

Implementación de algoritmos de selección de señales ECG para aprendizaje contrastivo, con el fin de mejorar la precisión en la clasificación de señales con arritmias

Laura Sofía Valero García y Erika Dayana Porras Torres

Trabajo de Grado para Optar al Título de Ingeniero Electrónico

Director

Alejandro Navarro Luna

Ingeniero Electrónico

Co-director

Carlos Augusto Fajardo Ariza

Doctor en Ingeniería

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones

Bucaramanga

2023

Dedicatoria

Este trabajo viene dedicado para todas aquellas personas que apoyaron el desarrollo y ejecución de este trabajo de grado.

En especial reconozco la permanente presencia de Dios en nuestro camino de vida.

Agradecimientos

Agradecemos a nuestra familia que nos dió su apoyo durante el desarrollo de nuestra carrera. También agradecemos a nuestros amigos y compañeros por las vivencias de estos inolvidables años de universidad.

Un reconocimiento y agradecimiento importante lo realizamos a nuestro director de trabajo de grado, por dedicar su tiempo, experiencia y conocimiento en la guía de nuestro proyecto.

Tabla de Contenido

Introducción	13
1 Objetivos	16
2 Contextualización	17
2.1 Redes neuronales convolucionales	17
2.2 Aprendizaje contrastivo	18
2.3 Clustering	18
2.4 k-means	20
3 Revisión de la literatura	22
3.1 Trabajos relacionados	22
4 Metodología	26
4.1 Icentia11k	26
4.2 Selección de datos	28
4.2.1 Algoritmo DCN:	28
4.2.2 Algoritmo CX-dagan:	31
4.3 Pre-entrenamiento	32
4.4 Entrenamiento	34

5 Resultados	35
5.1 Rendimiento del modelo	36
6 Conclusiones	41
Referencias bibliográficas	41

Lista de Figuras

Figura 1	<i>Arquitectura típica de una red neuronal convolucional (CNN).</i>	17
Figura 2	<i>Una taxonomía de enfoques de agrupación.</i>	19
Figura 3	<i>Resultado de la aplicación de k-means a un conjunto de datos no agrupados.</i>	21
Figura 4	<i>Etapas de un algoritmo de aprendizaje contrastivo: aumento de datos, extracción de características y pérdida contrastiva.</i>	24
Figura 5	<i>Distribución del dataset</i>	28
Figura 6	<i>Estructura etapa de selección de datos</i>	29
Figura 7	<i>Fórmula de función de costo</i>	30
Figura 8	<i>Arquitectura DCN</i>	30
Figura 9	<i>Distribución del set de datos</i>	32
Figura 10	<i>Arquitectura de aprendizaje contrastivo aplicado en señales ECG</i>	35
Figura 11	<i>Valores F1 y AUC alcanzados con el modelo M1 (66k parámetros) en el test set utilizando 5k, 10k y 15k datos etiquetados.</i>	37
Figura 12	<i>Valores F1 y AUC alcanzados con el modelo M2 (250k parámetros) en el test set utilizando 5k, 10k y 15k datos etiquetados.</i>	38

Lista de Tablas

Tabla 1	Estadísticas del conjunto de datos	27
Tabla 2	Valores promedio del puntaje F1 en el test set Icentia11k.	39
Tabla 3	Promedio del AUC en el test set Icentia11k.	40

Lista de Apéndices

(Ver apéndices adjuntos en el CD y pueden visualizarlos en la Base de Datos de la Biblioteca UIS)

Glosario

Aprendizaje contrastivo: es aquella técnica de deep learning que le permite a un modelo aprender sobre los datos sin necesidad de utilizar etiquetas, reconociendo las similitudes y diferencias en las características del set de datos.

Aprendizaje auto-supervisado: proceso de aprendizaje automático donde el modelo se entrena a si mismo al auto generar etiquetas.

Aprendizaje no supervisado: es un tipo de machine learning, en donde se utilizan datos no etiquetados, busca encontrar la estructura y las relaciones de los datos a partir de las entradas.

Arritmia: latidos anormales del corazón.

Clustering: agrupar objetos sin etiquetas por similitud en sus características.

Fibrilación auricular: tipo más común de arritmia, comúnmente se comporta como frecuencia cardíaca irregular y rápida.

Fine-tuning: tomar un modelo ya entrenado para una tarea determinada y ajustarlo para que realice una tarea similar.

Machine learning: darle la habilidad a un sistema de computo de realizar una tarea sin antes darle una explicación o paso a paso de como hacerlo.

Red Neuronal Convolutiva (CNN): es una arquitectura de red utilizada en aprendizaje profundo, suelen usarse para reconocer patrones, clases o categorías.

Señal ECG: una señal de electrocardiograma, mide las señales eléctricas que controlan el ritmo cardíaco.

Resumen

Título: Implementación de algoritmos de selección de señales ECG para aprendizaje contrastivo, con el fin de mejorar la precisión en la clasificación de señales con arritmias *

Autor: Laura Sofia Valero y Erika Dayana Porras Torres **

Palabras Clave: Aprendizaje contrastivo, Redes neuronales convolucionales, Señales ECG, Clustering, Aprendizaje auto-supervisado, Aprendizaje no supervisado, Arritmias cardíacas, Fibrilación auricular.

Descripción: La señal ECG proporciona información sobre el sistema cardiovascular, sin embargo, clasificar este tipo de señales es complicado y consume tiempo. La interpretación de grandes volúmenes de datos ECG es realizada por especialistas, lo que puede llevar a fatiga y errores médicos. El desarrollo de sistemas de deep learning ha abierto nuevas posibilidades en la interpretación automatizada de las señales ECG, pero se requieren conjuntos de datos etiquetados, que son costosos y pueden plantear problemas de privacidad. Además, los conjuntos de datos existentes son limitados en comparación con la diversidad de patrones de ECG. Por lo tanto, es necesario buscar alternativas que mejoren la clasificación de señales ECG sin depender únicamente de grandes conjuntos de datos etiquetados. Por medio de este proyecto se aporta a la búsqueda de soluciones utilizando señales ECG no etiquetadas con el objetivo de mejorar la clasificación de arritmias cardíacas de una CNN implementando distintos algoritmos de selección de datos para el aprendizaje contrastivo no supervisado durante el pre entrenamiento del CNN. Esta implementación se realizó en el lenguaje de programación python. El sistema utilizó la base de datos Icentia11k con 11.000 muestras de señales ECG.

* Trabajo de grado

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y telecomunicaciones.
Director: Alejandro Navarro Luna, Ingeniero Electrónico. Codirector: Ph.D Carlos Augusto Fajardo Ariza

Abstract

Title: Implementation of ECG signal selection algorithms for contrastive learning in order to improve accuracy in classification of signals with arrhythmias. *

Author: Laura Sofia Valero García y Erika Dayana Porras Torres **

Keywords: Contrastive learning, Convolutional Neural Networks, ECG signal, Clustering, Self-Supervised Learning, Unsupervised learning, Cardiac arrhythmias, Atrial fibrillation.

Description: The ECG signal provides information about the cardiovascular system, however, classifying this type of signals is complicated and time consuming. Interpretation of large volumes of ECG data is performed by specialists, which can lead to fatigue and medical errors. The development of deep learning systems has opened up new possibilities in the automated interpretation of ECG signals, but labeled data sets are required, which are expensive and can raise privacy concerns. Furthermore, existing data sets are limited compared to the diversity of ECG patterns. Therefore, it is necessary to look for alternatives that improve the classification of ECG signals without relying solely on large sets of labeled data. Through this project, we contribute to the search for solutions using unlabeled ECG signals with the objective of improving the classification of cardiac arrhythmias of a CNN implementing different data selection algorithms for unsupervised contrastive learning during CNN pretraining. This implementation was done in the Python programming language. The system utilized the Icential1k database, which consists of 11,000 samples of ECG signals.

* Bachelor Thesis

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y telecomunicaciones. Director: Alejandro Navarro Luna, Ingeniero Electrónico. Codirector: Ph.D Carlos Augusto Fajardo Ariza

Introducción

Las arritmias cardíacas son uno de los principales problemas de salud en el mundo, un trastorno que surge cuando los impulsos eléctricos del corazón no funcionan correctamente, lo que en el peor de los casos puede provocar un accidente cerebro vascular, insuficiencia cardíaca o muerte súbita (Clinic, 2018).

En Colombia, desde finales de los años setenta las enfermedades cardiovasculares empezaron a ser reconocidas como causas de morbilidad y mortalidad. Actualmente, la enfermedad isquémica cardíaca, el accidente cerebro vascular, la diabetes y la enfermedad hipertensiva ocupan puestos 1, 3, 8 y 9, respectivamente, entre las principales causas de mortalidad en Colombia (de Colombia, s.f.).

La fibrilación auricular (AFib), la cual es la arritmia más común, es responsable de una alta mortalidad, morbilidad y aumento de los costos de atención médica (Khan et al., 2020).

Los latidos del corazón se pueden detectar mediante electrocardiogramas (ECGs), que se pueden procesar para detectar y clasificar variedad de enfermedades cardíacas (Fonseca et al., 2022). Por medio de un ECG se representa la actividad eléctrica del corazón y esta puede estar compuesta por varias derivaciones dependiendo de cuantos electrodos se utilicen para la medición. Los ECGs se registran de forma rutinaria en unidades de cuidados intensivos, así como en monitores ambulatorios y portátiles, lo que produce una gran cantidad de datos todos los días.

Se han desarrollado numerosos sistemas basados en aprendizaje supervisado utilizando conjuntos de datos ECG con etiquetas proporcionadas por cardiólogos para clasificar los latidos car-

díacos en clases normales y varias anomalías (Fonseca et al., 2022). Sin embargo, los cardiólogos solo pueden analizar y etiquetar una pequeña porción de la cantidad masiva de datos ECG para indicar anomalías cardíacas comunes. Además, la mayoría de los conjuntos de datos ECG etiquetados se adquieren en entornos controlados, como hospitales y clínicas, y contienen muestras muy limitadas en comparación con la amplia variedad de patrones de ECG que pueden ocurrir en diferentes condiciones fisiológicas y fisiopatológicas. Por ejemplo, se muestra que los patrones en los datos ECG adquiridos de personas que experimentan estrés o padecen diabetes mellitus difieren de los adquiridos en condiciones normales (Fonseca et al., 2022).

Como tal, recientemente se han propuesto varios métodos de aprendizaje no supervisados para analizar datos ECG sin necesidad de etiquetas proporcionadas por cardiólogos, un proceso denominado clustering de datos ECG (Nezamabadi et al., 2022). El uso de clustering o agrupación de datos en los últimos años ha sido importante debido a su capacidad para agrupar objetos con características similares. En el contexto de la clasificación de arritmias cardíacas, el clustering permite identificar patrones y agrupar señales ECG que comparten características similares, lo que facilita la detección y clasificación de diferentes tipos de arritmias. Al utilizar técnicas de clustering, se pueden descubrir estructuras ocultas en los datos y obtener una comprensión más profunda de las relaciones entre las señales ECG, lo que contribuye a un análisis más preciso y eficiente de las arritmias cardíacas. Sin embargo, estas técnicas requieren una cantidad considerable de datos etiquetados. Icentia11k (Tan et al., 2019) aparece como la base de datos más grande de arritmias cardíacas de acceso público. Tiene una cantidad sustancial de señales ruidosas ECG de una derivación, estructurada en pacientes y segmentos. Esta base de datos se utilizará para en este

proyecto. Consta de datos suministrados por 11,000 pacientes y 2 billones de latidos etiquetados.

Gracias a este proyecto, el grupo de investigación contará con la información necesaria para introducir mejoras y ofrecer mejores servicios en la detección de arritmias cardiacas, elevando así su capacidad para aportar soluciones a esta problemática tanto a nivel nacional como internacional.

1. Objetivos

Objetivo general

Implementar algoritmos de selección de señales ECG para aprendizaje contrastivo, con el fin de mejorar la precisión en la clasificación de señales con arritmias.

Objetivos específicos

Determinar el modelo de deep learning y la base de datos que satisfaga los requerimientos del problema.

Implementar algoritmos de selección de datos para determinar las señales para el aprendizaje contrastivo.

Entrenar el modelo de deep learning para la clasificación de arritmias cardiacas. Evaluar el desempeño del modelo en cuanto a: Precisión, F1, ROC, y AUC.

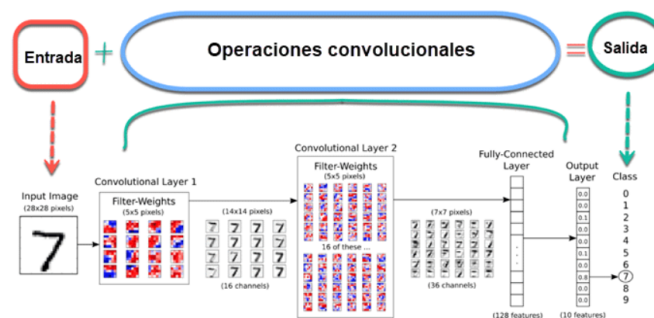
2. Contextualización

2.1. Redes neuronales convolucionales

Las redes neuronales convolucionales (CNNs) son uno de los grandes avances que ha logrado el aprendizaje profundo, también conocido como deep learning, en el campo del reconocimiento de imágenes e identificación de patrones (Shrestha and Mahmood, 2019; Wang et al., 2021). Estas redes tienen una arquitectura típica que se estructura mediante capas de nodos, que incluyen una capa inicial de entrada, una o varias capas ocultas y una capa final de salida. En la figura 1, se puede observar las características comunes de una CNN. Cada nodo establece conexiones con otros nodos y cuenta con un umbral y peso asociados, gracias a estas características esenciales las redes neuronales convolucionales pueden desempeñarse de manera efectiva en problemas de reconocimiento e interpretación de imágenes y vídeo, lo cual explica su amplio uso en estos ámbitos debido a su alto rendimiento (Massiris et al., 2018).

Figura 1.

Arquitectura típica de una red neuronal convolucional (CNN).



Nota. La figura muestra la capa de entrada, las capas ocultas y la capa de salida, con su respectivo ejemplo de lo que iría por dentro de cada una de estas capas, dando como ejemplo lo que es la arquitectura típica de una red neuronal convolucional. Tomado de (guru99es, 2020).

El objetivo de una CNN es obtener todas las características presentes en una imagen y utilizarlas posteriormente para detectar o clasificar objetos dentro de la misma. Estas CNNs pueden requerir un alto poder de procesamiento y, para entrenar modelos, se recomienda el uso de unidades de procesamiento gráfico (GPU) (Massiris et al., 2018).

2.2. Aprendizaje contrastivo

El aprendizaje contrastivo (CL) es una técnica de aprendizaje auto-supervisado que se utiliza para obtener representaciones de calidad a partir de datos no etiquetados. Esta técnica aprovecha la disponibilidad de este tipo de datos para mejorar el rendimiento de los modelos de aprendizaje profundo, especialmente en CNNs. El aprendizaje contrastivo se utiliza como una forma de pre-entrenamiento, permitiendo a la CNN aprender características útiles antes de realizar tareas específicas (Fonseca et al., 2022). Gracias a esto el aprendizaje contrastivo es de gran utilidad en el campo de la salud, ya que, esta industria genera una gran cantidad de datos fisiológicos no etiquetados que pueden ser aprovechados por esta técnica, como pueden ser, las señales ECG.

El objetivo del aprendizaje contrastivo es capacitar un modelo para distinguir entre muestras positivas y negativas, buscando maximizar la similitud entre los pares de muestras positivas y minimizar la similitud entre los pares de muestras negativas (peñarrubia Morcillo, 2023).

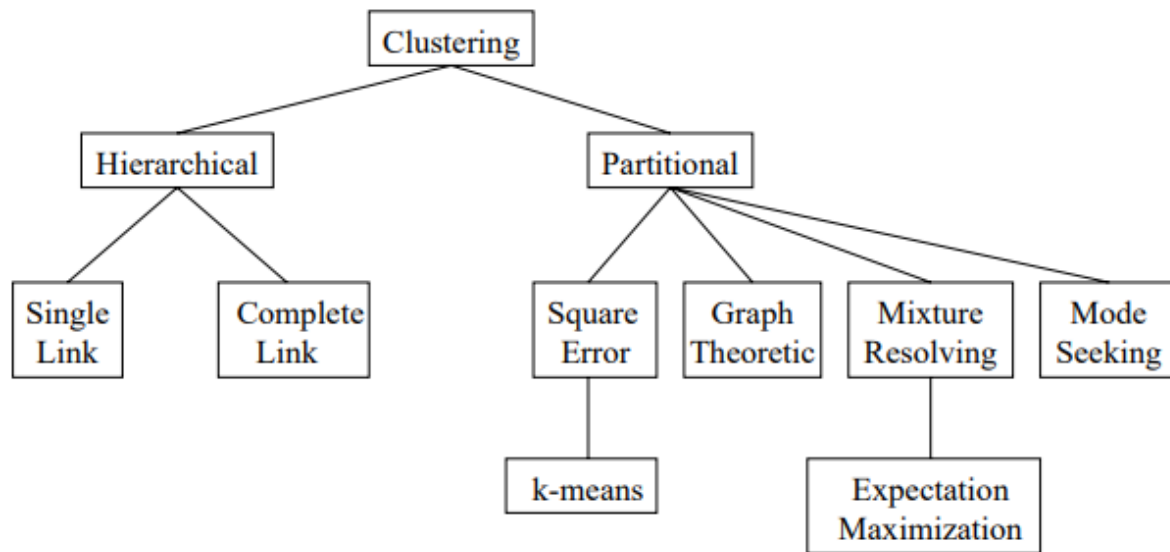
2.3. Clustering

El clustering consiste en agrupar objetos en grupos con características similares, un clúster es un conjunto de muestras de datos que son similares a otros dentro del mismo clúster y son distintos a las muestras de otros clústeres. El clustering es un ejemplo de aprendizaje no supervisado, ya que, este conjunto de muestras dentro del clúster no tiene etiquetas, si estas muestras estuvieran

etiquetadas, dejaría de ser clustering y se convertiría en clasificación. Actualmente, el análisis de clustering juega un rol muy importante en una gran variedad de áreas como en el procesamiento de imágenes, cómputo y multimedia, reconocimiento de patrones, análisis médico, entre otras. Gracias a esto el análisis de clustering se considera como una de las mejores técnicas a la hora de realizar exploraciones en los datos y obtener conocimiento (Farías et al., 2008). Mediante esta técnica, se puede identificar dentro de una región densa y esparcida patrones de distribución global y correlaciones entre patrones de datos (Farías et al., 2008). En la figura 2, se puede observar una clasificación de los enfoques de agrupación que se derivan del clustering.

Figura 2.

Una taxonomía de enfoques de agrupación.



Nota. La figura muestra los dos métodos principales de clustering, que son los métodos jerárquicos y los métodos de partición, con sus respectivos ejemplos de algoritmos que hacen parte de cada uno. Tomado de (Jain et al., 1999).

2.4. k-means

El algoritmo k-means es un método de agrupación particional que consiste en agrupar los datos intentando separar las muestras en n grupos de igual varianza, con el fin de minimizar un criterio conocido como la inercia el cual puede reconocerse como una medida de la coherencia interna de los grupos. (Scikit-Learn, s.f.). El algoritmo k-means divide un conjunto de muestras en grupos disjuntos, es decir, que no compartan ningún elemento en común, cada uno descrito por la media de las muestras en el clúster, los medios se denominan “centroide” del clúster (Scikit-Learn, s.f.).

El algoritmo de k-means suele conocerse como el algoritmo de Lloyd. Básicamente, el algoritmo tiene tres pasos, los cuales son (Scikit-Learn, s.f.) (Jain et al., 1999):

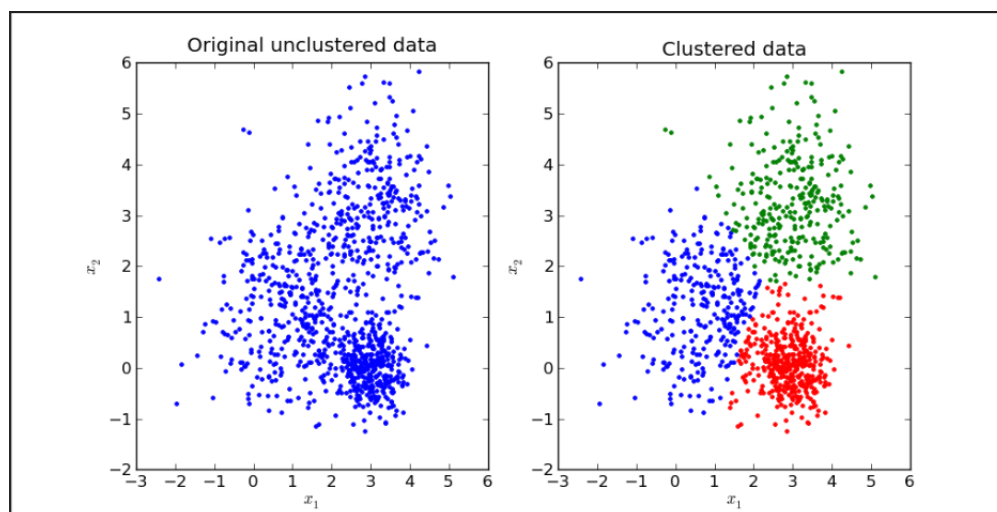
1. Se eligen n centros de clúster para que coincidan con n puntos definidos al azar dentro del hipervolumen que contiene el conjunto de datos.
2. Se asigna cada muestra a su centroide más cercano.
3. Se vuelve a calcular los centroides tomando el valor medio de todas las muestras asignadas a cada centroide anterior, creando así nuevos centroides. Se calcula la diferencia entre los viejos y nuevos centroides y el algoritmo repite estos dos últimos pasos hasta que este valor sea inferior a un umbral preespecificado, básicamente se repite hasta que los centroides no se muevan significativamente.

Una de las principales limitaciones del algoritmo k-means es la necesidad de calcularlo varias veces para obtener el resultado óptimo, debido a que se requiere inicializar el número de

centroides al comienzo de el algoritmo, lo cual afecta la eficacia de este mismo. En la práctica, no se conoce de antemano el número final de clústeres, lo cual dificulta aún más su desempeño (Cambroner and Moreno, 2006). Según (Scikit-Learn, s.f.) un método para ayudar a resolver este problema es el esquema de inicialización k-means++, que ellos han implementado, esto lo que hace es establecer una inicialización de los centroides de tal manera que estén alejados entre si, lo que conduce a resultados posiblemente mejores en comparación con una inicialización aleatoria.

Figura 3.

Resultado de la aplicación de k-means a un conjunto de datos no agrupados.



Nota. La figura ilustra en la parte izquierda un conjunto de datos que no han sido agrupados y en la parte derecha el resultado de el conjunto de datos anterior al aplicarles el algoritmo k-means, que como se puede observar, se realiza una división de los datos en tres clústeres, cada uno representado por un color. Tomado de (Landman et al., s.f.).

3. Revisión de la literatura

3.1. Trabajos relacionados

La electrocardiografía es la técnica de oro para la monitorización cardiaca (Fonseca et al., 2022). Los latidos del corazón se pueden analizar mediante ECGs, estos datos son importantes para detectar y clasificar muchas enfermedades del corazón como las arritmias cardíacas. En aplicaciones para el sector de la salud, la adquisición de datos etiquetados es desafiante y costosa.

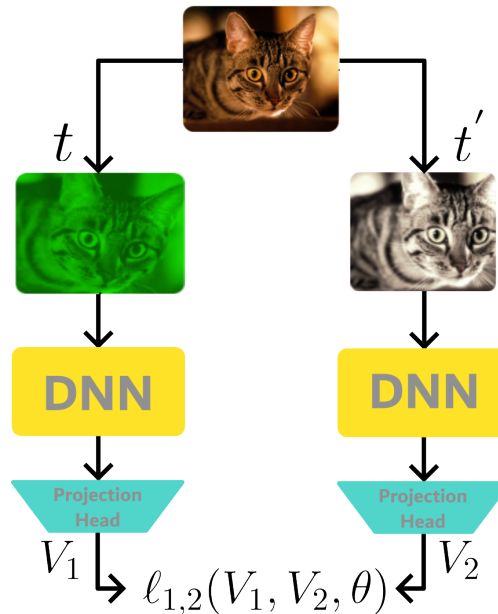
El aprendizaje supervisado y no supervisado han sido de las técnicas emergentes para tratar con esta problemática. Las técnicas de deep learning aplicadas a señales ECG han permitido el desarrollo de modelos capaces de detectar y clasificar arritmias cardíacas, sin embargo, estas técnicas requieren de una cantidad significativa de datos etiquetados. Por otro lado, el aprendizaje auto-supervisado también ha demostrado ser una técnica de aprendizaje que consiste en encontrar buenas representaciones de datos no etiquetados, aprovechando el fácil acceso a este tipo de datos. Un ejemplo de ello es el CL (Chen et al., 2020; Kiyasseh et al., 2021), es un método de aprendizaje auto-supervisado que tiene como objetivo aprender representaciones útiles de instancias que comparten algún contexto (Fonseca et al., 2022). Este método se compone generalmente de tres etapas: el aumento de datos, una estructura de red de aprendizaje no lineal y una función de pérdida contrastiva. Una serie de transformaciones produce múltiples vistas ampliadas de la misma instancia, denominadas pares positivos. En cambio, las vistas de instancias diferentes se definen como pares negativos. El modelo CL consta de dos partes: el codificador y la capa de proyección. En la fase de pre-entrenamiento, el codificador es una red convolucional que aprende a extraer representacio-

nes de los datos de entrada. A continuación, la capa de proyección procesa estas representaciones aplicando una pérdida contrastiva. Chen et al. (2020) sugieren añadir estas capas ayuda al modelo aprender información valiosa sobre los datos. Por último, la función de pérdida contrastiva aprende que las salidas de las capas de proyección son similares para los pares positivos y disímiles para los pares negativos. En el sector médico, grabaciones fisiológicas de un mismo paciente recolectadas en una escala pequeña de tiempo y escenarios similares son más probables de compartir un contexto, por lo tanto los pares positivos de grabaciones ECG pueden obtenerse aplicando divisiones en el tiempo y añadiendo ruido a la misma señal (Fonseca et al., 2022).

La figura 4 muestra el procedimiento general de un algoritmo de aprendizaje contrastivo. El proceso de aprendizaje de este método consiste en la semejanza entre dos entradas transformadas (t y t') de la misma instancia por el proceso de aumento de datos. El modelo debe predecir si t y t' provienen de los mismos datos (pares positivos) o (pares negativos), por lo que solo se requieren los datos por sí mismos sin ninguna etiqueta. Los autores del CL comentan sobre la importancia del aumento de datos múltiples para obtener una representación eficiente. Recientemente, Kiyasseh et al. (2021) propone CLOCS, la implementación de una familia de contrastes métodos de aprendizaje aplicados a las señales ECG que funcionan a través de espacio, tiempo y pacientes haciendo la representación similar a el uno al otro. Además, Diamant et al. (2021) propone otro tipo de CL aplicado a señales ECG. Ellos presentan un enfoque de pre-entrenamiento conocido como Patient Contrastive Learning of Representations (PCLR). Los pares positivos se definen a partir de muestras pertenecientes al mismo paciente y recogidas en diferentes lapsos de tiempo (Fonseca et al., 2022).

Figura 4.

Etapas de un algoritmo de aprendizaje contrastivo: aumento de datos, extracción de características y pérdida contrastiva.



Nota. Se comienza con un dato y se realizan técnicas de aumento de datos para obtener dos nuevas instancias de la misma clase del dato de entrada. Se extraen las características y cada vector de rasgos correspondiente a sus respectivas entradas se introduce en una función de pérdida contrastiva para la actualización de los pesos del modelo CNN. Tomado de (Navarro et al., 2022).

Navarro et al. (2022) propuso implementar una etapa anterior a CL, que busca mejorar el rendimiento de una CNN para la clasificación de fibrilación auricular, la CNN implementada fue encontrada en Fonseca et al. (2022) la cual tuvo un rendimiento prometedor en escenarios desafiantes. Esta etapa consiste en un algoritmo de agrupamiento manual o selección de datos propuesto en Sanchez et al. (2022), esto para seleccionar las muestras mas importantes de un conjunto de datos sin etiquetas, con tal de tener un modelo pre-entrenado óptimo. El resultado de este experimento sugirió que su enfoque superaba la línea de base de CL, obteniendo una solución prometedora en modelos pequeños ya que la eficacia del modelo disminuye a medida que el modelo

aumenta de tamaño.

Clustering o agrupación es el proceso de poner en grupos objetos con características similares, esta es una de las tareas fundamentales que tiene la inteligencia artificial. En el pasado ha surgido una gran familia de algoritmos de clustering desarrollada exitosamente y aplicadas en enormes tareas del mundo real (Nezamabadi et al., 2022).

En clustering hay una separación de los métodos de agrupación, teniendo en cuenta a la agrupación basada en similitudes y la agrupación basada en las características. La primera, construye modelos sobre una matriz de distancia la cual tiene dimensiones $N \times N$, esta mide la distancia entre cada par de N muestras. De los métodos basados en la similitud más famosos es el spectral clustering o agrupamiento espectral (SC) (von Luxburg, s.f.), la cual aprovecha los espectros laplacianos de la matriz para reducir la dimensionalidad antes del agrupamiento. Una de las ventajas que trae utilizar métodos de similitud es la similitud específica del dominio, las funciones se pueden implementar fácilmente en otros modelos, sin embargo, trae una desventaja, la cual es que sufre de problemas de escalabilidad debido al tiempo de ejecución súper-cuadrático para el espectro computacional. La agrupación basada en las características, toman una matriz $N \times D$, como entrada, donde N es el numero de muestras y D la dimensión de la característica. Varios métodos populares basados en esto son: k-means y Gaussian mixture model (GMM). Una ventaja de GMM sobre k-means es que este puede generar muestras por estimación de la densidad de datos. Sin embargo, k-means es fácil de implementar, se adapta fácilmente a nuevos modelos y es lineal en el tamaño de muestras ECG (Nezamabadi et al., 2022).

K-means, se ha utilizado ampliamente solo o junto con un pre-procesamiento adecuado,

normalmente se usa para agrupar datos de muestras que se distribuyen uniformemente alrededor de centroides, pero los datos en la vida real no exhiben esa estructura "k-means friendly" (Yang et al., 2017). En los últimos años, las redes neuronales profundas (DNN) han tenido un éxito en aplicaciones de aprendizaje supervisado y no supervisado, los enfoques de aprendizaje profundo ahora se usan ampliamente para reducción de dimensionalidad (DR) antes de hacer clustering. Por ejemplo, el codificador automático apilado (SAE) (Vincent et al., 2010), CCA profunda (DCCA) (Andrew et al., 2013), y autocodificador escaso (Ng et al., 2001) toman ideas de PCA, CCA y codificación escasa, respectivamente, y hacen uso de DNNs para aprender mapeos no lineales del dominio de datos a espacios latentes de baja dimensión. En estos trabajos, usan sus DNNs como etapas de pre-procesamiento realizada por separado de la etapa de clustering (Yang et al., 2017).

En Yang et al. (2017), proponen un criterio de optimización, en donde juntan k-means y DR, implementando al DR mediante el aprendizaje de un DNN, en lugar de un modelo lineal. Este criterio tiene tres partes: Reducción de dimensionalidad, reconstrucción de datos y regularización de la promoción de la estructura del clúster, implementando una red de decodificación crucial para evitar soluciones triviales. El criterio también es flexible: se puede extender a incorporar diferentes estructuras DNN (por ejemplo, CNNs (LeCun et al., 1998)) y criterios de agrupamiento, p. ej., agrupamiento subespacial, como lo vamos a observar también por medio de este proyecto.

4. Metodología

4.1. Icentia11k

Icentia11k (Tan et al., 2019) es una base de datos ECG pública que abarca una gran cantidad de señales sin procesar continuas para el aprendizaje de representaciones. Esta base de datos

está compuesta por señales obtenidas de 11.000 pacientes, mediante el uso de un dispositivo de monitorización cardíaca de una sola punta denominado CartioSTAT™. Las señales fueron registradas con una precisión de 16 bits y se tomaron muestras a una frecuencia de 250Hz, como se puede observar en la tabla 1. Los datos fueron recolectados de pacientes principalmente ubicados en Ontario, Canadá, provenientes de diversos centros médicos, con una edad promedio de 62.2 ± 17.4 años.

Tabla 1

Estadísticas del conjunto de datos

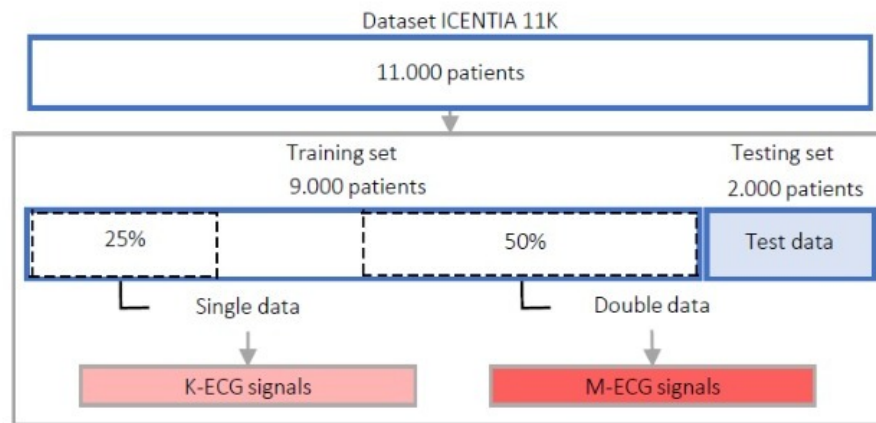
Statistic	# (units)
Number of Patients	11,000
Number of labeled beats	2,774,054,987
Sample Rate	250Hz
Frame size	2,049 samples
Segment size	1,048,577 samples
Total number of frames	1,084,314

Nota. Esta tabla muestra los detalles de la base de datos Icentia11k. Adaptada de (Tan et al., 2019).

El uso principal del dispositivo CardioSTAT™ es para llevar a cabo exámenes de tercera línea, el cual busca proporcionar una evaluación más completa y precisa de las arritmias, lo que implica que la mayoría de los registros en la base de datos exhiben arritmias. La clasificación de la base de datos se realiza de acuerdo con el tipo de ritmo: ritmo sinusal normal (NSR), fibrilación auricular (AFib), aleteo auricular (AF) y ruido. En este proyecto el 56% de los datos están etiquetados como NSR, el 4% están etiquetadas como arritmias (AFib y AF) y el 40% corresponden a ruido en la base datos, lo que resulta en un desequilibrio en esta. Además, los datos se someten a

un preprocesamiento según lo indicado en Fonseca et al. (2022). Para garantizar la diversidad de los datos utilizados en cada etapa, aseguramos que tanto los datos simples como los datos dobles se obtengan de conjuntos de datos de entrenamiento separados. Esto se realiza para evitar la utilización de los mismos pacientes en cada etapa, como se muestra en la Figura 5 (Fonseca et al., 2022).

Figura 5.
Distribución del dataset



Nota. Esta figura muestra como se dividió la base de datos en dos grupos con diferentes porcentajes de muestras, llamados entrenamiento y pruebas. Del conjunto de entrenamiento, se extraen los datos singulares y dobles. Tomado de (Fonseca et al., 2022).

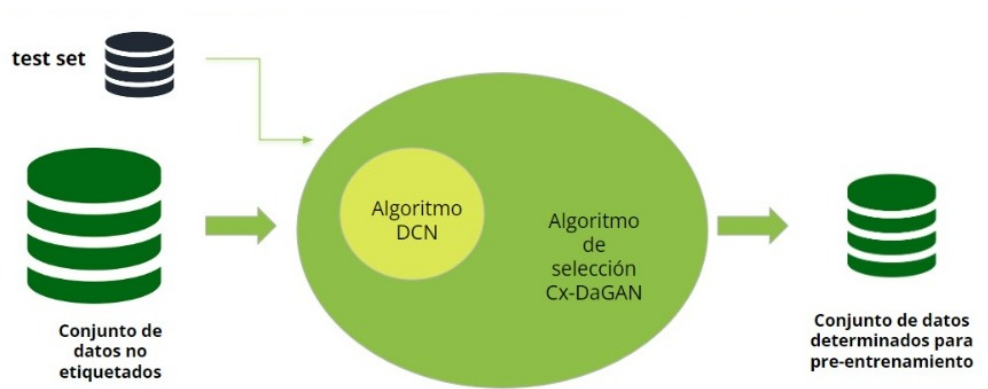
4.2. Selección de datos

Para la etapa de selección de datos se implementaron dos algoritmos: DCN (Yang et al. (2017)) y CX- Dagan (Sanchez et al. (2022))

4.2.1. Algoritmo DCN:. La mayoría de los enfoques de aprendizaje por lo general tratan la DR y el agrupamiento de manera separada. Sin embargo, investigaciones recientes han

Figura 6.

Estructura etapa de selección de datos



Nota. Esta figura muestra la estructura que tiene la etapa de selección de datos, en donde se implementan dos algoritmos para seleccionar los datos adecuados para la etapa de pre-entrenamiento. Creado por los autores.

demostrado que optimizar ambas tareas de forma conjunta puede mejorar significativamente su rendimiento. Esta nueva aproximación considera que las muestras de datos se obtienen a través de una transformación lineal de representaciones latentes que son fáciles de agrupar, aunque en la práctica esta transformación puede ser más compleja.

En este trabajo, se aborda el desafío de optimizar conjuntamente la reducción de dimensionalidad (DR) y el agrupamiento de datos mediante una aproximación basada en aprendizaje profundo. A diferencia de enfoques tradicionales, se considera una transformación no lineal y desconocida para obtener representaciones latentes que sean adecuadas para el agrupamiento, tal como se propone en (Yang et al., 2017). En dicho trabajo, Yang et al. presentaron un enfoque conjunto de DR y k-means, en el cual la DR se logra mediante el aprendizaje de una Red Neuronal Profunda (DNN), aprovechando la capacidad de la red neuronal de aproximar funciones no lineales.

El algoritmo de selección de datos se basa en el criterio de optimización propuesto por

(Yang et al., 2017), que consta de tres partes: reducción de dimensionalidad, reconstrucción de datos y regularización que promueve la estructura de los grupos. Para evitar soluciones triviales en la DR lineal, se utiliza una función de costo (7) que adopta la conocida función de pérdida por mínimos cuadrados. Además, se introduce una restricción en el espacio latente para garantizar que cada muestra x_i esté asociada a un solo grupo, como se expresa en (Yang et al., 2017)

Figura 7.

Fórmula de función de costo

$$\min_{W, Z, M, \{s_i\}} \sum_{i=1}^N \left(\ell(g(f(x_i)), x_i) + \frac{\lambda}{2} \|f(x_i) - Ms_i\|_2^2 \right)$$

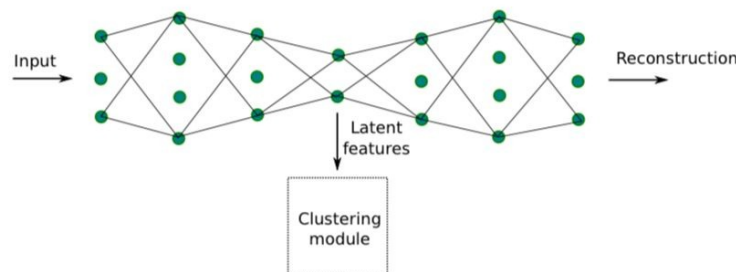
Nota. Esta figura muestra como se expresa la función de costo, la cual adopta la famosa función de pérdida por mínimos cuadrados. Donde:

$$s.t. s_j, i \in 0, 1, 1^T s_i = 1 \forall i, j \quad (1)$$

Tomado de (Yang et al., 2017).

Figura 8.

Arquitectura DCN



Nota. Esta figura muestra la estructura de deep clustering network. Tomado de (Yang et al., 2017).

La arquitectura de la red utilizada para este enfoque es conocida como Deep Clustering Network (DCN), cuya estructura se presenta en la Figura 8. Esta red incluye capas de codificación

y decodificación que transforman los datos en bruto a un espacio de baja dimensión y luego los reconstruyen a partir del espacio latente. La tarea de k-means se realiza en la capa bottleneck, la cual une ambas redes.

4.2.2. Algoritmo CX-dagan: . Este algoritmo, es un modelo de aprendizaje profundo diseñado en (Sanchez et al., 2022) , el cual presenta el enfoque CX-DaGAN para abordar el problema de adaptación de dominio en el diagnóstico de neumonía a partir de imágenes de radiografías de tórax. El objetivo de este enfoque es mejorar la precisión y el rendimiento del diagnóstico al adaptar un modelo entrenado en un conjunto de datos de entrenamiento (dominio fuente) a un conjunto de prueba diferente (dominio objetivo).

El algoritmo de selección de datos en CX-DaGAN consta de tres etapas principales:

- Normalización y Análisis de Componentes Principales (PCA): Se normalizan los datos en el conjunto de prueba (dominio objetivo) para tener media cero y desviación estándar unitaria. Luego, se aplica el Análisis de Componentes Principales (PCA) para seleccionar las principales características y reducir la dimensionalidad de los datos.
- Proyecciones y Funciones de Similitud: Se proyectan las imágenes de los conjuntos en los subespacios correspondientes del conjunto de prueba (T). Se calculan las funciones de similitud que miden el error de proyección de cada dato en los subespacios de características.
- Proyecciones y Funciones de Similitud: Se proyectan los datos en los subespacios correspondientes del conjunto de prueba (T). Se calculan las funciones de similitud que miden el error de proyección de cada dato en los subespacios de características.

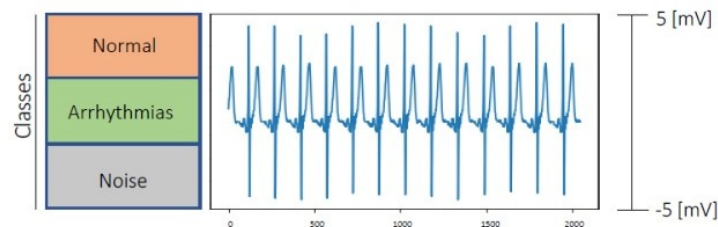
A partir de las proyecciones, se construyen vectores de error que contienen los errores de proyección de los datos en ambos subespacios. Luego, se seleccionan las imágenes con los menores errores de proyección dentro de cada clase para formar el conjunto de datos adaptado.

El objetivo del algoritmo de selección de datos es identificar las imágenes más similares entre el conjunto de entrenamiento (dominio fuente) y el conjunto de prueba (dominio objetivo). Esto se logra al seleccionar las imágenes que tienen menores errores de proyección en los subespacios de características.

En este proyecto, se encarga también, de llamar al algoritmo de DCN, seleccionando que datos son los que más se parecen a un conjunto de datos elegidos anteriormente llamado test set o test de prueba, formando así el conjunto final de datos seleccionados para la etapa de pre-entrenamiento con aprendizaje contrastivo.

4.3. Pre-entrenamiento

Figura 9. *Distribución del set de datos*



Nota. La figura muestra las diferentes clases usadas para la clasificación, las cuales son: normal, arritmia y ruido. También muestra cómo se delimita el rango de la señal a 5mV. Tomado de (Fonseca et al., 2022).

La etapa de pre-entrenamiento implica aplicar CL a las señales seleccionadas durante la

etapa de selección de datos. CL busca aprender representaciones útiles de datos al comparar pares de ejemplos similares y diferentes, sin depender de etiquetas predefinidas. Se maximiza la similitud entre pares de ejemplos con características similares y se minimiza la similitud entre pares con características diferentes. Al utilizar el aumento de datos, el modelo debe predecir si las entradas transformadas de una misma instancia pertenecen a la misma clase, evitando la necesidad de etiquetas.

Los escenarios de aprendizaje profundo que requieren una gran cantidad de datos pueden beneficiarse del aumento de datos (DA) (Fajardo and Roy, 2021). En el campo médico, circunstancias comparables y plazos cortos pueden compartirse mediante registros fisiológicos del mismo paciente que se obtuvieron (Chen et al., 2020). La misma señal puede transformarse para generar conjuntos positivos de registros de ECG.

El modelo de CL implementado en este proyecto se basa de dos partes: Un codificador (encoder) y proyección de capas. Durante el pre-entrenamiento, el encoder es una CNN que aprende a extraer representaciones de los datos de entrada; el encoder utilizado es el propuesto en (Fajardo and Roy, 2021), este modelo permite cambiar el tamaño del modelo, modificando el tamaño del filtro convolucional inicial.

La capa de proyección procesa esas representaciones, se construye una capa de proyección (Tan et al., 2019), con un tamaño de entrada dado por el largo de la salida del encoder, esa red neuronal consiste en 2 capas densas con un numero correspondiente de unidades (width). Las capas finales están compuestas por capas densas con 128 neuronas (dense width) y una capa densa con 3 neuronas, usadas para la clasificación. El conjunto de transformaciones genera múltiples vistas

aumentadas de la misma instancia, llamados pares positivos, por otro lado, las vistas aumentadas de diferente instancia se llamarán pares negativos. Por último, la función de pérdida contrastiva aprende las salidas de las capas de proyección para que estas sean similares a los pares positivos y disimilares a los pares negativos.

En resumen, se hace el pre-entrenamiento del encoder utilizando CL. Luego, se aplica una tarea de clasificación con el fin de evaluar la inicialización introducida por el pre entrenamiento.

Realizando los siguientes pasos:

- Se toman las señales M de los datos dobles no etiquetados, esto con el fin de construir batches (lotes) contrastivos de un tamaño específico.
- Se utilizan los batches contruidos para el pre-entrenamiento del encoder con las capas de proyección. De (Tan et al., 2019) se toma la función de pérdida la cual va a maximizar la similitud entre las proyecciones del par positivo. La función de pérdida de un par positivo i , j es la siguiente:

$$l_{i,j} = -\log \left(\frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{s=1}^{2N, s \neq i} \exp(\text{sim}(z_i, z_s)/\tau)} \right) \quad (2)$$

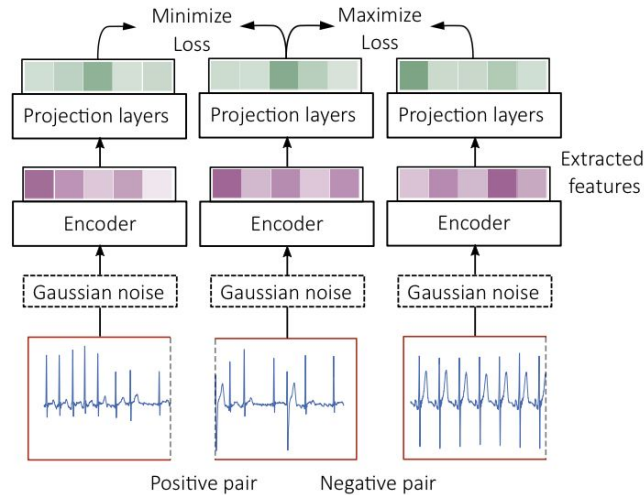
Donde, sim corresponde a la similitud del coseno, z indica las salidas de las capas de proyección, N se refiere al tamaño del lote contrastivo.

4.4. Entrenamiento

Durante el proceso de entrenamiento, se emplea la función de pérdida de entropía cruzada categórica dispersa y el optimizador Adam con su configuración por defecto. Se toma el codificador

Figura 10.

Arquitectura de aprendizaje contrastivo aplicado en señales ECG



Nota. La figura muestra las representaciones obtenidas al pasar la señal ECG singular por medio del encoder y las capas de proyección, obteniendo al final que las representaciones para pares positivos se fomentan a ser similares entre si y no similares a las otras representaciones de las demás señales. Tomado de (Fonseca et al., 2022).

pre-entrenado y se agrega una capa de sondeo lineal al finalizar la fase de pre-entrenamiento. Por último, se utiliza un conjunto determinado de señales para entrenar el codificador. Las métricas de clasificación incluyen el puntaje F1, el área bajo la curva (AUC), la pérdida y la precisión. Durante el entrenamiento, se utiliza early stopping y una tasa de aprendizaje de 0.01, la cual se reduce en un factor de diez cuando la precisión de validación deja de mejorar después de cuatro épocas.

5. Resultados

En esta sección, presentaremos las condiciones de entrenamiento y los experimentos realizados para validar nuestro método.

Se entrenan los modelos mencionados en la sección 4.4 y se calculan la puntuación F1 y

el AUC en el test set. Además, se calcula la media y la desviación estándar en el test set. Todas las simulaciones se ejecutaron en GPUs Nvidia V100 SXM2. Para las fases de pre-entrenamiento y entrenamiento, se estableció un tamaño de batch de 256 señales ECG. La cantidad de datos sin etiquetar utilizados para la etapa de pre-entrenamiento osciló entre 0, 5k, 10k, 15, 20k, 25k y 50k.

Se realizan los experimentos utilizando dos procedimientos de selección de datos diferentes: Selección Aleatoria (RS, por sus siglas en inglés) y Selección con Restricción de Similitud (SCS, por sus siglas en inglés). Ambos procedimientos implicaron la selección de K señales sin etiquetar de longitud doble y ambos fueron seguidos por el algoritmo de Aprendizaje Contrastivo. Además, también se comparan nuestros resultados con la Inicialización Aleatoria (RI, por sus siglas en inglés), que no implica ningún proceso de pre-entrenamiento.

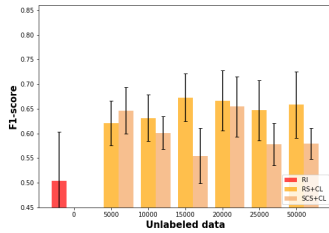
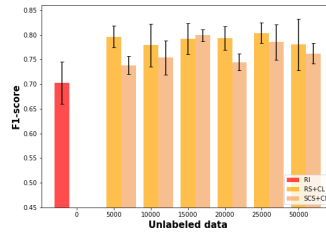
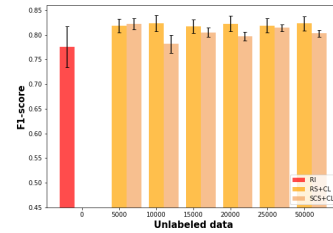
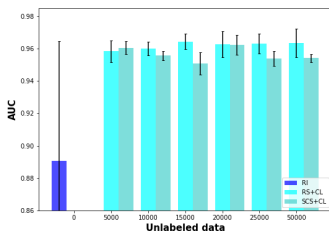
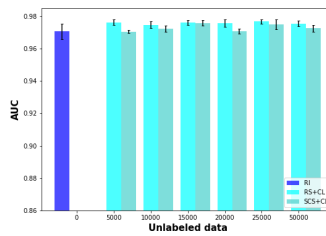
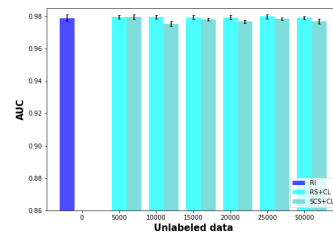
El proceso de entrenamiento se repite diez veces y se promedian los resultados de puntuación F1 y AUC obtenidos. La Figura 11 muestra la cantidad de datos no etiquetados utilizados en esta etapa de pre-entrenamiento. Se evalúa cada caso en un modelo pequeño que se denomina M1 (66k parámetros) y en un modelo de tamaño más grande llamado M2 (250k parámetros). Las tablas 2 y 3 muestran el promedio de las métricas para cada caso y la diferencia con respecto a cada procedimiento de selección de datos utilizado.

5.1. Rendimiento del modelo

Tras analizar los resultados obtenidos, se observa que aunque no se evidencia una mejora significativa en las métricas, sí se registra una reducción en la desviación estándar de casi un punto. Esta estabilidad se atribuye a la capacidad del enfoque utilizado para seleccionar los datos más adecuados durante el pre-entrenamiento. Estos hallazgos sugieren que dicho enfoque puede

Figura 11.

Valores $F1$ y AUC alcanzados con el modelo $M1$ (66k parámetros) en el test set utilizando 5k, 10k y 15k datos etiquetados.

(a) $F1$ -score, 5k labeled data**(b)** $F1$ -score, 10k labeled data**(c)** $F1$ -score, 15k labeled data**(d)** AUC , 5k labeled data**(e)** AUC , 10k labeled data**(f)** AUC , 15k labeled data

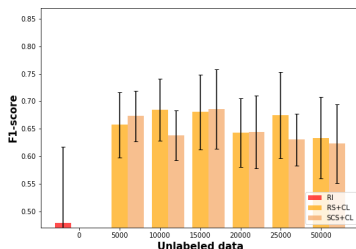
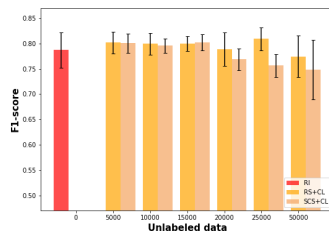
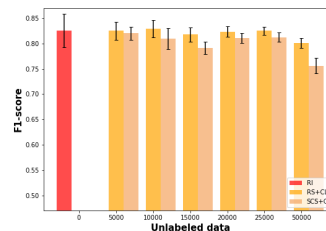
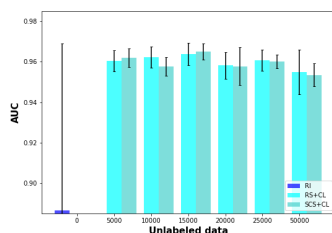
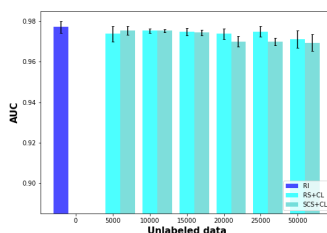
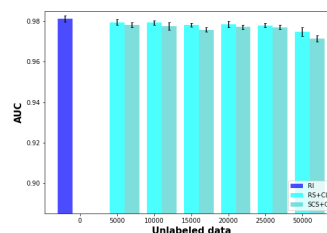
Nota. Cada subfigura tiene tres etiquetas las cuales describen los pasos tomados en el pre-entrenamiento: Selección Aleatoria (RS) y Selección con Restricción de Similitud (SCS) e Inicialización Aleatoria (RI). Creado por los autores.

resultar beneficioso para mejorar el entrenamiento de modelos pequeños, como se evidencia en los datos de 5K, al proporcionar una mejor inicialización.

No obstante, se debe tener en cuenta una limitación potencial de este trabajo la cual es que todos los datos utilizados en los experimentos pertenecían al mismo conjunto de datos recopilados de un tipo específico de dispositivo portátil. Esto podría introducir sesgos que limiten la generalización de los hallazgos a otros dispositivos y conjuntos de datos. Para abordar esta limitación, se recomienda que futuros trabajos prueben el enfoque en un conjunto más amplio y diverso de fuentes y tipos de dispositivos. En general, el enfoque muestra promesa para mejorar el entrenamiento

Figura 12.

Valores $F1$ y AUC alcanzados con el modelo $M2$ (250k parámetros) en el test set utilizando 5k, 10k y 15k datos etiquetados.

(a) F1-score, 5k labeled data**(b)** F1-score, 10k labeled data**(c)** F1-score, 15k labeled data**(d)** AUC, 5k labeled data**(e)** AUC, 10k labeled data**(f)** AUC, 15k labeled data

Nota. Cada subfigura tiene tres etiquetas las cuales describen los pasos tomados en el pre-entrenamiento: Selección Aleatoria (RS) y Selección con Restricción de Similitud (SCS) e Inicialización Aleatoria (RI). Creado por los autores.

de modelos pequeños con datos etiquetados limitados, y se alentaría a realizar investigaciones adicionales para explorar su aplicabilidad en otros dominios y conjuntos de datos.

La relación entre el tamaño del modelo, la cantidad de datos etiquetados y el rendimiento del método propuesto destaca la importancia de considerar estos factores al aplicar el enfoque en diferentes entornos.

Tabla 2

Valores promedio del puntaje F1 en el test set Icentia11k.

Modelo	Método de Inicialización	Datos etiquetados		
		5k	10k	15k
M1 (66k parám.)	RI	50.4 ± 9.8	70.3 ± 4.3	77.6 ± 4.1
	RS+CL (Fonseca et al., 2022)	64.9 ± 5.5	79.1 ± 3.2	82.1 ± 1.5
	SCS (DCN+CxDaGAN) + CL	60.2 ± 4.5	76.4 ± 2.3	80.4 ± 1.0
	Δ_1	9.8	6.1	2.8
	Δ_2	-4.7	-2.7	-1.7
M2 (250k parám.)	RI	47.9 ± 14.0	78.7 ± 3.5	82.5 ± 3.3
	RS+CL (Fonseca et al., 2022)	66.2 ± 6.6	79.6 ± 2.6	82.0 ± 1.3
	SCS (DCN+CxDaGAN) + CL	64.9 ± 5.8	77.9 ± 2.5	80.0 ± 1.3
	Δ_1	17.0	-0.9	-2.0
	Δ_2	-1.3	-1.7	-2.5

Nota. Puntuaciones promedio de F1 en el test set Icentia11k (Tan et al., 2019), utilizando diferentes datos para la etapa de selección de datos. Δ_1 y Δ_2 son las diferencias entre nuestro método en comparación con RI y RS+CL, respectivamente. Creado por los autores.

Tabla 3*Promedio del AUC en el test set Icentia11k.*

Modelo	Método de Inicialización	Datos etiquetados		
		5k	10k	15k
M1 (66k parám.)	RI	89.1 ± 7.4	97.1 ± 0.5	97.9 ± 0.2
	RS+CL (Fonseca et al., 2022)	96.2 ± 0.6	97.6 ± 0.2	97.9 ± 0.1
	SCS (DCN+CxDaGAN) + CL	95.6 ± 0.4	97.3 ± 0.2	97.8 ± 0.1
	Δ_1	6.6	0.2	-0.1
	Δ_2	-0.6	-0.3	-0.2
M2 (250k parám.)	RI	88.6 ± 8.3	97.7 ± 0.3	98.1 ± 0.2
	RS+CL (Fonseca et al., 2022)	96.0 ± 0.7	97.4 ± 0.3	97.8 ± 0.1
	SCS (DCN+CxDaGAN) + CL	95.9 ± 0.5	97.2 ± 0.2	97.6 ± 0.1
	Δ_1	7.3	-0.5	-0.5
	Δ_2	-0.1	-0.2	-0.2

Nota. Promedio del AUC en el test set Icentia11k (Tan et al., 2019), utilizando diferentes datos para la etapa de selección de datos. Δ_1 y Δ_2 son las diferencias entre nuestro método en comparación con RI y RS+CL, respectivamente. Creado por los autores.

6. Conclusiones

A partir de los desarrollos presentados y los resultados obtenidos en el presente trabajo de grado, es posible enunciar la siguiente conclusión general: El trabajo se ha centrado en mejorar el aprendizaje contrastivo para la detección de arritmias cardíacas utilizando datos no etiquetados. El enfoque propuesto utilizó un algoritmo de selección de datos con restricciones de similitud para determinar las señales utilizadas en el pre-entrenamiento de un CNN. Los resultados experimentales sugieren que el enfoque propuesto no supera al método de referencia del aprendizaje contrastivo. No obstante, se observó de manera consistente una reducción en la variabilidad de los resultados, lo que indica una mejora en la robustez.

Aunque no se lograron mejoras significativas en comparación con las métricas del método de referencia, los hallazgos proporcionan una base sólida para futuras investigaciones en el campo del aprendizaje contrastivo aplicado a la detección de arritmias cardíacas. La identificación de la reducción en la variabilidad de los resultados es un indicio prometedor para la potencial utilidad del enfoque en situaciones donde la robustez es un factor crítico.

Referencias Bibliográficas

- Andrew, G., Arora, R., Bilmes, J., and Livescu, K. (2013). Deep canonical correlation analysis. In *International conference on machine learning*, pages 1247–1255. PMLR.
- Cambronero, C. G. and Moreno, I. G. (2006). Algoritmos de aprendizaje: knn & kmeans. *Inteligencia en Redes de Comunicación, Universidad Carlos III de Madrid*, 23.
- Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.
- Clinic, C. (2018). Arrhythmia: Types, symptoms, causes, diagnosis treatments. *Cleveland Clinic*. Recuperado 2023, de <https://my.clevelandclinic.org/health/diseases/16749-arrhythmia>.
- de Colombia, M. d. S. y. P. S. (s.f.). Enfermedades cardiovasculares. *Ministerio de Salud y Protección Social de Colombia*. Recuperado 2023, de <https://www.minsalud.gov.co/salud/publica/PENT/Paginas/enfermedades-cardiovasculares.aspx>.
- Diamant, N., Reinertsen, E., Song, S., Aguirre, A., Stultz, C., and Batra, P. (2021). Patient Contrastive Learning: a Performant, Expressive, and Practical Approach to ECG Modeling. *arXiv preprint arXiv:2104.04569*, pages 1–17.
- Fajardo, C. and Roy, K. (2021). A compressed deep learning model to detect cardiac arrhythmias.

In *43rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (Submitted)*.

Farías, R., Durán, E. B., and Figueroa, S. G. (2008). Las técnicas de clustering en la personalización de sistemas de e-learning. In *XIV Congreso Argentino de Ciencias de la Computación*.

Fonseca, K., Osorio, S., Castillo, J., and Fajardo, C. (2022). Contrastive learning for atrial fibrillation detection in challenging scenarios. In *2022 30th European Signal Processing Conference (EUSIPCO)*, pages 1218–1222. IEEE.

guru99es (2020). Clasificación de imágenes de tensorflow: Cnn (red neural convolucional). *Guru99*. Recuperado 2023, de <https://guru99.es/convnet-tensorflow-image-classification/>.

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):1–1.

Khan, M. A., Hashim, M. J., Mustafa, H., Baniyas, M. Y., Al Suwaidi, S. K. B. M., AlKatheeri, R., Alblooshi, F. M. K., Almatrooshi, M. E. A. H., Alzaabi, M. E. H., Al Darmaki, R. S., et al. (2020). Global epidemiology of ischemic heart disease: results from the global burden of disease study. *Cureus*, 12(7).

Kiyasseh, D., Zhu, T., and Clifton, D. A. (2021). Clocs: Contrastive learning of cardiac signals across space, time, and patients. In *International Conference on Machine Learning*, pages 5606–5615. PMLR.

- Landman, N., Pang, H., and Williams, C. (s.f.). K-means clustering. *Brilliant.org*. Recuperado 2023, de <https://brilliant.org/wiki/k-means-clustering/>.
- LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.
- Massiris, M., Delrieux, C., and Fernández Muñoz, J. Á. (2018). Detección de equipos de protección personal mediante red neuronal convolucional yolo. In *XXXIX Jornadas de Automática*, pages 1022–1029. Área de Ingeniería de Sistemas y Automática, Universidad de Extremadura.
- Navarro, A., Fonseca, K., and Fajardo, C. (2022). A similarity constrained signal selection algorithm for contrastive learning in the detection of atrial fibrillation.
- Nezamabadi, K., Sardaripour, N., Haghi, B., and Forouzanfar, M. (2022). Unsupervised ecg analysis: A review. *IEEE Reviews in Biomedical Engineering*, 16:1–1.
- Ng, A., Jordan, M., and Weiss, Y. (2001). On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 14.
- peñarrubia Morcillo, C. (2023). Aprendizaje auto-supervisado para reconocimiento óptico de música. Master's thesis, Universidad de Alicante.
- Sanchez, K., Hinojosa, C., Arguello, H., Kouamé, D., Meyrignac, O., and Basarab, A. (2022). Cx-dagan: Domain adaptation for pneumonia diagnosis on a small chest x-ray dataset. *IEEE Transactions on Medical Imaging*, 41(11):3278–3288.

- Scikit-Learn (s.f.). Clustering. *Scikit-Learn*. Recuperado 2023, de <https://scikit-learn.org/stable/modules/clustering.html>.
- Shrestha, A. and Mahmood, A. (2019). Review of deep learning algorithms and architectures. *IEEE access*, 7:53040–53065.
- Tan, S., Androz, G., Chamseddine, A., Fecteau, P., Courville, A., Bengio, Y., and Cohen, J. P. (2019). Icentia11k: An unsupervised representation learning dataset for arrhythmia subtype discovery. *arXiv preprint arXiv:1910.09570*.
- Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.-A., and Bottou, L. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(12).
- von Luxburg, U. (s.f.). A tutorial on spectral clustering. statistics and computing. *Data Structures and Algorithms (cs. DS); Machine Learning*, pages 395–416.
- Wang, T., Lu, C., Sun, Y., Yang, M., Liu, C., and Ou, C. (2021). Automatic ecg classification using continuous wavelet transform and convolutional neural network. *Entropy*, 23(1).
- Yang, B., Fu, X., Sidiropoulos, N. D., and Hong, M. (2017). Towards k-means-friendly spaces: Simultaneous deep learning and clustering. In *international conference on machine learning*, pages 3861–3870. PMLR.