al número de predicciones correctas, haciendo que el modelo haga sea correcto por agregación.[4]

RF utiliza esta cantidad inmensa de árboles para promediar, en caso de regresión, o sacar un voto mayoritario, cuando se trata de clasificación. Un árbol de decisión es una estructura de árbol simple donde el nodo hasta arriba se considera la raíz que es particionando de manera recursiva hasta llegar al nodo de decisión. Este algoritmo, por su simplicidad, se considera algoritmos avaros que divide los datos de entrenamiento en grupos más pequeños. El nodo raíz en este caso se refiere al mejor predictor y se divide en dos ramas cada vez hasta llegar a los nodos de decisión que es donde se realiza la predicción. Utilizar algoritmos de árbol para el entrenamiento supone diversas ventajas como ser eficientes al tener muchos datos, poder evaluar tanto variables cuantitativas como cualitativas, o tener muy pocos parámetros que considerar. El problema de los árboles de decisión recae en que son muy susceptibles a tener un mal rendimiento y a ser sobre ajustados. RF soluciona este problema al usar un ensamble de árboles donde los valores de un árbol son aleatorios e independientes. [5]

Además de esto, RF implementa una idea más para generar un modelo más eficaz. Adicional a particionar los datos en grupos más pequeños, separa los árboles usando un número aleatorio de predictores para ese grupo de datos asignado, como se puede observar en la Figura 1. De este modo, RF se vuelve un modelo bastante robusto y fácil de implementar. [3]



**Fig. 1.** Ejemplo de ensamble de arboles (RF), utilizando un número aleatorio de predictores  $N_i$ , para llegar a una mayoría de voto

Al haber seleccionado el modelo a utilizar se lleva a la fase de mejora, donde se ajustan los hiperpárametros del modelo, probando con diferentes configuraciones del mismo modelo para comparar las mejoras o falta de las mismas a diferencia de los resultados del modelo base. Al obtener los resultados, se selecciona la mejor configuración de hiperpárametros y se ajusta el modelo al conjunto de datos completo. De esta manera se consigue un modelo óptimo para la problemática que se está tratando.

### 3 Experimentos

#### 3.1 Datos Utilizados

El proyecto utilizó cuatro conjuntos de datos en formato SV:ptdbNormal.csv, ptdAbnormal.csv, mitbhTrain.csv y mitbhTest.csv. Estos conjuntos contiene registros de electrocardiograma con 187 características cada uno. La columna 187 representa la variable objetivo (Y), que toma los valores entre 0 y 4 representando las distintas clases de condiciones cardiacas:

- **0**: Ritmo sinusal normal (N).
- **1**: Arritmias supraventriculares (S).
- **2**: Desconocido o no clasificado (Q).
- **3**: Arritmias ventriculares (V).
- **4**: Latidos de fusión (F).

#### 3.2 Configuración de Hardware y Software

Los experimentos se desarrollaron utilizando Google Colab, una plataforma basada en la nube que proporciona un ambiente óptimo para este tipo de aplicaciones basadas en el lenguaje de programación Python. Las especificaciones de hardware utilizadas fueron las siguientes:

- Laptop con 16GB de RAM.
- Procesador Intel Core i5.
- Conexión a internet estable para la ejecución de Google Colab y acceso a las bibliotecas especializadas.

El lenguaje de programación utilizado en este proyecto fue Python 3, con las siguientes bibliotecas incluidas:

- **Scikit-learn** para la implementación de modelos.
- **Pandas** para el manejo y manipulación de los datos.
- **Matplotlib** para la visualización de los resultados.

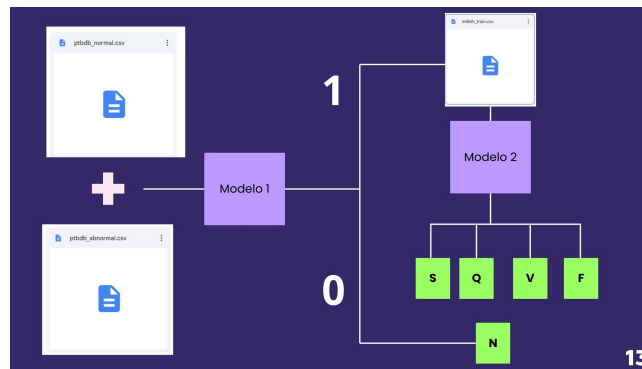
### 3.3 Preprocesamiento de datos

Los subconjuntos de datos utilizados ya vienen previamente limpios, lo que significa que no contiene valores nulos, ni datos inconsistentes. A su vez también se encuentran normalizados. El único preprocesamiento que se realizó fue el balanceo de los datos, mediante undersampling para manejar el desequilibrio en las clases minoristas, Esto fue fundamental para reducir el sesgo en el entrenamiento de los modelos y mejorar su capacidad de generalización.

### 3.4 Implementación de los modelos

El primer enfoque fue entrenar un modelo de Random Forest utilizando únicamente un conjunto de datos, específicamente mitbhtrain, el cual contiene las 5 clases, aplicando validación cruzada para evaluar su rendimiento y posteriormente evaluando con mitbhtest para corroborar su generalización. Sin embargo, se observó un rendimiento bajo en los valores de recall en algunas de las clases importantes, debido al desequilibrio entre registros.

Para mejorar el rendimiento del modelo, se implementó una estrategia de doble modelo:



**Fig. 2.** Flujo del proceso de clasificación utilizando dos modelos

- **Primer modelo:** Clasificación binaria entre electrocardiogramas normales (ptbdi\_normal.csv) y anormales (ptbdi\_abnormal.csv). Este modelo se entrenó usando Random Forest.
- **Segundo modelo:** Si un electrocardiograma es clasificado como anormal en el primer modelo, se procesa con el segundo modelo, entrenado con los datos mitbhtrain.csv y mitbhtest.csv. Este segundo modelo es capaz de distinguir entre las clases 1, 2, 3 y 4.

Esta combinación permite un rendimiento más consistente en ambos clasificadores, mitigando los problemas de sesgo y el desbalance inicial de los datos.

### 3.5 Optimización de hiperparámetros

Se realizó una búsqueda básica de hiperparámetros en el modelo Random Forest, ajustando el número de árboles y la profundidad máxima de los mismos. Dado que la búsqueda de hiperparámetros requiere un costo computacional alto, se mantuvo limitada, lo que resultó en una mejora marginal del rendimiento del modelo, aunque se reconoce que una búsqueda más exhaustiva podría haber mejorado aún más los resultados.

## 4 Resultados

En esta sección se presentan los resultados obtenidos a partir de la evaluación de los modelos de clasificación utilizados para la clasificación de electrocardiogramas. Los modelos evaluados incluyen un modelo inicial de regresión logística, seguido de siete modelos adicionales balanceados por RandomUnderSampler. Posteriormente, se desarrollaron dos modelos específicos para clasificar latidos normales y anormales, y luego para clasificar latidos anormales en cuatro categorías. A continuación, se detallan los resultados de cada modelo y su evaluación en términos de exactitud, precisión y sensibilidad.

### 4.1 Criterios de Evaluación

Antes de presentar los resultados, es importante definir los criterios de evaluación utilizados en este estudio:

- **Exactitud (Accuracy):** Mide la proporción de predicciones correctas en relación con el total de muestras. Se calcula como

$$\text{Exactitud} = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

donde  $TP$  es el número de verdaderos positivos,  $TN$  es el número de verdaderos negativos,  $FP$  es el número de falsos positivos, y  $FN$  es el número de falsos negativos.

- **Precisión (Precision):** Mide la proporción de verdaderos positivos entre todas las instancias clasificadas como positivas. Se calcula como

$$\text{Precisión} = \frac{TP}{TP + FP} \quad (2)$$

- **Sensibilidad (Recall):** Mide la capacidad del modelo para detectar todas las instancias positivas. Se calcula como

$$\text{Sensibilidad} = \frac{TP}{TP + FN} \quad (3)$$

## 4.2 Modelos Evaluados

Inicialmente, se utilizó la regresión logística sin balanceo. Posteriormente, se probaron otros siete modelos balanceados mediante RandomUnderSampler, que incluyeron Random Forest, SVM, K-Nearest Neighbors, Decision Tree, Naive Bayes, Gradient Boosting y AdaBoost. La selección de Random Forest se basó en su capacidad para manejar datos desbalanceados y en su desempeño superior en comparación con los otros modelos evaluados.

**Modelo Inicial de Clasificación** El primer modelo evaluado fue el Random Forest sin balanceo, que clasificaba entre seis clases. Los resultados se muestran en la Tabla 1.

**Table 1.** Reporte de Clasificación - Random Forest (6 Clases)

Clase	Precisión	Recall	F1-Score	Soporte
1.0	0.98	0.92	0.95	556
2.0	0.95	0.98	0.96	1448
3.0	0.84	0.81	0.82	162
4.0	0.99	0.99	0.99	1608
5.0	0.00	0.00	0.00	0
6.0	0.00	0.00	0.00	0
Exactitud	0.97			

**Explicación:** La exactitud del modelo Random Forest para clasificar entre seis clases es del 97%. Las clases 1.0 y 4.0 tienen altos valores de precisión, recall y F1-Score, indicando un buen rendimiento en estas categorías. La clase 3.0 muestra un rendimiento inferior, lo cual podría deberse a su menor representación en el conjunto de datos. Las clases 5.0 y 6.0 no tienen soporte en el conjunto de datos, lo que indica que el modelo no tuvo instancias para clasificarlas, afectando así el rendimiento global.

**Modelo de Clasificación de Latidos Normales y Anormales** El primer modelo específico creado para clasificar entre latidos normales y anormales, utilizando Random Forest con RandomUnderSampler, mostró los resultados en la Tabla 2.

**Table 2.** Reporte de Clasificación - Random Forest con Undersampling (2 Clases)

Clase	Precisión	Recall	F1-Score	Soporte
0.0	0.91	0.94	0.92	6046
1.0	0.96	0.94	0.95	10506
Exactitud	0.94			

**Explicación:** La exactitud del modelo en la clasificación binaria es del 94%. El modelo muestra una alta precisión y recall para la clase 1.0 (latidos anormales), indicando que es efectivo para identificar latidos anormales. La precisión para la clase 0.0 (latidos normales) es un poco más baja, pero sigue siendo aceptable. Estos resultados demuestran que el modelo tiene un buen equilibrio en la identificación de ambas clases.

**Prueba con Datos Nuevos del Conjunto MIT-BIH** El desempeño del modelo de clasificación binaria fue evaluado en un conjunto de datos completamente nuevo, como se muestra en la Tabla 3.

**Table 3.** Reporte de Clasificación mitbih\_test - Random Forest (2 Clases)

Clase	Precisión	Recall	F1-Score	Soporte
0.0	0.91	0.94	0.92	18118
1.0	0.66	0.52	0.59	3774
Exactitud	0.87			

**Explicación:** La exactitud del modelo en el conjunto de datos de prueba es del 87%. Aunque el recall para la clase 0.0 sigue siendo alto, la precisión y el recall para la clase 1.0 (latidos anormales) muestran una disminución significativa. Esto sugiere que el modelo tiene dificultades para generalizar a datos no vistos anteriormente, lo que puede ser debido a diferencias en la distribución de datos entre el conjunto de entrenamiento y el conjunto de prueba.

**Modelo de Clasificación de Latidos Anormales en 4 Clases** Para clasificar los latidos anormales en cuatro categorías específicas (S, Q, V, F), se utilizó el clasificador de Random Forest. Los resultados se muestran en la Tabla 4.

**Table 4.** Reporte de Clasificación - Random Forest (4 Clases)

Clase	Precisión	Recall	F1-Score	Soporte
1.0	0.97	0.94	0.95	2223
2.0	0.96	0.98	0.97	5788
3.0	0.92	0.82	0.87	641
4.0	0.99	0.99	0.99	6431
Exactitud	0.97			

**Explicación:** El modelo de Random Forest para clasificar los latidos anormales en cuatro categorías muestra una exactitud del 97%. Las clases 1.0 (Supraventricular Arrhythmias), 2.0 (Unknown or Unclassified), y 4.0 (Fusion Beats) tienen altas puntuaciones de precisión y recall, indicando un buen rendimiento en la clasificación. La clase 3.0 (Ventricular Arrhythmias) tiene una precisión y recall



relativamente más bajos, lo que podría deberse a su menor representación en el conjunto de datos.

**Prueba con Datos Nuevos del Conjunto MIT-BIH para 4 Clases** El modelo para clasificar los latidos anormales en cuatro clases fue probado en un conjunto de datos nuevo, con los resultados presentados en la Tabla 5.

**Table 5.** Reporte de Clasificación mitbih\_test - Random Forest (4 Clases)

Clase	Precisión	Recall	F1-Score	Soporte
1.0	0.91	0.81	0.86	5802
2.0	0.78	0.88	0.83	3500
3.0	0.61	0.40	0.48	612
4.0	0.94	0.98	0.96	3294
Exactitud	0.87			

**Explicación:** La exactitud del modelo en el conjunto de datos de prueba para las cuatro clases es del 87%. La clase 4.0 (Fusion Beats) muestra una alta precisión y recall, pero las clases 1.0, 2.0 y 3.0 tienen un rendimiento inferior en comparación con el conjunto de entrenamiento. Estos resultados reflejan la dificultad del modelo para generalizar a nuevas instancias, especialmente en las clases con menor representación en los datos de prueba.

**Optimización de Hiperparámetros con RandomizedSearchCV** La optimización de hiperparámetros es esencial para maximizar el rendimiento de los modelos de clasificación. Utilizamos `RandomizedSearchCV`, que evalúa una muestra aleatoria de combinaciones de parámetros, reduciendo el tiempo de cómputo en comparación con `GridSearchCV`.

**Table 6.** Reporte de Clasificación - Random Forest con Mejor Parámetro (Randomized Search)

Clase	Precisión	Recall	F1-Score	Soporte
0.0	0.97	0.94	0.95	10506
1.0	0.94	0.97	0.95	10506
Exactitud	0.95			
Macro promedio	0.95	0.95	0.95	21012
Promedio ponderado	0.95	0.95	0.95	21012

**Explicación:** El modelo Random Forest optimizado obtuvo una exactitud del 95%. La precisión y el recall para ambas clases son altas, lo que indica un equilibrio en la capacidad del modelo para identificar correctamente tanto los latidos normales como los anormales. El *macro promedio* y el *promedio ponderado*

muestran valores consistentes de 0.95, reflejando un excelente rendimiento en general. Este ajuste de hiperparámetros ayudó a mejorar la capacidad del modelo para generalizar en nuevos conjuntos de datos.

## 5 Conclusion

Los resultados muestran que el modelo **Random Forest** es el más efectivo en la clasificación de electrocardiogramas entre múltiples clases. Esto se debe en gran parte a la capacidad del modelo para trabajar con grandes cantidades de datos y su habilidad para mitigar el sobreajuste y el desbalanceo de datos. No obstante, los modelos de clasificación binaria y de cuatro clases también muestran buenos resultados, su rendimiento en conjuntos de datos nuevos sugiere la necesidad de mejoras adicionales. La optimización de hiperparámetros ha sido crucial para mejorar el rendimiento de los modelos y debe considerarse como parte integral del proceso de modelado, sin embargo esta optimización puede conllevar un alto costo computacional, por lo que se debe lograr un balance, entre buena precisión del modelo y el tiempo de procesamiento de hiperparametros.

Finalmente, este trabajo expone la importancia de del desarrollo de modelos predictivos y su gran impacto en las distintas áreas, En el contexto de la salud, donde puede significar la diferencia entre un diagnóstico temprano y una complicación severa, es fundamental alcanzar un modelo que sea no solo preciso, sino también confiable.

## 6 Conclusiones personales

**Jared Andres Silva Villa:**

A lo largo de esta investigación, me he percatado de lo relevante que es el avance tecnológico y su impacto en las distintas áreas, particularmente en este caso en el área de la salud, la capacidad de poder o lograr predecir los resultados a través de modelos predictivos me a resultado bastante interesante, ya que esto puede llevar a soluciones cruciales. Es muy sonado como un error de algún doctor en a la hora de leer un electrocardiograma puede tener consecuencia graves en la vida de una persona, por lo que el desarrollo de estos modelos automáticos que serán los encargados de leer con una alta precisión, es una avance significativo para la mejora de diagnósticos y la prevención de tragedias.

Personalmente, ha sido una experiencia muy satisfactoria el llevar a cabo todas las pruebas y ajustes necesarios para llegar a los resultados obtenidos. Disfrute el proceso de probar múltiples modelos, es interesante el saber que un modelo no siempre será el mejor para todos los distintos datos y que en cada escenario existe un modelo más ideal que otro, es por eso que es importante la interpretación correcta de los resultados en cada caso y lograr adaptar las soluciones a cada contexto diferente. Este trabajo ha destacado la importancia de los modelos predictivos en áreas de la salud, y a nivel personal, ha sido una oportunidad valiosa para profundizar en manejo de datos y modelos predictivos.

**Paola Félix Torres:** yupy yupy  
**Alma Paulina González Sandoval:**

En la elaboración del proyecto, he podido observar la importancia de la tecnología en el ámbito de la salud, específicamente en la predicción y diagnóstico de enfermedades. La capacidad de usar modelos de machine learning para analizar grandes volúmenes de datos y ofrecer diagnósticos precisos es una herramienta poderosa que puede ayudar a evitar errores humanos. Me pareció interesante cómo cada modelo tiene sus fortalezas y debilidades dependiendo del tipo de datos, y cómo resulta crucial ajustar los modelos para lograr el mejor desempeño posible. Esta experiencia me ha permitido no solo adquirir conocimientos técnicos, sino también apreciar el impacto que estas herramientas pueden tener en la vida de las personas, aportando una mejora en los diagnósticos médicos y a la prevención de enfermedades graves.

**Herbert:** yupy yupy yeiiyeii

## References

1. Saira Aziz, Sajid Ahmed, and Mohamed-Slim Alouini. Ecg-based machine-learning algorithms for heartbeat classification. *Scientific reports*, 11(1):18738, 2021.
2. Daniel Berrar et al. Cross-validation., 2019.
3. Ernest Yeboah Boateng, Joseph Otoo, and Daniel A Abaye. Basic tenets of classification algorithms k-nearest-neighbor, support vector machine, random forest and neural network: A review. *Journal of Data Analysis and Information Processing*, 8(4):341–357, 2020.
4. Julian Hatwell, Mohamed Medhat Gaber, and R Muhammad Atif Azad. Chirps: Explaining random forest classification. *Artificial Intelligence Review*, 53:5747–5788, 2020.
5. Kaitlin Kirasich, Trace Smith, and Bivin Sadler. Random forest vs logistic regression: binary classification for heterogeneous datasets. *SMU Data Science Review*, 1(3):9, 2018.
6. Carlos Enrique Ochoa-Castro, Héctor Cobos-Aguilar, Patricia Pérez-Cortez, María de los Ángeles Marroquín-Cardona, and Michel Marcela Gómez-Sánchez. Aptitud clínica en la interpretación de electrocardiograma en una muestra de médicos becarios. *Investigación en educación médica*, 3(9):9–15, 2014.