



Universidad **Mariana**

Desarrollo e implementación de un algoritmo de aprendizaje automático para la reconstrucción de imágenes electrocardiográficas (ECGI)

Oscar Daza Renza
Santiago Holguín Torres

Universidad Mariana
Facultad Ingeniería
Programa Mecatrónica
San Juan de Pasto

2024

Desarrollo e implementación de un algoritmo de aprendizaje automático para la reconstrucción
de imágenes electrocardiográficas (ECGI)

Oscar Daza Renza
Santiago Holguín Torres

Mg. Dagoberto Mayorca Torres
Asesor

Trabajo de grado para optar al título de Ingeniero Mecatrónico

Universidad Mariana
Facultad Ingeniería
Programa Mecatrónica
San Juan de Pasto
2024

Artículo 71: los conceptos, afirmaciones y opiniones emitidos en el Trabajo de Grado son responsabilidad única y exclusiva del (los) Educando (s)

Reglamento de Investigaciones y Publicaciones, 2007
Universidad Mariana

Contenido

Introducción	9
1. Resumen del proyecto	10
1.1. Descripción del problema	10
1.1.1. Formulación del problema	11
1.2. Justificación.....	11
1.3. Objetivos	13
1.3.1. Objetivo general.....	13
1.3.2. Objetivos específicos	13
1.4. Marco referencial o fundamentos teóricos.....	14
1.4.1. Antecedentes	14
1.4.1. Marco teórico.....	15
1.4.2. Machine learning.....	15
1.4.2.1. Aprendizaje supervisado	16
1.4.2.2. Aprendizaje sin supervisión	16
1.4.2.3. Aprendizaje por refuerzo.....	16
1.4.3. Métodos de clasificación de machine learning	17
1.4.3.1. Análisis de componentes principales.....	17
1.4.3.2. Análisis de componentes independientes	17
1.4.4. Regresión por vectores de soporte	18
1.4.5. KNN Regressor.....	19
1.4.6. Decision Tree Regressor	19
1.4.7. AdaBoost Regressor	19
1.4.8. Gradient Boosting Regressor	19
1.4.9. Bagging Regressor.....	20
1.4.10. Extra Tree Regressor.....	20

1.4.11. Hiperparámetros	21
1.4.12. Electrocardiograma (ECG).....	21
1.4.12.1. Imágenes electrocardiográficas (ECGI)	22
1.4.12.2. Ondas electrocardiográficas	22
1.4.13. Métricas por validación de métodos ECGI	23
1.4.13.1. Correlación de Pearson.....	23
1.4.13.2. Error relativo o absoluto.....	24
1.4.13.3. Error cuadrático medio.....	24
1.5. Metodología	24
1.5.1. Tipo de investigación.....	24
1.5.2. Línea de investigación	25
1.5.3. Hipótesis de investigación	25
1.5.4. Matriz metodológica	25
1.5.4.1. Fase 1 – Análisis de Requerimientos y Selección de Base de Datos	25
1.5.4.1.1. Revisión Bibliográfica y Análisis del Estado del Arte.....	25
1.5.4.1.2. Identificación de Requerimientos Funcionales	25
1.5.4.1.3. Evaluación y Selección de Base de Datos.....	26
1.5.4.2. Fase 2 – Desarrollo de Algoritmo para Modelo de Función Eléctrica Cardíaca.....	26
1.5.4.2.1. Diseño Preliminar del Algoritmo de Aprendizaje Automático	26
1.5.4.2.2. Desarrollo de modelos y pruebas iniciales	26
1.5.4.3. Fase 3 – Validación y evaluación de Requerimientos del Algoritmo	26
1.5.4.3.1. Desarrollo de pruebas y Establecimiento de Métricas de Calidad	26
1.5.4.3.2. Validación y Ajustes Finales.....	27
2. Presentación y discusión de resultados	27
2.1. Análisis e interpretación de resultados.....	27
2.1.1. Análisis del primer objetivo.....	27

2.1.1.1. Revisión bibliográfica y análisis del estado del arte	28
2.1.1.2. Identificación de requerimientos funcionales	30
2.1.1.3. Selección del conjunto de datos	31
2.1.2. Desarrollo del segundo objetivo	31
2.1.2.1. Diseño preliminar del algoritmo de aprendizaje automático	32
2.1.2.2. Desarrollo de modelo y pruebas iniciales	34
2.1.3. Análisis del tercer objetivo	45
2.1.3.1. Desarrollo de pruebas y Establecimiento de Métricas de Calidad	45
2.1.3.2. Validación y ajustes finales	46
3. Conclusiones	49
4. Recomendaciones	50
Referencias bibliográficas	51

Índice de Tablas

Tabla 1. Matriz de artículos relacionados.....	28
Tabla 2. Recopilación de datos obtenidos	41
Tabla 3. Hiperparámetros usados con KNN Regressor	43

Índice de Figuras

Figura 1. Ondas ECG. Tomado de Merck Sharp & Dohme Corp. (s.f.)	22
Figura 2 . Etapas del modelo	33
Figura 3. Resultados obtenidos KNN Regressor	35
Figura 4. Resultados obtenidos Decision Tree Regressor.....	36
Figura 5. Resultados obtenidos Bagging Regressor	37
Figura 6. Resultados obtenidos AdaBoost Regressor.....	38
Figura 7. Resultados obtenidos Extra Tree Regressor.....	39
Figura 8. Resultados obtenidos Gradient Boosting Regressor	40
Figura 9. Resultados MSE.....	41
Figura 10. Resultados R2	42
Figura 11. Métricas de validación	44
Figura 12. Reconstrucción de señal ECGI con KNN Regressor y conjunto de datos Kitpvc ...	45
Figura 13. Reconstrucción de señal ECGI con KNN Regressor y conjunto de datos Utah	47
Figura 14. Comparación de resultados entre los estudios	48

Introducción

El análisis de la actividad eléctrica cardíaca es esencial para diagnosticar enfermedades relacionadas con el funcionamiento electrofisiológico anómalo del corazón, tanto en la investigación como en la práctica clínica. Sin embargo, la electrocardiografía se enfrenta a desafíos cada vez más complejos en la era contemporánea, donde la demanda de diagnósticos precisos y la localización de arritmias cardíacas constituyen retos cruciales. (Ramanathan, Jia, Ghanem, Calvetti, & Rudy, 2003).

Para abordar estos desafíos, la comunidad médica y científica está adoptando una combinación de técnicas tradicionales de detección, como el Electrocardiograma (ECG), con herramientas de análisis asistido por computadora y métodos avanzados de procesamiento de señales. Esta convergencia permite una evaluación más precisa y automatizada de la función cardíaca, abriendo nuevas posibilidades para mejorar tanto el diagnóstico como el tratamiento de enfermedades cardiovasculares.

Sin embargo, el problema inverso de la electrocardiografía presenta desafíos particulares. A pesar de sus aplicaciones clínicas, este problema está mal planteado, con múltiples soluciones posibles y una inestabilidad inherente. Por lo tanto, una de las principales líneas de investigación en este campo es el desarrollo de un algoritmo para el tratamiento de las señales registradas en la superficie del torso, que permitan resolver eficazmente este problema.

1. Resumen del proyecto

El presente estudio se enfoca en la implementación de algoritmos de aprendizaje automático de regresión para el procesamiento de señales temporales de ECG, con el fin de optimizar su rendimiento mediante la selección y ajuste de hiperparámetros. Este enfoque permite diseñar un algoritmo que pueda abordar el problema inverso de la electrocardiografía de manera eficaz, proporcionando resultados precisos y confiables.

El diseño del algoritmo comprende varias etapas, desde la selección de la base de datos hasta la validación de la funcionalidad y el rendimiento del sistema de reconstrucción de imágenes electrocardiográficas. Se hacen pruebas iniciales con distintos métodos de regresión, evaluando métricas de calidad de imagen para comparar su rendimiento y seleccionar el modelo óptimo.

La implementación de hiperparámetros se realiza para mejorar aún más el rendimiento del modelo seleccionado, utilizando técnicas como GridSearch para encontrar la mejor combinación de parámetros. Finalmente, se valida el modelo con conjuntos de datos adicionales para garantizar su eficacia en la reconstrucción de señales ECGI.

1.1. Descripción del problema

El estudio de la actividad eléctrica cardíaca cobra una gran importancia a la hora de diagnosticar enfermedades relacionadas con el funcionamiento electrofisiológico anómalo del corazón, utilizándose tanto en el ámbito de la investigación como en el clínico. Sin embargo, en la era contemporánea, se enfrenta a una serie de desafíos cada vez más complejos en el ámbito clínico. La demanda de diagnósticos precisos y la localización de arritmias cardíacas representan dos de los principales retos que la electrocardiografía debe abordar en la práctica médica actual.

Para hacer frente a estos desafíos, la comunidad médica y científica está recurriendo a una combinación de técnicas tradicionales de detección como Electrocardiograma, con herramientas de análisis asistido por computadora y métodos avanzados de procesamiento de señales. Esta

convergencia de enfoques permite una evaluación más precisa y automatizada de la función cardíaca, abriendo nuevas posibilidades para mejorar tanto el diagnóstico como el tratamiento de enfermedades cardiovasculares.

Según Pedrón (2010) el problema directo de la electrocardiografía posee una única solución y es estable, resolviéndose fácilmente. No obstante, el problema inverso de la electrocardiografía, a pesar de tener aplicaciones clínicas, está mal planteado, posee múltiples soluciones posibles y es inestable, siendo necesaria la aplicación de técnicas de regularización y estabilización. Por ello, en la actualidad, una de las grandes líneas de investigación en el campo de la electrocardiografía es el desarrollo de algoritmos para el tratamiento de las señales registradas en la superficie del torso que permita resolver el problema inverso de la electrocardiografía, evitando dichos inconvenientes tanto como sea posible.

Al adoptar un enfoque basado en la inteligencia artificial, este trabajo se posiciona en la frontera del conocimiento, buscando ofrecer soluciones avanzadas para el modelado de la función eléctrica del corazón. A través de esta investigación, se aspira a contribuir significativamente al avance de la cardiología, proporcionando herramientas más precisas y efectivas para el diagnóstico y tratamiento de enfermedades cardiovasculares.

1.1.1. Formulación del problema

¿Cómo puede un algoritmo de machine learning basada en modelos de regresión de múltiples entradas y salidas de datos temporales de ECGI resolver eficazmente el problema inverso de la electrocardiografía?

1.2. Justificación

La electrocardiografía (ECG) constituye un pilar fundamental en el ámbito de la cardiología, permitiendo una evaluación minuciosa de la actividad eléctrica del corazón. Sin embargo, en la actualidad, se encuentran desafíos considerables a enfrentar en el intento de abordar problemas clínicos cada vez más complejos, como el diagnóstico preciso y la localización de arritmias

cardíacas. Afortunadamente, la convergencia de los métodos de detección convencionales con herramientas de análisis asistido por computadora y técnicas avanzadas de procesamiento de señales como Redes Neuronales Convolucionales o Recurrentes que ofrecen perspectivas alentadoras para mejorar la precisión y la automatización del diagnóstico cardiovascular.

En esta situación, se presenta la necesidad de desarrollar e implementar una arquitectura de aprendizaje automático capaz de abordar el problema inverso de la electrocardiografía: la reconstrucción de imágenes electrocardiográficas (ECGi) a partir de datos temporales de ECG. Este enfoque busca aprovechar las capacidades del machine learning para modelar de manera efectiva la función eléctrica del corazón, de modo que permita contribuir al avance del conocimiento existente al redor de este campo, ampliando la información ya existente y que pueda ser usada para el desarrollo de futuras investigaciones.

A medida que la medicina avanza, la demanda de soluciones más avanzadas y precisas para afrontar desafíos diagnósticos y terapéuticos crece. De modo que el desarrollo de una arquitectura de aprendizaje automático específicamente diseñada para la reconstrucción de imágenes electrocardiográficas emerge como un avance, a raíz de que esta arquitectura trabajará en mejorar la precisión y eficacia en comparación con métodos que ya han sido usados a la hora de abarcar esta problemática. Además de sus implicaciones clínicas, este proyecto representa una contribución valiosa a la investigación científica en el campo de la medicina cardiovascular, al fusionar técnicas de aprendizaje automático con la interpretación de señales de ECG, se abren nuevas perspectivas y avances en el diagnóstico y tratamiento de enfermedades cardíacas.

La aplicación exitosa de este algoritmo de aprendizaje automático podría ofrecer herramientas avanzadas y exactas para diagnosticar y vigilar enfermedades cardíacas. Esto no solo mejoraría la calidad de la atención al paciente, sino que también posibilitaría diagnósticos más oportunos y precisos, así como tratamientos más efectivos y adaptados a cada caso, lo cual tendría un impacto considerable en el ámbito clínico relacionado con la cardiología.

El avance científico en este campo se liga directamente al transcurso de los años, de modo que distintos métodos han sido usados con el fin de proponer distintas soluciones a este característico

problema de ECG inverso, por lo que es posible encontrar distintos estudios que han brindado valiosa información a la hora de resolver dicha problemática, como se encuentra en el estudio de Lui (2012) o Jiang (2013). Las ciencias computacionales ofrecen una amplia gama de métodos que pueden aplicarse en un mismo campo, es por esto que a través de este proyecto se busca implementar y recurrir a métodos diferentes a los tradicionales que suelen usarse para el desarrollo de este estudio, a raíz de que algunos cuentan con porcentajes bajos de exactitud.

1.3. Objetivos

1.3.1. Objetivo general

Desarrollar un algoritmo basado en técnicas de aprendizaje automático que facilite el modelado de la función eléctrica del corazón con la capacidad para procesar señales temporales orientadas a la reconstrucción de imágenes electrocardiográficas.

1.3.2. Objetivos específicos

- Seleccionar la base de datos de imágenes electrocardiográficas que mejor se ajuste a las necesidades del proyecto en términos de tamaño, calidad, diversidad y características de las señales, considerando los requerimientos de la arquitectura y el objetivo de la investigación.
- Implementar algoritmos de aprendizaje automático de regresión con la capacidad para procesar señales temporales de ECG, optimizando su rendimiento mediante la selección y ajuste de hiperparámetros.
- Validar la funcionalidad y el rendimiento del sistema de reconstrucción de imágenes electrocardiográficas mediante la evaluación de métricas de calidad de imagen, comparando su rendimiento con otros métodos existentes para mejorar la precisión de la reconstrucción, utilizando bases de datos validadas.

1.4. Marco referencial o fundamentos teóricos

1.4.1. Antecedentes

En la investigación llevada a cabo por Jiang (2011), menciona a la electrocardiografía inversa como un método no invasivo para visualizar la actividad eléctrica del corazón a partir de los potenciales de superficie corporal (BSPs). Sin embargo, destaca que este método se enfrenta a un problema mal planteado, (MacLeod and Brooks 1998, Cheng et al 2003, Ramanathan et al 2004, citado en Jiang, 2011). Las soluciones convencionales abordan este problema mediante modelos basados en activación o potenciales. La relación entre los potenciales transmembrana (TMPs) en la superficie cardíaca y los BSPs se describe mediante una ecuación integral de Fredholm. Debido a la mala condición de la matriz de transferencia, el problema de la ECG inversa se torna severamente mal condicionado, requiriendo técnicas de regularización. Para superar las limitaciones de los métodos tradicionales, en este estudio se propone un enfoque más robusto que se centra en los TMPs sobre la superficie cardíaca utilizando la regresión vectorial de soporte (SVR). Se trata el problema inverso de la ECG como una tarea de regresión con múltiples entradas (BSPs) y salidas (TMPs). Para abordar esto, el autor sugiere el uso de algoritmos, como lo pueden ser algoritmos genéticos y métodos de optimización simplex para ajustar los parámetros SVR de manera óptima. Para desarrollar un modelo SVR eficaz, recurren a la extracción de características mediante análisis de componentes principales (PCA) y análisis de componentes principales de kernel (KPCA). Los desempeños de los métodos PCA-SVR y KPCA-SVR se comparan con el método SVR único en la reconstrucción de los TMPs cardíacos a partir de BSPs.

En el estudio Jiang (2013), sobre el problema del ECG inverso, se aborda la reconstrucción no invasiva de potenciales transmembrana (TMP) a partir de potenciales de superficie corporal (BSP). Se trata como un problema de regresión con múltiples entradas (BSP) y salidas (TMP), resolviéndose mediante el método de regresión del vector de soporte (SVR). La optimización de los hiperparámetros de SVR es crucial para lograr precisión y rendimiento óptimos. Se proponen tres métodos de optimización: algoritmo genético (GA), evolución diferencial (DE) y optimización de enjambre de partículas (PSO). El artículo investiga la efectividad de reconstruir TMP cardíacos a partir de BSP, comparando el desempeño de los tres métodos. Los resultados experimentales

indican que los tres métodos son eficaces, pero el algoritmo PSO destaca por su eficiencia en la optimización de parámetros, mostrando un mejor rendimiento en la reconstrucción precisa de TMP en comparación con DE y GA.

Este artículo Liu (2012), se aborda la técnica de imágenes no invasivas de la actividad eléctrica del corazón a partir de los potenciales en la superficie corporal (BSPs), un aspecto del problema inverso del ECG. Las soluciones para el problema inverso del ECG generalmente se basan en modelos de activación o modelos basados en potenciales, que incluyen potenciales epicárdicos, endocárdicos o transmembranales. Los modelos de activación investigan el tiempo de llegada de la onda de propagación dentro del miocardio, mientras que los modelos basados en potenciales evalúan los valores de potencial en la superficie cardíaca o dentro del miocardio en momentos específicos. Este estudio también destaca la eficacia de los modelos híbridos que combinan diferentes métodos, como el uso de Self-Organizing Map (SOM) junto con SVR o LS-SVM. Se propone un modelo híbrido MCC-SVR que integra el método de Maximum Margin Clustering (MMC) modificado con SVR para abordar el problema inverso del ECG de manera más eficiente. La eficiencia del método MMC se mejora mediante un procedimiento de optimización alternativo implementado con el método SVR con pérdida Laplaciana en el subproblema de optimización interno.

1.4.1. Marco teórico

1.4.2. Machine learning

El aprendizaje automático o machine learning es una forma de inteligencia artificial que permite que los sistemas aprendan de los datos en lugar de mediante programación explícita. Sin embargo, el aprendizaje automático no es un proceso simple. Dado que el algoritmo toma datos de entrenamiento, se pueden construir modelos más precisos basados en estos datos. Un modelo de aprendizaje automático es el resultado generado cuando un algoritmo de aprendizaje automático se entrena con datos. Después del entrenamiento, darle al modelo una entrada le dará una salida. Por ejemplo, un algoritmo de predicción generará un modelo predictivo. Luego, cuando alimenta un modelo predictivo con los datos, obtiene una predicción basada en los datos en los que se entrenó el modelo.

1.4.2.1. Aprendizaje supervisado. En el aprendizaje supervisado, la máquina aprende con el ejemplo. De esta forma, el operador proporciona al algoritmo un conjunto conocido de datos que contienen las entradas y salidas deseadas, y el algoritmo necesita encontrar una forma de determinar cómo llegar a esas entradas y salidas. Mientras que el operador sabe la respuesta exacta al problema, el algoritmo detectará patrones en los datos, aprenderá de las observaciones y hará predicciones. El algoritmo hace predicciones y el operador lo ajusta, y este proceso continúa hasta que el algoritmo alcanza un alto nivel de precisión y rendimiento.

1.4.2.2. Aprendizaje sin supervisión. En el aprendizaje supervisado, la máquina aprende con el ejemplo. De esta forma, el operador alimenta al algoritmo con un conjunto de datos conocido que contiene las entradas y salidas deseadas, y el algoritmo debe encontrar la manera de determinar cómo tomar estas entradas y salidas. Siempre que el operador sepa la respuesta correcta al problema, el algoritmo detectará patrones en los datos, aprenderá de las observaciones y hará predicciones.

1.4.2.3. Aprendizaje por refuerzo. El aprendizaje por refuerzo se centra en los procesos de aprendizaje guiado, donde los algoritmos de aprendizaje automático están dotados de un conjunto de operaciones, parámetros y puntos finales.

Al definir reglas, el algoritmo intenta explorar diferentes opciones y posibilidades, monitoreando y evaluando cada resultado para determinar cuál es óptimo. Por lo tanto, este sistema

entrena la máquina por ensayo y error. Aprenda de su experiencia pasada y comience a adaptar su enfoque a la situación para lograr el mejor resultado posible.

1.4.3. Métodos de clasificación de machine learning

El machine learning cuenta con una amplia variedad de métodos a través de los cuales es posible obtener un aprendizaje automático por parte de un sistema, a continuación, serán mencionados algunos. Análisis de componentes principales: El análisis de componentes principales (PCA, por sus siglas en inglés Principal Component Analysis) es una herramienta de reducción de datos que se puede utilizar para transformar un conjunto bastante grande de variables en un conjunto más pequeño que contiene tanta información como un conjunto grande.

En este sentido, PCA es un procedimiento matemático que convierte un conjunto de variables correlacionadas en un conjunto (más pequeño) de variables no correlacionadas llamadas componentes principales. El primer componente principal representa el máximo número posible de variaciones en los datos, y cada componente sucesivo explica la mayor parte de la variación restante, es decir, el conjunto de datos con la menor cantidad de dimensiones debe mantener el error de los datos originales. En este sentido, PCA es un procedimiento matemático que convierte un conjunto de variables correlacionadas en un conjunto (más pequeño) de variables no correlacionadas llamadas componentes principales.

1.4.3.1. Análisis de componentes independientes. El análisis de componentes independientes (ICA, por sus siglas en inglés Independent Component Analysis) es una técnica estadística utilizada para descubrir factores ocultos en un conjunto de datos de variables aleatorias, señales o mediciones. Como alternativa al análisis de componentes principales (PCA), ICA ayuda a separar señales multivariantes en subcomponentes que se consideran de naturaleza no gaussiana e independientes entre sí.

Por ejemplo, en el caso de la separación de fuentes de audio, ICA transforma los diferentes vectores en conjuntos de datos independientes que ayudan a identificar las fuentes de audio. El concepto de ICA, inventado en 1986, fue introducido por Herault y Yutten para resolver el

problema de la separación ciega de fuentes (BSS) en el procesamiento de señales. Aunque el concepto es similar al sistema PCA, es una forma activa de detectar subcomponentes independientes en un conjunto de datos y es mucho más preciso. Al mismo tiempo, ICA es conocida por su eficacia en los campos del aprendizaje automático y las estadísticas.

1.4.4. Regresión por vectores de soporte

La Regresión de Vectores de Soporte (SVR, por sus siglas en inglés Support Vector Regression) es una extensión de las Máquinas de Vectores de Soporte (SVM) en el contexto de la regresión. Mientras que las SVM se utilizan para problemas de clasificación, SVR aborda la tarea de predecir valores numéricos. La base de SVR reside en la misma idea fundamental que las SVM: la identificación de vectores de soporte, que son puntos de datos cruciales que definen la frontera de decisión del modelo. SVR se enfrenta a la dificultad de predecir valores continuos. En problemas de clasificación, las opciones son limitadas y discretas, mientras que, en la regresión, hay infinitas posibilidades. Por lo tanto, SVR busca ajustarse a la variabilidad inherente en la predicción de valores reales. El "kernel trick" es una característica esencial de SVR. Este truco permite mapear los datos originales a un espacio dimensional superior, lo que facilita el manejo de relaciones no lineales entre las variables. Al hacerlo, SVR puede capturar patrones complejos y no lineales en los datos de entrada, mejorando su capacidad para realizar predicciones precisas.

En términos prácticos, SVR minimiza una función de pérdida que considera tanto el error de ajuste entre las predicciones y los valores reales como una penalización por la desviación de los vectores de soporte. La elección del kernel, como lineal, polinómico o de función de base radial (RBF), dependerá de la naturaleza de los datos y la complejidad de la relación a modelar.

En la regresión lineal se busca minimizar una función de error, generalmente el error cuadrático, en todos los puntos del conjunto de entrenamiento, en la SVR se define una zona en torno al hiperplano donde se ignora los errores. Lo que significa que en SVR se busca aproximar el mejor valor dentro de un margen dado.

1.4.5. KNN Regressor

El método KNN (k-Nearest Neighbors) es un algoritmo de aprendizaje supervisado que se utiliza tanto para problemas de clasificación como de regresión. Es considerado un algoritmo de "aprendizaje basado en instancias" porque no construye un modelo explícito durante la fase de entrenamiento, sino que memoriza las instancias de datos de entrenamiento.

1.4.6. Decision Tree Regressor

Los árboles de decisión son un tipo de algoritmo de aprendizaje automático que clasifica o hace regresión de datos usando preguntas con respuestas verdaderas o falsas. Imagina un árbol donde cada nodo representa una pregunta y las ramas son posibles respuestas. Al final, las hojas del árbol son las categorías o valores finales. Para crear el árbol, se comienza con una pregunta que divide los datos en grupos más puros (menos mezclados). Se usan métodos como Gini o entropía para medir cuán mezclados están los datos en cada grupo. Se elige la pregunta que reduce la mezcla (impureza) al máximo. La impureza de Gini, por ejemplo, se calcula ordenando los datos y viendo cómo las divisiones clasifican correctamente los datos. Este proceso se repite para cada nivel del árbol, creando una estructura fácil de entender y explicar.

1.4.7. AdaBoost Regressor

El método AdaBoost Regressor es un algoritmo de ensemble learning que combina múltiples modelos de regresión débiles para mejorar la precisión. Cada modelo débil se entrena para corregir los errores del anterior, asignando pesos a las instancias según su rendimiento. Durante la predicción, se combinan las predicciones de todas las modelos ponderadas por su rendimiento, lo que permite crear un modelo robusto y de alta precisión, especialmente útil en problemas complejos o ruidosos.

1.4.8. Gradient Boosting Regressor

Gradient Boosting es una técnica de Machine Learning utilizada para hacer predicciones en problemas de regresión y clasificación. Consiste en crear un modelo predictivo usando un conjunto de modelos de predicción débiles, generalmente árboles de decisión individuales.

En este método, los árboles de decisión se entrenan uno tras otro de manera secuencial, y cada nuevo árbol intenta corregir los errores de los árboles anteriores. Para predecir un nuevo dato, se suman las predicciones de todos los árboles del modelo. Esta técnica es poderosa porque los árboles de decisión pueden dividir el espacio de los datos en regiones simples, facilitando la comprensión de las interacciones entre variables.

LightGBM es un framework específico de Gradient Boosting que se enfoca en la eficiencia. En este método, cada nuevo árbol se construye para predecir el error en la suma de las predicciones de los árboles anteriores. Esto significa que se presta más atención a las muestras en las que el modelo está fallando, mejorando continuamente la precisión del modelo. Al final, las predicciones se calculan sumando los resultados de todos los árboles construidos, lo que ayuda a mejorar el rendimiento general del modelo.

1.4.9. Bagging Regressor

El método Bagging Regressor es un algoritmo de ensemble learning que mejora la precisión y robustez al promediar múltiples modelos de regresión entrenados en conjuntos de datos bootstrap. Cada modelo base se entrena de forma independiente en un conjunto de datos generado mediante muestreo con reemplazo. Durante la predicción, se promedian las predicciones de todos los modelos, lo que resulta en un modelo final estable y preciso, útil para problemas de regresión con datos ruidosos o con valores atípicos.

1.4.10. Extra Tree Regressor

El método Extra Trees Regressor es un algoritmo de ensemble learning que construye múltiples árboles de decisión de forma aleatoria. A diferencia de Random Forest, selecciona características y umbrales al azar para cada árbol, reduciendo la varianza del modelo. Durante la predicción, la

combinación de las predicciones de los árboles proporciona una estimación robusta y precisa, siendo útil para problemas de regresión donde se busca un modelo resistente al sobreajuste y con bajo sesgo.

1.4.11. Hiperparámetros

Los hiperparámetros son esenciales en el mundo del machine learning, siendo considerados como las variables de ajuste externas que los científicos de datos manipulan para dirigir el proceso de entrenamiento de los modelos. A menudo denominados como los "ajustes finos" del modelo, estos parámetros se establecen manualmente antes de iniciar el proceso de entrenamiento. A diferencia de los parámetros, que son internos y se ajustan automáticamente durante el proceso de aprendizaje, los hiperparámetros son configurados deliberadamente por los científicos de datos.

Estos hiperparámetros pueden variar ampliamente, desde el número de nodos y capas en una red neuronal hasta la profundidad de un árbol de decisiones. Son críticos para definir aspectos clave del modelo, tales como su arquitectura, velocidad de aprendizaje y nivel de complejidad.

1.4.12. Electrocardiograma (ECG)

Desde su introducción a principios del siglo XX por Einthoven, el electrocardiograma no ha sufrido cambios sorprendentes, el electrocardiograma permite registrar simultáneamente la activación del miocardio en múltiples puntos de la superficie del cuerpo y la activación eléctrica en diferentes regiones del cuerpo. y permite el análisis de la activación eléctrica de diferentes regiones del miocardio. Los electrocardiogramas de superficie se pueden complementar con registros intracardiacos, lo que es particularmente útil en el diagnóstico y tratamiento de arritmias. (Goldman, 2021)

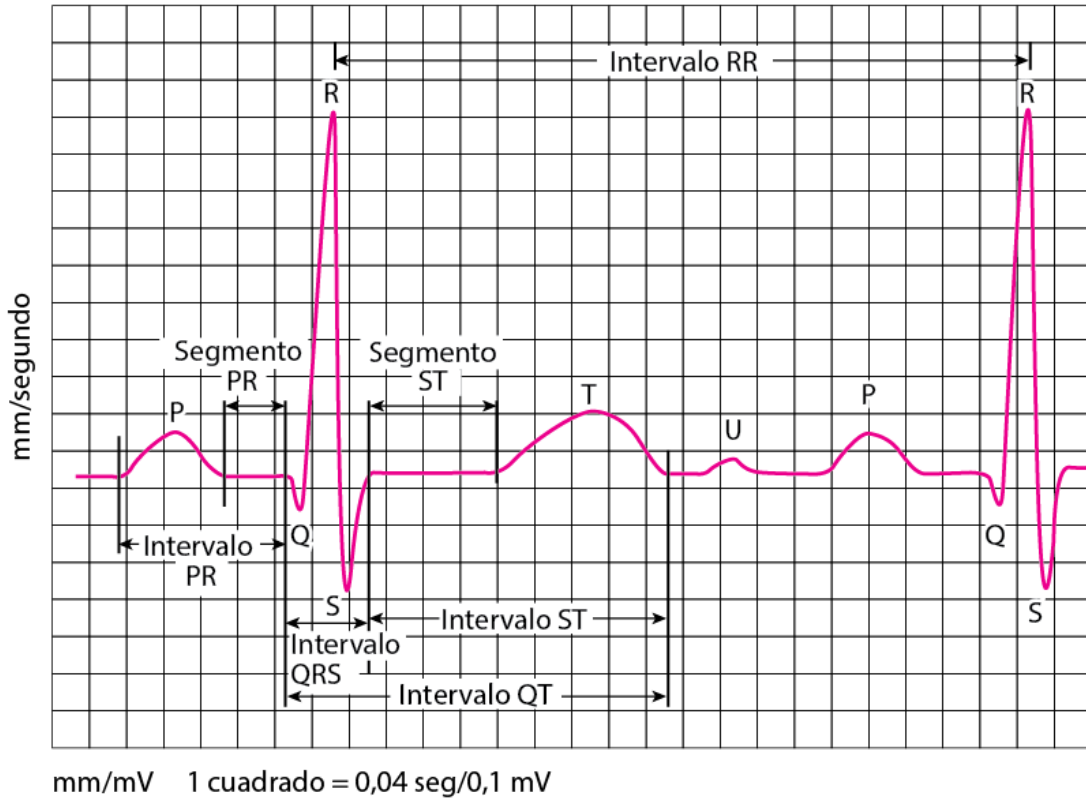
1.4.12.1. Imágenes electrocardiográficas (ECGI). La Imagen Electrocardiográfica (ECGI) representa una innovadora técnica de imagen no invasiva utilizada en la electrofisiología cardíaca (EP). Surge como respuesta a la necesidad de obtener una visión más completa y detallada de las arritmias cardíacas, más allá de lo que proporcionan los métodos convencionales de 12 derivaciones. Estos métodos no ofrecen información precisa sobre la ubicación de la actividad eléctrica en el corazón ni sobre la secuencia de activación durante las arritmias. En cambio, el mapeo detallado del potencial en la superficie corporal puede brindar una mejor comprensión eléctrica y diagnóstica.

Para llevar a cabo la reconstrucción de la ECGI, es necesario resolver tanto el problema directo como el inverso. En el modelado del corazón para el problema directo, se establece una relación entre las señales de entrada medidas en la superficie cardíaca y las señales de salida registradas en la superficie del torso.

1.4.12.2. Ondas electrocardiográficas. Las ondas básicas del electrocardiograma que corresponden a los episodios eléctricos generados por el corazón son: la onda P, complejo QRS, intervalo PR, segmento ST, onda T, onda U y el punto J; **Error! No se encuentra el origen de la referencia.** se presentan las ondas y los intervalos del ECG (Goldman, 2021).

Figura 1

Ondas ECG. Tomado de Merck Sharp & Dohme Corp. (s.f.).



1.4.13. Métricas por validación de métodos ECGI

1.4.13.1. Correlación de Pearson. El coeficiente de correlación de Pearson es una medida estadística que evalúa la relación entre dos variables continuas. Sin embargo, es importante destacar que esta prueba asume una relación lineal entre las variables; si la asociación entre los elementos no es lineal, el coeficiente puede no reflejar adecuadamente esta relación.

El coeficiente de correlación puede variar entre +1 y -1. Un valor de 0 indica que no hay asociación entre las dos variables. Un valor mayor que 0 sugiere una asociación positiva, lo que implica que a medida que aumenta el valor de una variable, también lo hace el valor de la otra. Por otro lado, un valor menor que 0 indica una asociación negativa, lo que significa que a medida que aumenta el valor de una variable, el valor de la otra disminuye. La Ecuación 1 muestra la fórmula de correlación de Pearson.

$$r_{xy} = \frac{\sum z_x z_y}{N} \quad (1)$$

1.4.13.2. Error relativo o absoluto. El error relativo o absoluto son medidas críticas para evaluar la precisión de los cálculos o predicciones en diversas disciplinas, desde la ingeniería hasta la ciencia de datos. El error absoluto simplemente representa la diferencia entre el valor real y el valor estimado, sin considerar su dirección. Por otro lado, el error relativo es una medida más relativa que expresa esta diferencia como un porcentaje del valor real. Ambos son herramientas esenciales para entender cuán cerca o lejos están nuestras estimaciones de la realidad, lo que nos permite ajustar nuestros métodos y mejorar la precisión de nuestros resultados. Esto se muestra en la Ecuación 2 y Ecuación 3.

$$\text{Error absoluto} = |\text{Valor medido} - \text{Valor exacto}| \quad (2)$$

$$\text{Error relativo} = \frac{|\text{Valor medido} - \text{Valor exacto}|}{\text{Valor exacto}} \quad (3)$$

1.4.13.3. Error cuadrático medio. El Error Cuadrático Medio (MSE, por sus siglas en inglés, Mean Squared Error) es una métrica de evaluación utilizada para medir la precisión de un modelo de regresión. El MSE cuantifica la media de los cuadrados de los errores o desviaciones, es decir, las diferencias entre los valores predichos y los valores reales. Es una medida de la calidad del estimador: cuanto más bajo sea el MSE, mejor será el modelo en términos de precisión predictiva. Se muestra en la Ecuación 4.

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (4)$$

1.5. Metodología

1.5.1. Tipo de investigación

Investigación cuantitativa cuasi experimental.

1.5.2. Línea de investigación

Dentro del programa Ingeniería Mecatrónica de la Universidad Mariana funciona el grupo de investigación GRIM, cuya línea de investigación está dada por el Diseño y Desarrollo Mecatrónico, que se encuentra enlazada con este proyecto de grado que está siendo realizado. Teniendo en cuenta el enfoque que se le ha dado a esta investigación es posible determinar que se suma al área de Ingeniería Computacional.

1.5.3. Hipótesis de investigación

Una arquitectura de machine learning basada en modelos de regresión de múltiples entradas y salidas de datos temporales de ECG tiene el potencial de resolver eficazmente el problema inverso de la electrocardiografía.

1.5.4. Matriz metodológica

1.5.4.1. Fase 1 – Análisis de Requerimientos y Selección de Base de Datos

1.5.4.1.1. Revisión Bibliográfica y Análisis del Estado del Arte. Conducir un estudio de la literatura científica y técnica en torno al procesamiento de imágenes electrocardiográficas (ECGI) y las arquitecturas de aprendizaje automático en el ámbito cardiológico. Este análisis debe abarcar investigaciones previas para comprender las soluciones y tecnologías emergentes, evaluando su potencial de replicabilidad y adaptabilidad a nuevos contextos.

1.5.4.1.2. Identificación de Requerimientos Funcionales. Detallar las necesidades específicas en la captura, procesamiento y análisis de ECGI. Este proceso tiene como objetivo esclarecer los requerimientos funcionales críticos del sistema, desde la perspectiva de la precisión diagnóstica hasta la usabilidad en entornos clínicos.

1.5.4.1.3. Evaluación y Selección de Base de Datos. Analizar comparativamente las bases de datos de ECGI disponibles, como la EDGAR Database, considerando aspectos clave como la accesibilidad, la diversidad y calidad de los datos (incluyendo señales ECG y EMG), y la conformidad con normativas legales y éticas. La meta es seleccionar la base de datos que optimice la replicabilidad y la relevancia clínica del proyecto.

1.5.4.2. Fase 2 – Desarrollo de Algoritmo para Modelo de Función Eléctrica Cardíaca

1.5.4.2.1. Diseño Preliminar del Algoritmo de Aprendizaje Automático. Definir la estructura y los componentes esenciales del algoritmo, seleccionando algoritmos de regresión adecuados para el análisis de señales temporales de ECG, con miras a una reconstrucción precisa de las imágenes.

1.5.4.2.2. Desarrollo de modelos y pruebas iniciales. Construir un modelo basado en métodos de regresión que cumplan con las especificaciones técnicas definidas, desarrollando el código necesario para el tratamiento y análisis de ECGI. Realizar pruebas preliminares para validar la funcionalidad básica y estimar las métricas de evaluación preliminares.

1.5.4.3. Fase 3 – Validación y evaluación de Requerimientos del Algoritmo

1.5.4.3.1. Desarrollo de pruebas y Establecimiento de Métricas de Calidad. Elaborar un diseño experimental para evaluar de manera integral la arquitectura, abarcando desde pruebas de rendimiento hasta la verificación de la precisión en la reconstrucción de imágenes y su correlación con los datos reales.

Definir métricas de calidad precisas para cuantificar la eficacia del sistema y ajustar la hiperparametrización para optimizar el rendimiento del modelo.

1.5.4.3.2. Validación y Ajustes Finales. Ejecutar un conjunto riguroso de pruebas de validación utilizando la base de datos elegida y datos de pruebas reales, para confirmar la exactitud y confiabilidad del sistema. Esta etapa es crucial para afinar el modelo y asegurar su aplicabilidad práctica en contextos clínicos.

2. Presentación y discusión de resultados

2.1. Análisis e interpretación de resultados

2.1.1. Análisis del primer objetivo

Seleccionar la base de datos de imágenes electrocardiográficas que mejor se ajuste a las necesidades del proyecto en términos de tamaño, calidad, diversidad y características de las señales, considerando los requerimientos del algoritmo y el objetivo de la investigación.

2.1.1.1. Revisión bibliográfica y análisis del estado del arte. A continuación, en la Tabla 1, se muestran los artículos seleccionados para el estudio y análisis de las bases para el presente proyecto.

Tabla 1

Matriz de artículos relacionados

Artículo	Técnica	Metodología	Resultados
Non-invasive Estimation of Atrial Fibrillation Driver Position Convolutional Neural Networks and Body Surface Potentials.	Redes neuronales convolucionales.	Datos de entrada, Arquitectura CNN, Entrenamiento y ajustes hiperparámetros, Métricas de rendimiento	Los resultados se visualizaron mediante una matriz de confusión y se proporcionaron métricas de precisión, sensibilidad y especificidad para cada etiqueta.
Solving Inverse Electrocardiographic Mapping Using Machine Learning and Deep Learning Frameworks.	Redes neuronales CNN Y RNN para el Procesamiento	Los datos se recopilaron de dos conjuntos: uno que registraba electrogramas desde un	Los resultados se evalúan utilizando coeficientes de correlación y se presenta una

	de señales chaleco alrededor del validación cruzada ECGI. cuerpo y otro alrededor tanto dentro del del corazón. mismo cerdo como entre diferentes cerdos.
Inista-net: learning nonlocal regularized iterative shrinkage-thresholding network for noninvasive cardiac transmembrane potential imaging.	Redes neuronales convolucionales. Se reemplaza el pasode recuperación con una red neuronal que utiliza capas convolucionales para lograr una mayor capacidad de aprendizaje profundo. Se observó que LNISTA-NET logra identificar con mayor precisión la ubicación del núcleo infartado y la zona isquémica, a pesar del ruido en los datos.
Deep learning formulation of electrocardiographic imaging integrating image and signal information with data-driven regularization.	Redes neuronales convolucionales. Redes Neuronales Artificiales Distribución Variacional y Divergencia Función de Pérdida Simulación de Datos Sintéticos El modelo generativo probabilístico demuestra su eficacia al generar múltiples soluciones desde una sola entrada, utilizando el muestreo del espacio latente.
Study on Parameter Optimization for Support Vector Regression in Solving the Inverse ECG Problem.	Vector de soporte (SVR) de Para la reconstrucción de los potenciales transmembrana a partir de los potenciales de superficies, se genera la comparación de métodos GA-SVVR, DE-SVR y PSO-SVR Al ver los resultados se destaca que el método PSO-SVR funciona mejor en la optimización de los parámetros en el algoritmo híbrido, con referencia a los otros dos.

Neutral Networks on	Redes	Uso de datos	Los resultados
Noninvasive	neuronales	experimentales,	incluyen una mejora
Electrocardiographic	convolucionales.	preprocesamiento de	significativa en el
Imaging Reconstruction:		datos, y la aplicación	rendimiento sobre
Preliminary Results		de métodos de mapeo	datos reales, mediante
		inverso de Tikhonov de	el coeficiente de
		orden cero (ZOT),	correlación de
		primero (FOT) y	Pearson (CC) y el
		segundo (ZOT)	error cuadrático
			medio (RMSE)

2.1.1.2. Identificación de requerimientos funcionales. La identificación de los requerimientos funcionales que serán tomados como referencia para el desarrollo del algoritmo permiten tener una visión clara sobre qué debe hacer, como debe funcionar, como debe ser usado e incluso evaluado. De este modo se propone estructurar un algoritmo que funcione a través de métodos de regresión que logre generalizar los resultados de predicción de señales temporales ECG al menos con dos conjuntos de datos.

La propuesta para este algoritmo a través de métodos de regresión se respalda bajo el concepto de abordar un enfoque diferente al que se le ha dado anteriormente, utilizar técnicas no convencionales como lo ha sido a través del método de Tikonov o métodos de Deep learning como lo pueden ser Redes Neuronales Recurrentes (por sus siglas en inglés, RNN), Redes Neuronales Convolucionales (por sus siglas en inglés, CNN) entre otros, utilizar técnicas no convencionales y obtener un modelo con métricas de precisión y calidad del ajuste del modelo para evaluar algoritmos ECGI, haciendo uso de las métricas MSE y Coeficiente de determinación (R^2).

Un modelo de regresión cuyo tiempo de entrenamiento sea apto es otro de los factores para tener en cuenta, esto se justifica a raíz de la eficiencia al momento de usar los recursos computacionales, un alto tiempo de entrenamiento requiere una cantidad significativa de recursos computacionales, además del tiempo de respuesta por parte del algoritmo.

2.1.1.3. Selección del conjunto de datos. En este apartado definiremos el conjunto de datos que serán usados para el estudio en curso, estos datos deberán contener señales temporales ECG; por este motivo fueron seleccionadas las bases de datos Utah y Kitpvc, dada a su consistencia y tienen datos de potenciales validados lo que ayuda a la generalización de los modelos.

Se realizarán pruebas con estas dos bases de datos haciendo uso de distintos métodos de regresión con el fin de comprobar el comportamiento de los datos contenidos, y así como resultado obtener valores que demuestren la efectividad no solo de la base de datos, sino que también de los métodos.

Estos Los datos Kitpvcproviene de la simulación de 17 latidos ventriculares en una malla basada en vóxeles utilizando un autómata celular, se interpolaron los BSMP de estos latidos simulados en la malla tetraédrica y se calcularon hacia delante utilizando el método de elementos finitos y el modelo de bidominio. Los potenciales extracelulares se extrajeron de la malla tetraédrica en 163 posiciones de electrodos y en los nodos de la malla pericárdica, y de la superficie endo y epicárdica.

Los datos Utah provienen del corazón aislado de un perro que se perfunde a través de la sangre de un perro de apoyo. Los datos se registraron a partir de una guía de electrodos de la cavidad pericárdica (256 electrodos) y un tanque del torso (192 electrodos). El torso se llenó con un fluido conductor que se aproximaba a la conductividad del torso humano. Se incluyen grabaciones de tres secuencias de activación diferentes. La primera es un latido ventricular anterior (avp), la segunda es un latido ventricular posterior (pvp) y la tercera es un latido sinusal. Las tres secuencias de activación se registraron simultáneamente en el tanque del torso y en el electrodo de la cavidad.

2.1.2. Desarrollo del segundo objetivo

Implementar algoritmos de aprendizaje automático de regresión con la capacidad para procesar señales temporales de ECG, optimizando su rendimiento mediante la selección y ajuste de hiperparámetros.

2.1.2.1. Diseño preliminar del algoritmo de aprendizaje automático. El diseño del algoritmo requiere de distintas etapas de modo que permita definir la estructura y componentes que tendrá; inicialmente se requiere seleccionar la base de datos con la que se va a trabajar, eligiendo una fuente confiable y de este modo evitar que tenga efectos perjudiciales sobre los resultados, la importancia de seleccionar una base de datos confiable se representa directamente en los valores resultantes que serán obtenidos a la hora de realizar el entrenamiento de los datos, y así cualquier deficiencia observada en los resultados se pueda entender que no surge a partir de los datos que están siendo usados. El algoritmo primero contará con una etapa donde se cargarán los datos y se hará elección de los datos a entrenar, que en este caso con señales temporales ECGi.

La propuesta para el estudio en curso es hacer uso de técnicas diferentes a algunas que han sido tratadas anteriormente con el fin de incursionar en este campo de ECGI, es por este motivo que se plantea usar distintos métodos de regresión que permitan obtener datos con altos estándares; de este modo la siguiente etapa del diseño del algoritmo es la selección y uso de los métodos de regresión adecuados para el análisis de las señales temporales. A esto se liga el entrenamiento de los datos donde se irán probando cada uno de los métodos, dado que cada una de las bases de datos que serán usadas cuentan con múltiples salidas, se requiere hacer uso de un regresor de múltiples salidas que permita recorrer cada salida a la hora de realizar el entrenamiento.

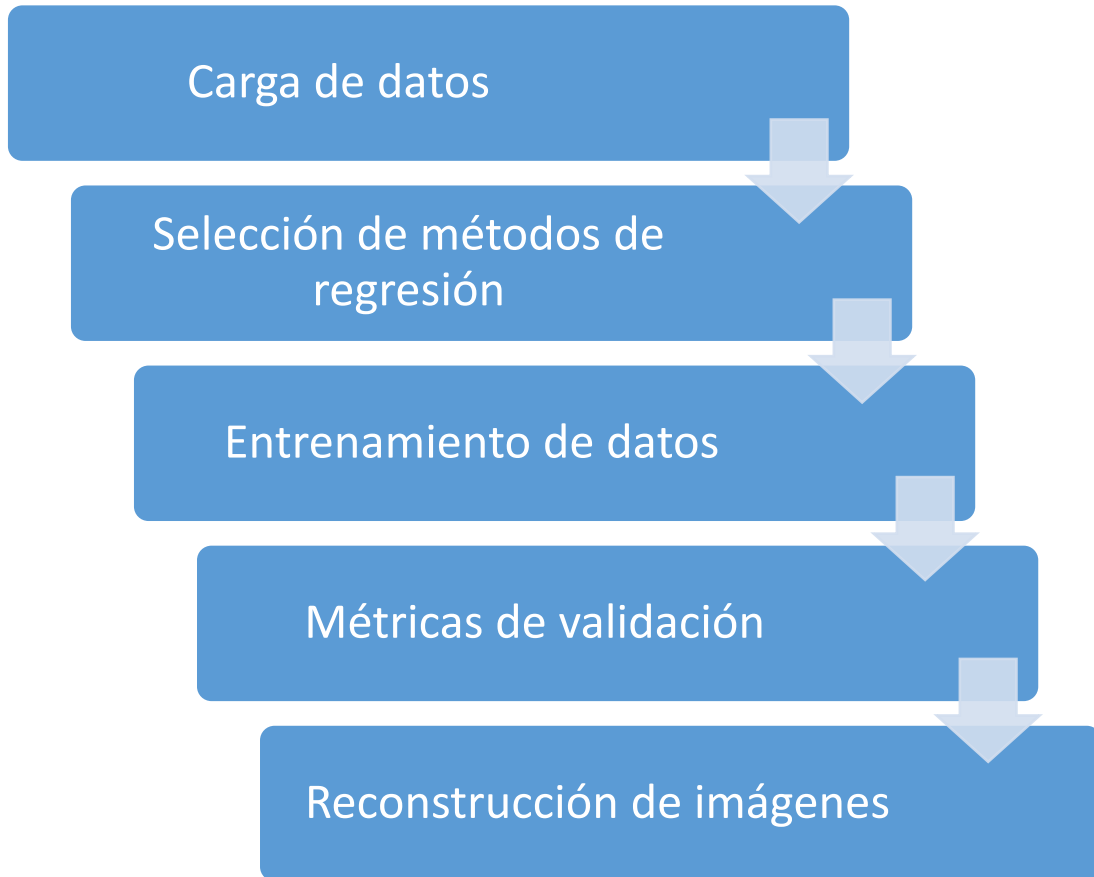
Las métricas de evaluación del algoritmo son de alta importancia debido a que gracias a estos parámetros se conoce la eficiencia, rendimiento, desempeño y precisión del modelo, además de permitir definir su relevancia de si los resultados son los esperados o incluso en definir si el algoritmo necesita algo más, como lo podrían ser ajustes de hiperparámetros o métodos para optimizar el modelo, esto con el fin de obtener estándares que sean representativos y de este modo permita realizar una comparación entre los métodos usados; incluso permiten identificar problemas en el modelo como podrían ser el sobreajuste de los datos, variabilidad de los datos y calidad del ajuste de los datos reales en comparación con los predichos.

Finalmente, como última etapa del algoritmo, se realizará la reconstrucción de las señales temporales ECG reales en comparación con los resultados obtenidos y de este modo visualizar la

exactitud con la que el modelo predice los datos permitiendo observar la similitud entre las señales, y por consiguiente poder determinar las conclusiones del funcionamiento del modelo.

Figura 2

Etapas del modelo



2.1.2.2. Desarrollo de modelo y pruebas iniciales. Para el desarrollo del algoritmo inicialmente se implementa la etapa donde se carga la base de datos, en este apartado se hará el uso de dos conjuntos de datos, estas serán aquellas que fueron seleccionadas con anteriormente ya que cumplían con los aspectos que fueron planteados para tener en cuenta como lo pueden ser la accesibilidad, diversidad y calidad de los datos. Seguido a esto se hará uso de los distintos métodos de regresión que fueron seleccionados para el entrenamiento de los datos de modo que permitan estudiar sus resultados y compararlos entre si con el fin de determinar cuál es el que ofrece mejores resultados en términos de desempeño, rendimiento, efectividad y exactitud.

Una vez definidos los métodos con lo que se trabajará, inicia la etapa de entrenamiento donde se utiliza el conjunto de datos seleccionado y se usan los parámetros que están definidos por defecto para cada método, este modelo será implementado en el lenguaje de programación Python, a través de Google Colab. Con el fin de iniciar las pruebas preliminares se hará con el conjunto de datos Kitpvc, se usa el 80% de los datos para entrenar y el 20% para validar.

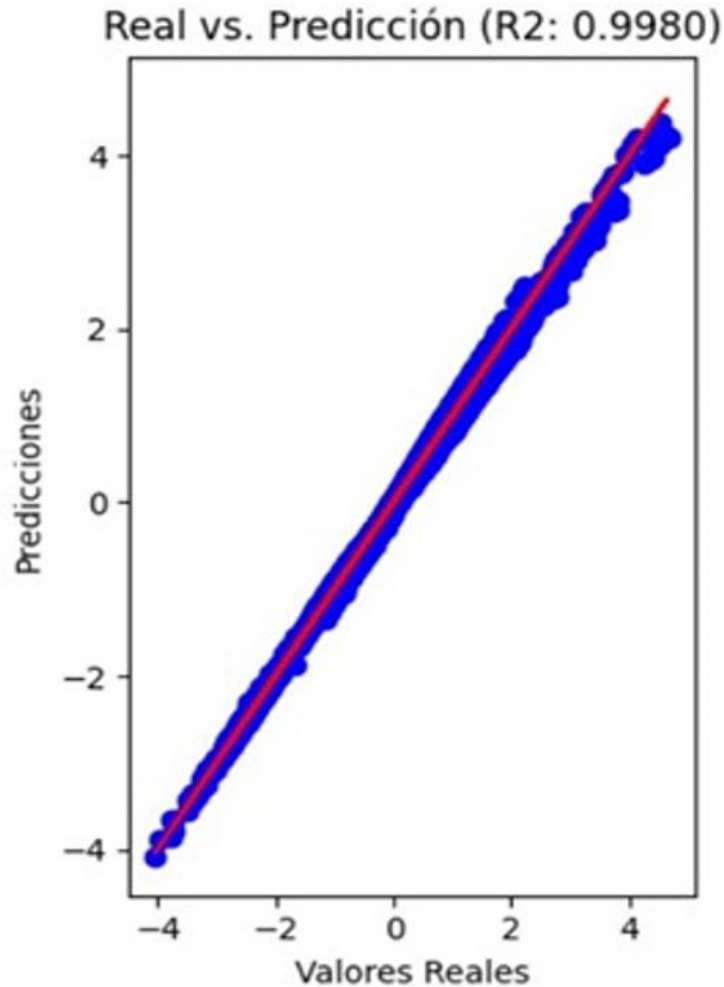
Como entradas tenemos las señales que incluidas contenidas en los conjuntos de datos y como salidas se tendrán los resultados que demuestran que tan bien se ajusta el modelo a los datos reales, además de su precisión y cómo se comportan los distintos modelos, para como finalidad obtener la reconstrucción de imágenes ECGI.

Para KNN Regressor, los parámetros por defectos que utiliza son: (N_neighbors = 5, Weights = Uniform, P = 2).

Como resultados se obtuvieron valores de $MSE = 0.002 \pm 0.0014$ de desviación estándar, $R2 = 0.998 \pm 0.00029$ desviación estándar, con un tiempo de respuesta de 9 segundos, esto demuestra un método sumamente optimo y apto para realizar pruebas futuras dado que el hecho de contar un tiempo de respuesta tan bajo brinda la flexibilidad de realizar ajustes sobre el modelo sin tener que esperar por largos lapsos de tiempo. El modelo demuestra eficiencia, teniendo buen rendimiento y desempeño, además de no tener alta variabilidad de los datos y tener un buen ajuste de los datos al modelo.

Figura 3

Resultados obtenidos KNN Regressor



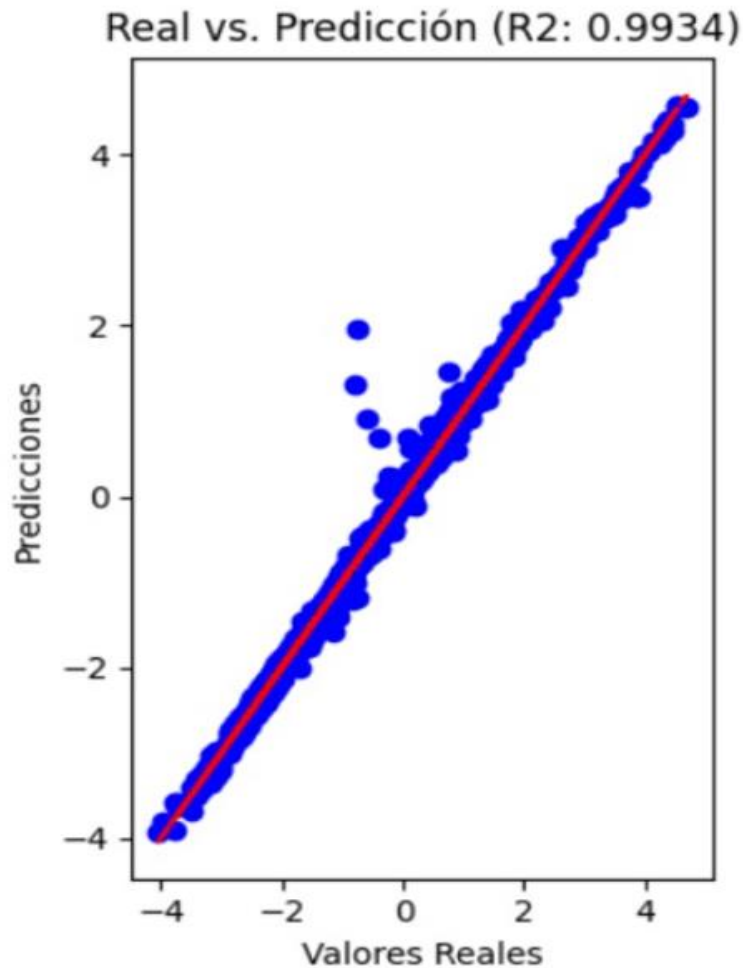
Para Decision Tree Regressor, los parámetros por defecto que utiliza son: (Max_depth = None (Los nodos se expanden hasta que todas las hojas sean puras o hasta que todas las hojas contengan menos muestras que min_samples_split), Min_samples_split = 2).

Como resultados se obtuvieron valores de $MSE = 0.006 \pm 0.00354$ de desviación estándar, $R2 = 0.993 \pm 0.00483$ de desviación estándar, con un tiempo de respuesta de 4 minutos 49 segundos, esto demuestra un método que sigue apto para realizar futuras pruebas, si bien se nota un incremento en comparación con el método anterior, sigue teniendo un tiempo de respuesta bajo, de

modo que no sería necesario esperar largos lapsos de tiempo. Además, el modelo aun demuestra eficiencia, teniendo buen rendimiento y desempeño, sigue contando con muy baja variabilidad de los datos y buen ajuste de los datos al modelo.

Figura 4

Resultados obtenidos Decision Tree Regressor



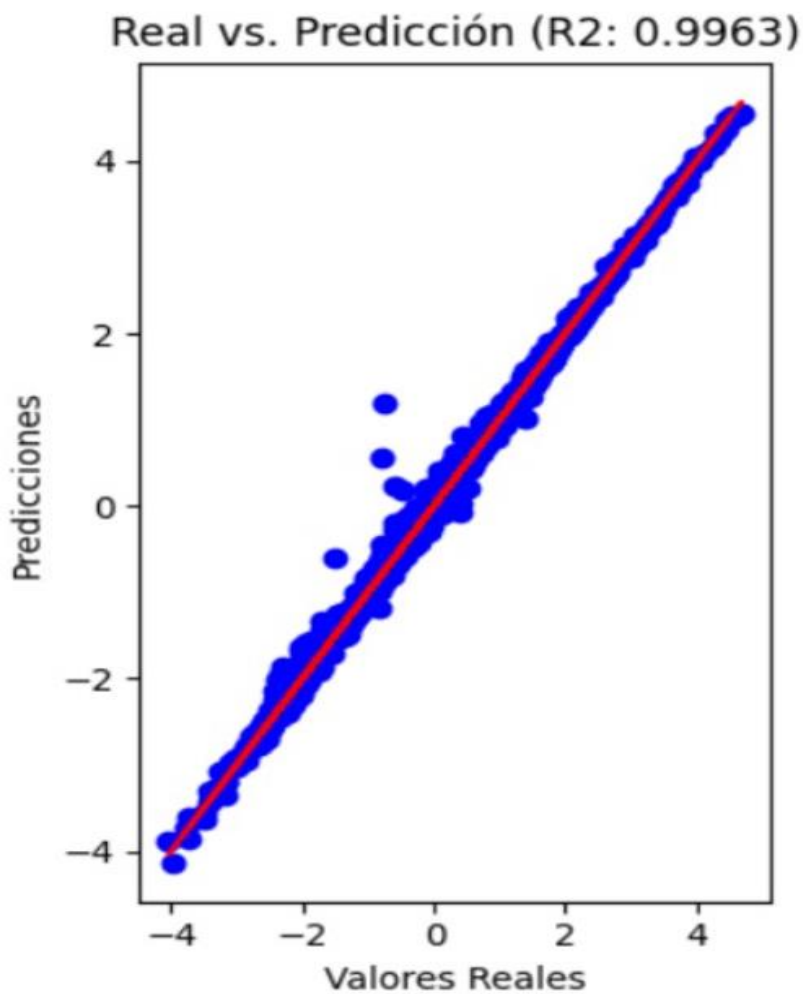
Para Bagging Regressor los parámetros que utiliza por defecto son: (Estimador = None (El estimador base es DecisionTreeRegressor.), N_estimators = 10).

Como resultados se obtuvieron valores de $MSE = 0.002 \pm 0.00211$ de desviación estándar, $R2 = 0.996 \pm 0.00228$ de desviación estándar, con un tiempo de respuesta de 27 minutos 44 segundos,

estos valores aun demuestran un modelo con buen desempeño, contando con muy baja variabilidad de los datos y un buen ajuste de los datos al modelo. Sin embargo, el mayor cambio por parte de este método de regresión en comparación con los anteriores es su tiempo de respuesta, incremento considerablemente de modo que se descartan futuras pruebas para este modelo haciendo uso de este método de regresión, el hecho de contar con un alto tiempo de respuesta descarta la posibilidad de realizar ajustes al modelo y estudiar sus resultados.

Figura 5

Resultados obtenidos Bagging Regressor

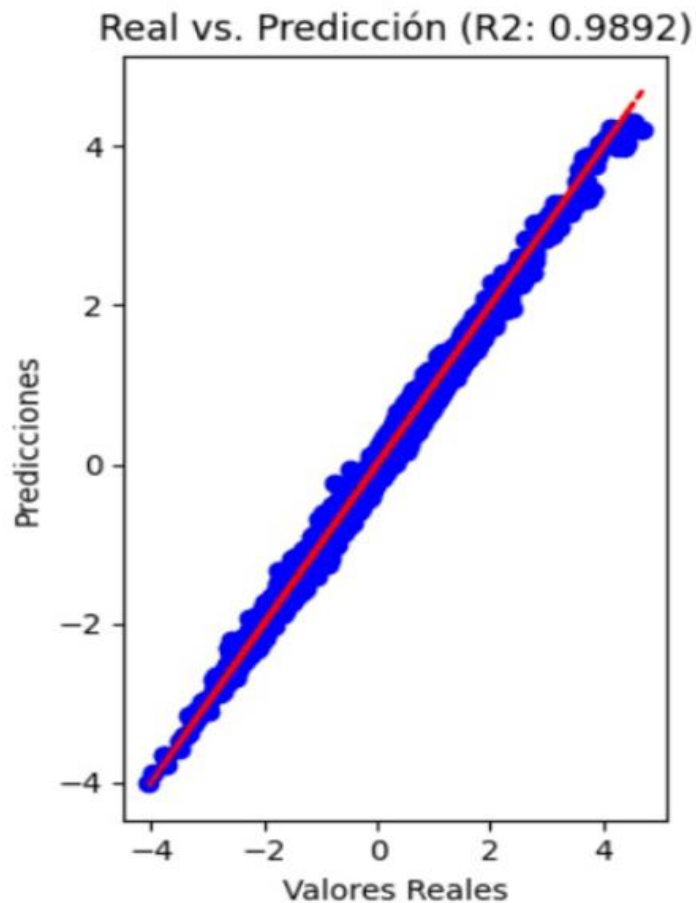


Para Adaboost Regressor parámetros que utiliza por defecto son: (Estimator = None, N_estimators 50, Learning rate = 1.0).

Como resultados se obtuvieron valores de $MSE = 0.011 \pm 0.00489$ de desviación estándar, $R^2 = 0.989 \pm 0.00294$ de desviación estándar, con un tiempo de respuesta de 1 hora 9 minutos 19 segundos, el modelo aún sigue entregando buenos resultados en cuestión de desempeño teniendo en cuenta las métricas, contando con baja variabilidad de los datos y un buen ajuste de los datos al modelo. Sin embargo, para este método de regresión el tiempo de respuesta ya es muy alto, lo que complica el hecho de realizar más pruebas, tomando en cuenta que se requeriría esperar largos lapsos de tiempo.

Figura 6

Resultados obtenidos AdaBoost Regressor

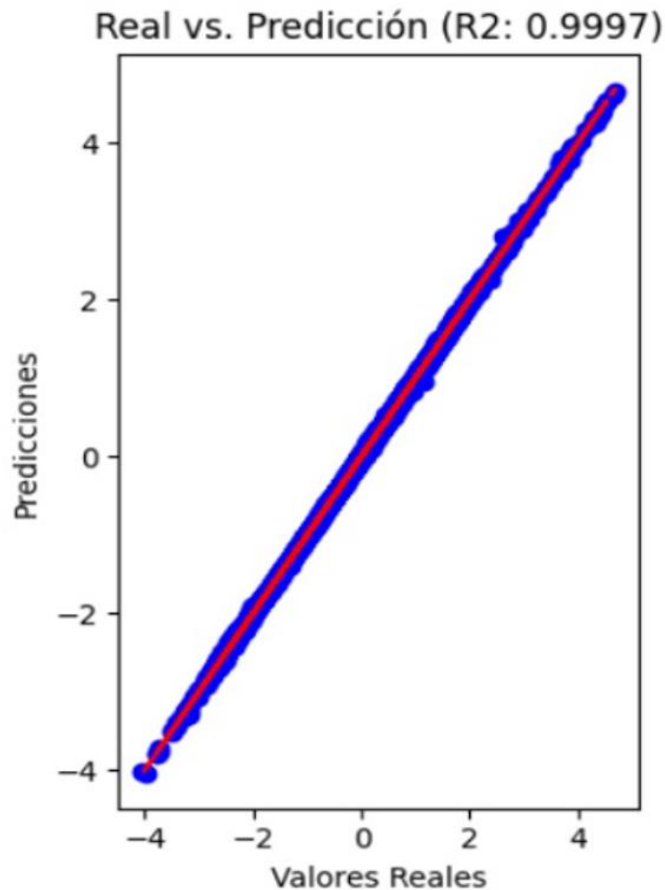


Para Extra Trees Regressor los parámetros que utiliza por defecto son: (Max_Depth = None, Min_samples_leaf = 1, Min_samples_split = 2).

Como resultados se obtuvieron valores de $MSE = 0.000 \pm 0.00013$ de desviación estándar, $R2 = 1.000 \pm 0.00017$ de desviación estándar, con un tiempo de respuesta de 2 horas 30 minutos 12 segundos, este caso en particular arroja una exactitud del 100% para realizar una predicción, se podría deducir que este es el mejor método y sería el indicado para continuar con las pruebas, pero el tan alto tiempo de respuesta lo descarta por completo, el intentar hacer más pruebas con este método tomaría demasiado tiempo por lo que no es óptimo y en términos de rendimiento no es el adecuado.

Figura 7

Resultados obtenidos Extra Tree Regressor

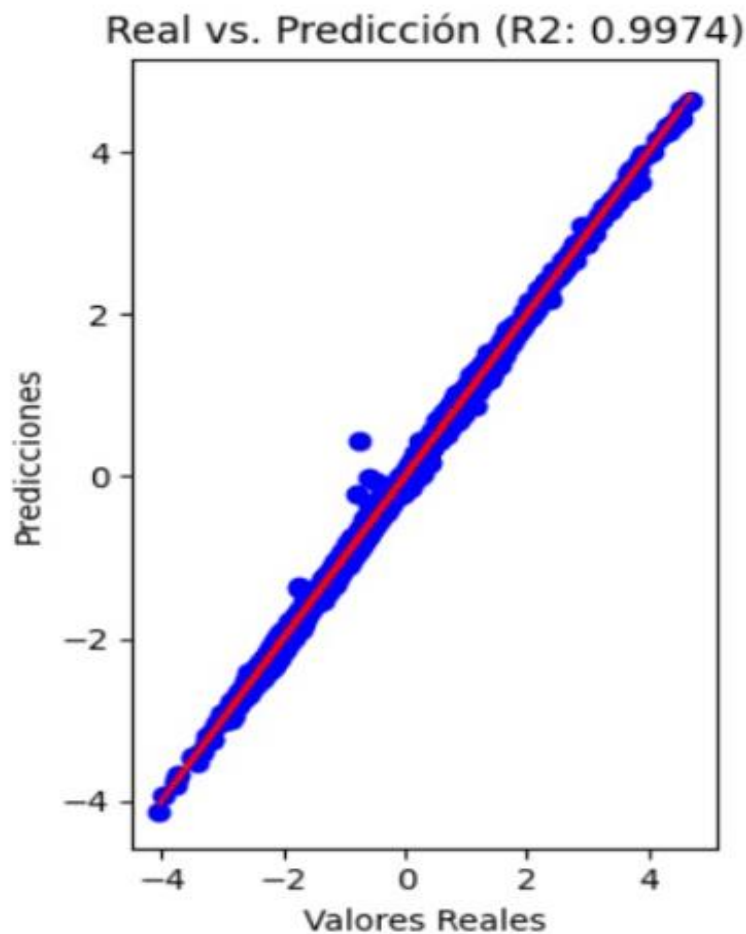


Para Gradient Boosting Regressor los parámetros que utiliza por defecto son: (N_estimators = 100, Learning_rate = 0.1, Max_depth = 3).

Como resultados se obtuvieron valores de $MSE = 0.002 \pm 0.00101$ de desviación estándar, $R^2 = 0.997 \pm 0.0012$ de desviación estándar, con un tiempo de respuesta de 3 horas 15 minutos 23 segundos, como se ha podido concluir en anteriores métodos, sin importar que en términos de desempeño el modelo arroje valores que permitan entender que cuenta con un buen ajuste de los datos o que cuenta con una baja de variabilidad de los datos, se descarta por completo su uso a raíz del alto tiempo de respuesta, determinando que no es bueno su rendimiento.

Figura 8

Resultados obtenidos Gradient Boosting Regressor



A continuación, como se muestra en la Tabla 2, se sintetizan los datos obtenidos y se muestra el tiempo de respuesta de cada algoritmo que se muestra en Google Colab al finalizar la compilación. Permitiendo tener una mejor representación de los resultados:

Tabla 2

Recopilación de datos obtenidos

Base de datos	Método de regresión	Métricas		Tiempo
		MSE	R2	
Kitpvc	Knn	0.002	0.998	9 seg
	Decision tree	0.006	0.993	4 min 49 sg
	Bagging	0.003	0.996	27 min 44 seg
	Adaboost	0.011	0.989	1 h 9 min 19 seg
	Extra Trees	0.000	1.0	2 h 30 min 43 seg
	Gradient Boosting	0.002	0.997	3 h 15 min 35 seg

Figura 9

Resultados MSE

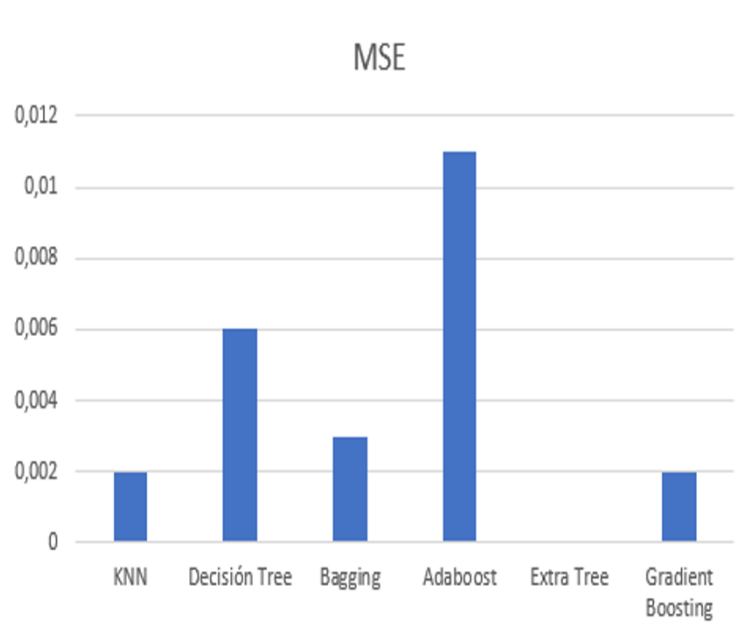
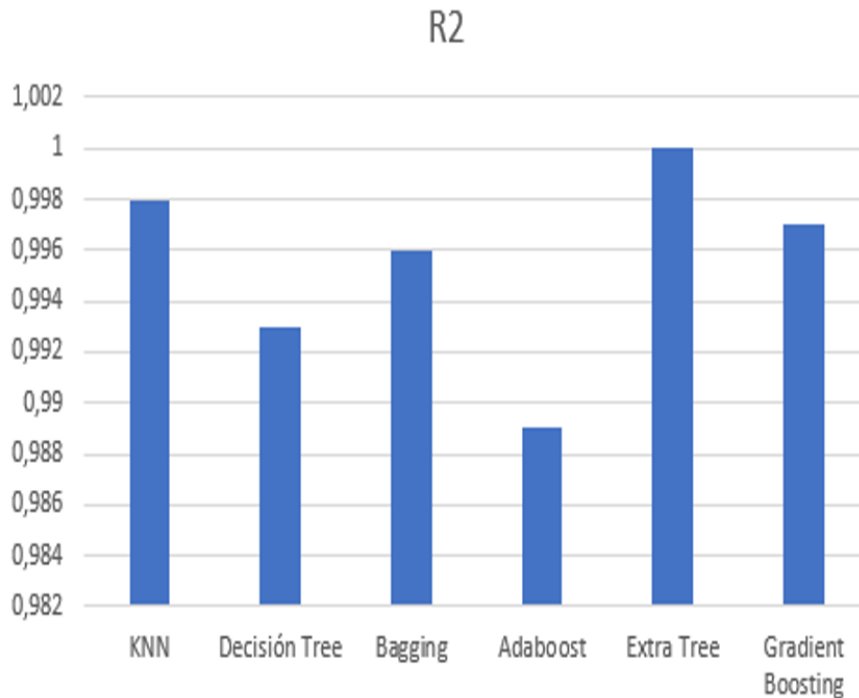


Figura 9

Resultados R2



Teniendo en cuenta estos resultados obtenidos en la Tabla 2, es posible determinar que el método con los mejores resultados en términos de eficiencia, exactitud, rendimiento y desempeño es el modelo entrenado haciendo uso de KNN Regressor. MSE es un valor que entre más cercano sea a cero demuestra el buen rendimiento del modelo, ya que teniendo esto como referencia indica que los valores predichos son muy cercanos a los valores reales, en este caso es posible determinar que al momento de reconstruir las señales temporales ECGI, serán prácticamente iguales a las reales; el valor de R2 demuestra la calidad del ajuste de los datos reales en comparación de los predichos, para esta métrica el valor entre más cercano sea a 1, demuestra un modelo con mejor ajuste a los datos reales. Como aspecto final a tomar en cuenta, para poder determinar qué tan optimo es el modelo, entra en juego el tiempo de respuesta por parte del algoritmo, lo ideal es tener un modelo con tiempo de respuesta lo menor posible; de este modo podemos definir que el mejor modelo será

aquel con valor de MSE más cercano a cero, R2 con valor más cercano a 1 y con el menor tiempo de respuesta.

Una vez ya seleccionado el modelo con las mejores características, teniendo en cuenta tantos sus resultados de mse y r2, como el tiempo de respuesta, interpretando esto como un bajo consumo de recursos computacionales, el paso a seguir es implementar el uso de hiperparámetros con el fin de estudiar si el modelo puede presentar mejorías en los resultados que se obtengan, optimizando el rendimiento del modelo. Como se muestra en la Tabla 3. El método de regresión KNN Regressor los hiperparámetros que pueden ser usados son:

Tabla 3

Hiperparámetros usados con KNN Regressor

N - neighbors	Weights		P
Es el número de vecinos que se utilizan para realizar la predicción	Controla como se ponderan las contribuciones de los vecinos y puede ser de dos tipos		Parámetro de potencia para la métrica de Minkowski: Cuando P = 1 usa la distancia de Manhattan y cuando P = 2 usa la distancia Euclidiana
	Uniform	Distance	
	Significa que los vecinos son ponderados por igual	los pesos inversamente proporcionales a la distancia: Los vecinos más cercanos tienen más influencia en la predicción que los vecinos más alejados	

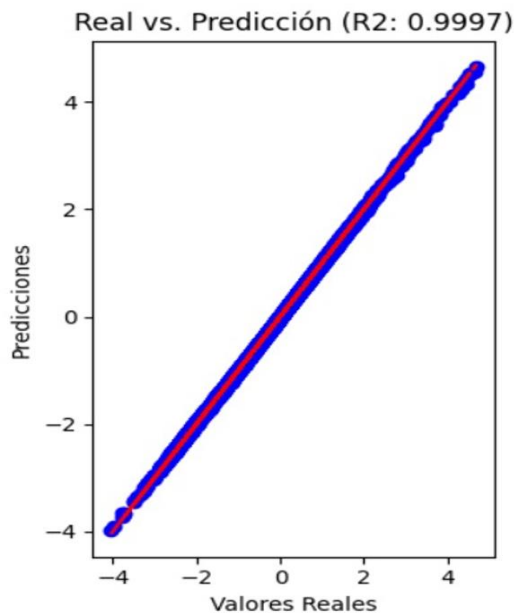
Estos distintos hiperparámetros pueden tomar diferentes valores, de modo que realizar distintas pruebas con el fin de obtener las mejores métricas que permitan obtener los mejores resultados posibles por parte del algoritmo sería tedioso y tomaría bastante tiempo, es por este motivo que se hará uso de GridSearch, esta es una técnica que permite encontrar la mejor combinación de hiperparámetros para el modelo, inicialmente se definen los posibles valores de cada uno y esta técnica se encarga de entrenar el modelo con cada uno de ellos para finalmente obtener la mejor combinación y por consiguiente un mejor modelo de entrenamiento.

Haciendo uso de la técnica GridSearch, se obtuvo la siguiente combinación de hiperparámetros para el método de regresión KNN Regressor: N_neighbors = 3, Weights = Distance, P = 1

Tomando como referencia estos valores para los hiperparámetros, se realiza el entrenamiento del modelo, y como resultado se obtuvo un valor para $MSE = 0.000 \pm 0.00024$ de desviación estándar, $R2 = 0.9997 \pm 0.00005$ de desviación estándar, con un tiempo de respuesta de 1 minutos 40 segundos, demostrando que esta es la mejor configuración posible para este modelo, entregando muy buenos resultados en términos de rendimiento, desempeño, eficiencia y exactitud. Como se evidencia en la Figura 11.

Figura 10

Métricas de validación



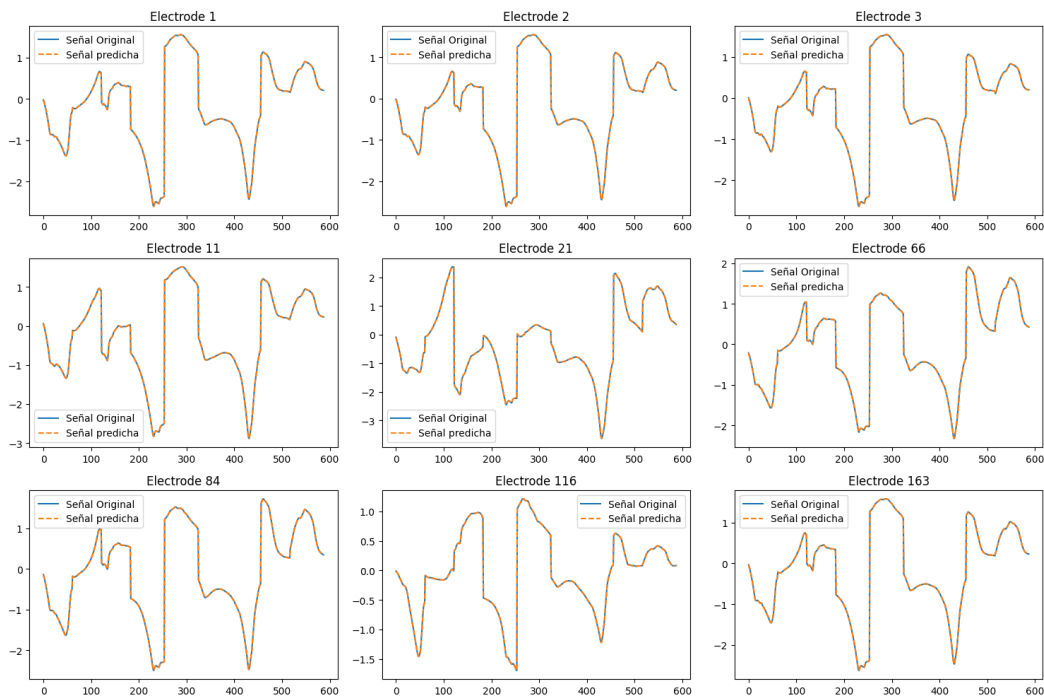
2.1.3. Análisis del tercer objetivo

Validar la funcionalidad y el rendimiento del sistema de reconstrucción de imágenes electrocardiográficas mediante la evaluación de métricas de calidad de imagen, comparando su rendimiento con otros métodos existentes para mejorar la precisión de la reconstrucción, utilizando bases de datos validadas.

2.1.3.1. Desarrollo de pruebas y Establecimiento de Métricas de Calidad. Para la reconstrucción de la imagen se hará uso del modelo con los mejores resultados obtenidos después de todas las pruebas realizadas, en este caso es el modelo que hace trabajar con el método de regresión Knn Regressor incluyendo sus hiperparámetros, con valores de $N_neighbors = 3$, $Weights = Distance$ y $P = 1$, cuyos resultados de entrenamiento fueron $MSE = 0.000 \pm 0.00024$ de desviación estándar, $R2 = 1.000 \pm 0.00005$ de desviación estándar, con un tiempo de respuesta de 46 segundos. A continuación, en la Figura 12, se presenta el resultado obtenido:

Figura 11

Reconstrucción de señal ECGI con KNN Regressor y conjunto de datos Kitpvc



Como es posible observar, el modelo permite reconstruir con éxito la señal ECGI, notando la exactitud de la señal predicha en comparación con la señal original, es de este modo que se logra visualizar que tan representativos son los valores obtenidos anteriormente.

El tiempo de respuesta para este modelo que permite reconstruir una imagen ECGI fue de 1 minuto 46 segundos, esto nos permite reafirmar que el mejor modelo es aquel que usa Knn regressor, resaltando la importancia de que siempre contó con un bajo tiempo de respuesta.

2.1.3.2. Validación y ajustes finales. Con el fin de validar el comportamiento de los modelos, se realizará una prueba con la base de datos de Utah, y se llevará a cabo con los dos mejores métodos de regresión, en este caso serán KNN Regressor y Decision Tree Regressor, un factor a tener en cuenta es el tiempo de respuesta de cada uno, a raíz de tener los menores tiempos, será posible realizar las pruebas de manera más rápida. Para esta prueba no será necesaria la Hiperparametrización, dado que solo se plantea comprobar el comportamiento de los modelos. Un factor a tener en cuenta es que posiblemente los tiempos de respuesta aumenten debido a que el conjunto de datos de Utah es mayor que el de Kitpvc.

Para el modelo que hace uso del método de regresión KNN Regressor, se obtuvieron valores de $MSE = 0.000 \pm 0.00006$ de desviación estándar, $R2 = 0.0998 \pm 0.00154$ de desviación estándar, con un tiempo de respuesta 1 min 7 segundos, lo que demuestra un modelo muy estable, con resultados representativos en términos de rendimiento y desempeño, se aprecia incremento en el tiempo de respuesta, pero esto es debido a que esta base de datos cuenta con mayor cantidad de salidas.

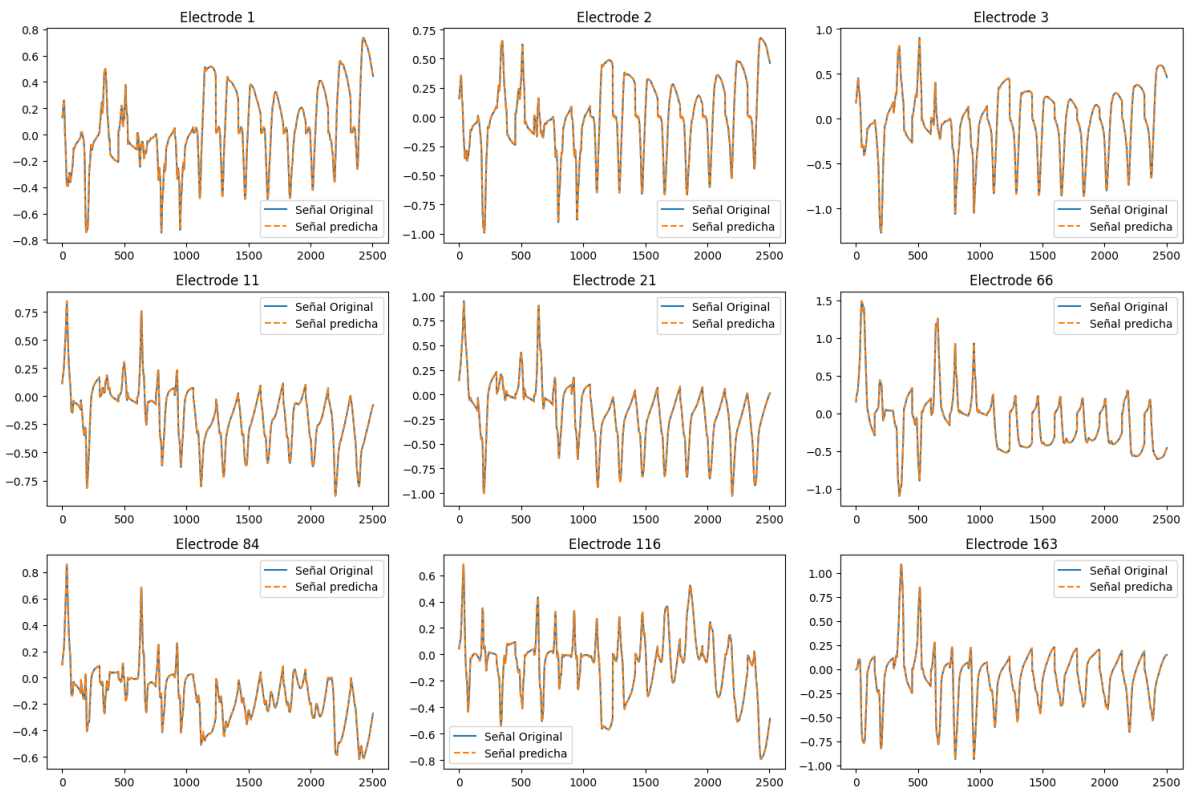
Para el modelo que hace uso del modelo de regresión Decision Tree Regressor, se obtuvieron valores de $MSE = 0.000 \pm 0.00016$ de desviación estándar, $R2 = 0.996 \pm 0.00189$ de desviación estándar, con un tiempo de respuesta de 42 minutos 56 segundos, en este caso se observa un aumento con creces en el tiempo de respuesta, de este modo es posible descartar seguir usando este método de regresión ya que, en caso de requerir más pruebas o distintos ajustes sobre el modelo, llevaría mucho tiempo realizarlos. Teniendo en cuenta los distintos aspectos para definir qué modelo es apto para continuar con las pruebas y el estudio en curso, es necesario trabajar con el

método de regresión KNN Regressor, ya que, al realizar el entrenamiento de datos, este tiene el menor tiempo de respuesta.

Finalmente, como ultima última prueba, se procederá a hacer la reconstrucción de la señal ECGI haciendo uso de Knn regressor incluyendo sus hiperparámetros y entrenando el conjunto de datos de Utah. A continuación, en la Figura 13, se presenta el resultado obtenido:

Figura 12

Reconstrucción de señal ECGI con KNN Regressor y conjunto de datos Utah



Como es posible observar, el modelo permite reconstruir con éxito la señal ECGI, notando la exactitud de la señal predicha en comparación con la señal original, resaltando el correcto funcionamiento del modelo, sin importar con qué conjunto de datos se esté trabajando. El tiempo

de respuesta para este modelo que permite reconstruir una imagen ECGI fue de 30 minutos 1 segundos

Finalmente, con el propósito de demostrar la importancia del resultado obtenido, se tomó como referencia el estudio Neural Networks on Noninvasive Electrocardiographic Imaging Reconstruction: Preliminary Results, con la intención de comparar los resultados obtenidos en dicho estudio con el estudio realizado, entre métodos tradicionales como el de Tikhnov, el método que fue implementado en este estudio y métodos más avanzados como las Redes Neuronales Convolucionales (por sus siglas en ingles, CNN). A continuación, en la Figura 14, se presentan los resultados obtenidos:

Como es posible observar, los resultados obtenidos permiten darle relevancia y veracidad al estudio realizado, a raíz de obtener un menor error en la precisión del modelo en comparación con los otros dos, de este modo es posible resaltar el hecho de que se puede abordar esta problemática desde enfoques diferentes a los que normalmente se suelen recurrir, abriendo paso a que se puedan realizar más investigaciones e implementar más métodos, permitiendo contribuir al avance en el area de la electrocardiografía, teniendo una vista en el campo clínico.

Figura 13

Comparación de resultados entre los estudios

Desarrollo e implementación de un algoritmo de aprendizaje automático para la reconstrucción de imágenes electrocardiográficas (ECGI)		Neural networks on Noninvasive Electrocardiographic Imaging Reconstructions: Preliminary results	
Método	Resultado	Método	Resultado
Regresor KNN con Hiperparámetros	0.999	Tikhonov orden cero	0.864
		RNN	0.960

3. Conclusiones

Los resultados obtenidos indican que el modelo desarrollado utilizando técnicas de machine learning, específicamente el algoritmo KNN Regressor con la selección de hiperparámetros adecuados, ofrece una precisión y rendimiento satisfactorios en la reconstrucción de señales de ECGI. Esto sugiere que es posible establecer un modelo confiable para este contexto sin incurrir en costos computacionales excesivos. El uso de técnicas de machine learning, como el KNN Regressor, permite evitar la necesidad de recurrir a modelos tradicionales como el modelo de Tikhonov. Estos modelos tradicionales suelen requerir un alto costo computacional y tiempo de resolución significativo, lo cual puede ser evitado al utilizar modelos de machine learning, que son más eficientes en términos de tiempo de entrenamiento y predicción. Una vez que el modelo de machine learning ha sido entrenado, el tiempo necesario para realizar predicciones es mínimo. Los modelos de machine learning, como el KNN Regressor utilizado en este estudio, ofrecen una alternativa eficaz y eficiente para resolver problemas de reconstrucción de señales de CGI. Además de proporcionar resultados precisos, estos modelos permiten reducir significativamente los costos computacionales y los tiempos de cálculo.

4. Recomendaciones

Buscar nuevos conjuntos de datos de señales de ECGI que puedan proporcionar información adicional, con el fin de seguir estudiando el comportamiento de los métodos de regresión enfocados en la reconstrucción de imágenes ECGI, permitiendo evaluar la capacidad con la que los modelos cuentan para ajustarse a los distintos conjuntos de datos. Extender la investigación para evaluar la aplicabilidad de los modelos desarrollados en entornos clínicos reales, colaborando con profesionales de la salud para evaluar la utilidad de los modelos en el diagnóstico y tratamiento de enfermedades cardiovasculares. Desarrollar herramientas de software accesibles y fáciles de usar que implementen los modelos propuestos, con el fin de facilitar su adopción por parte de la comunidad científica y clínica en el análisis de señales de ECGI. Se recomienda explorar el potencial de los métodos de deep learning. Esta investigación puede abrir nuevas oportunidades para avanzar en el campo de la reconstrucción de señales de ECGI, su aplicación en el diagnóstico y tratamiento de enfermedades cardiovasculares.

Referencias bibliográficas

Ley 1090 de 2006. (2006, 6 de septiembre). Congreso de la República. Diario oficial No 46.383.

<http://www.secretariasenado.gov.co/senado/basedoc/ley>

Analytics Lane. (2021, diciembre 17). Regresión de Vectores de Soporte (SVR, Support Vector Regression). *Recuperado de* <https://www.analyticslane.com/2021/12/17/regresion-de-vectores-de-soporte-svr-support-vector-regression/#:~:text=La%20Regresi%C3%B3n%20de%20Vectores%20de%20Soporte%20%28SVR%2C%20del,Machines%29%20para%20la%20creaci%C3%B3n%20de%20modelos%20de%20clasificaci%C3%B3n.>

Brindley, K. (1988). *Sensors and Transducers*. Heinemann Newnes. Camacho, J. A. (15 de 06 de 2021). *JacobSoft. Obtenido de PCA Principal Component Analysis: https://www.jacobsoft.com.mx/es_mx/analisis-del-componenteprincipal/#:~:text=El%20PCA%20es%20una%20herramienta,contenida%20en%20el%20conjunto%20grande.*

Cluitmans, M., Brooks, D. H., MacLeod, R., Dössel, O., Guillem, M. S., van Dam, P. M., Svehlikova, J., He, B., Sapp, J., Wang, L., & Bear, L. (2018). Validation and Opportunities of Electrocardiographic Imaging: From Technical Achievements to Clinical Applications. *Frontiers in Physiology*, 9, 1305. <https://doi.org/10.3389/fphys.2018.01305>

Compartimos, S. (2024). *Introducción a Light GBM y Gradient Boosting*. Compartimos S. *Recuperado de* <https://www.compartimoss.com/revistas/numero-53/introduccion-a-light-gbm-y-gradient-boosting/>

Esri, E. (2024). How Decision Tree Classification and Regression Works. *Recuperado de* <https://pro.arcgis.com/es/pro-app/latest/tool-reference/geoai/how-decision-tree-classification-and-regression-works.htm>

Goldman, L. (2021). *Tratado de medicina interna*. Elsevier Limited (UK).

IBM. (2024). Bagging Progressor. *Recuperado de* <https://www.ibm.com/mx-es/topics/bagging#:~:text=Este%20método%20de%20remuestreo%20genera,la%20misma%20instancia%20varias%20veces>

Jagan-Singhh. (2024). *Extra Extremely Randomized Trees*. Medium. Recuperado de <https://jagan-singhh.medium.com/extra-extremely-randomized-trees-5ce9026bd07f>

Jiang, Mingfeng & Zhu, Lingyan & Wang, Yaming & Xia, Ling & Shou, Guofa & Liu, Feng & Crozier, Stuart. (2011). Application of kernel principal component analysis and support vector regression for reconstruction of cardiac transmembrane potentials. *Physics in medicine and biology*. 56. 1727-42. 10.1088/0031-9155/56/6/013.

Jiang, M., Jiang, S., Zhu, L., Wang, Y., Huang, W., & Zhang, H. (2013). Study on Parameter Optimization for Support Vector Regression in Solving the Inverse ECG Problem. *Computational and Mathematical Methods in Medicine*, 15, 56-80. <https://doi.org/10.1155/2013/158056>

Jiang, M., Liu, F., Wang, Y., Shou, G., Huang, W., & Zhang, H. (2012). A Hybrid Model of Maximum Margin Clustering Method and Support Vector Regression for Noninvasive Electrocardiographic Imaging. *Computational and Mathematical Methods in Medicine*, 16, 62-81. <https://doi.org/10.1155/2012/436281>

Lux, R. (2010). Body Surface Potential Mapping Techniques. *Comprehensive Electrocardiology*., 1361-1374. doi:10.1007/978-1-84882-046-3_31

Madrigal, E. (2022, octubre 28). Conoce las métricas de precisión más comunes para Modelos de Regresión. Grow Up. <https://www.growupcr.com/post/metricas-precision>

Pedrón, J. (2010). Implementación y estudio de métodos numéricos para la resolución del problema directo e inverso de la electrocardiografía (Tesis de maestría). *Repositorio Universidad de Valencia*.

Robinson, M., & Curzen, N. (2009). Electrocardiographic Body Surface Mapping: Potential Tool for the Detection of Transient Myocardial Ischemia in the 21st Century, doi:10.1111/j.1542-474X.2009.00284.x.