Caso Práctico:
Métodos de Análisis Topológico para la Clasificación de Arritmias Cardiacas con ECG

Maritza Barrios Macías A00836821

Introducción

Las enfermedades cardiovasculares (ECV) siguen siendo una de las principales causas de muerte en el mundo. En México, en 2021, provocaron cerca de 220 mil fallecimientos, de los cuales 177 mil fueron por infarto al miocardio (Gobierno de México, 2023). La detección temprana mediante señales de electrocardiograma (ECG) es crucial para prevenir estos eventos, pero el desarrollo de sistemas automáticos de clasificación enfrenta retos como la variabilidad fisiológica entre pacientes, la alta dimensionalidad y el ruido en las señales.

En este contexto, el Análisis Topológico de Datos (TDA) se ha propuesto como una alternativa innovadora. Esta técnica permite extraer estructuras geométricas y topológicas robustas de los datos, como componentes conexos y ciclos, mediante herramientas como la homología persistente, Mapper y paisajes de persistencia Hernández-Lemus et al., 2024. Su aplicación en señales ECG ha mostrado ser especialmente útil para identificar patrones asociados a arritmias que los métodos estadísticos tradicionales no capturan fácilmente.

Estudios recientes demuestran que incorporar descriptores topológicos en modelos de aprendizaje automático mejora notablemente la clasificación de latidos anómalos, incluso en presencia de ruido significativo o una distribución desbalanceada entre clases Dindin et al., 2019. Esta robustez y capacidad de generalización hacen del TDA una estrategia prometedora en el análisis computacional de señales cardiacas.

A partir de este contexto, se plantean las siguientes preguntas de investigación:

- ¿Qué impacto tienen las características topológicas extraídas mediante homología persistente en la clasificación de latidos cardíacos frente a métodos tradicionales?
- ¿Qué tan robustos son estos modelos topológicos ante variaciones interpaciente, tipo de latido y ruido?
- ¿Cómo se compara una arquitectura modular que combine TDA, reducción de dimensionalidad y SVM con modelos que no lo incluyan?

Objetivo e Hipótesis

El objetivo principal de este trabajo es realizar un análisis comparativo entre modelos de clasificación de latidos cardíacos, evaluando el impacto de incorporar características extraídas mediante Homología Persistente (TDA) frente a modelos que no las utilizan. Se busca identificar las ventajas y desventajas de integrar TDA en modelos de aprendizaje automático, considerando criterios como precisión, robustez ante variaciones interpaciente y resistencia al ruido presente en las señales ECG. A partir de este objetivo, se plantea la hipótesis de que los modelos de clasificación que integran características topológicas derivadas de la Homología Persistente presentan un desempeño superior respecto a los modelos tradicionales. Esto se debe a su capacidad para capturar patrones estructurales más robustos, menos sensibles al ruido y a las variaciones fisiológicas entre pacientes.

Metodología

La base de datos utilizada en este trabajo fue la MIT-BIH Arrhythmia Database (dividida en conjuntos de entrenamiento y prueba), que contiene un total de 109,446 registros de latidos cardíacos, segmentados a partir de señales de electrocardiograma (ECG) muestreadas a 125 Hz. Cada segmento representa un latido individual, etiquetado según su clase correspondiente. Las anotaciones están distribuidas en cinco clases: N para latidos normales, S para latidos supraventriculares, V para latidos ventriculares, F para latidos de fusión y Q para latidos no clasificables (Fazeli, s.f.).

Para la preparación de los datos, se verificó que todas las señales estuvieran normalizadas en un rango de 0 a 1. Además, se analizó la distribución de frecuencia por clase en el conjunto de entrenamiento, observándose un importante desbalance de clases (fig. 1).

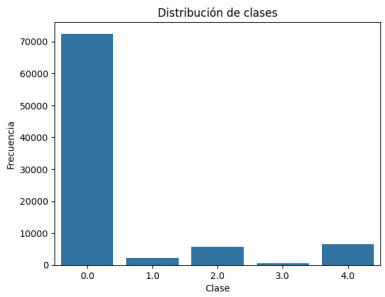

Figura 1: Distribución de frecuencia por clase en el conjunto de entrenamiento.

La clase 0 (latidos normales) tiene 72,470 muestras, la clase 1 (latidos supraventriculares) tiene 2,223 muestras, la clase 2 (latidos ventriculares) tiene 5,788 muestras, la clase 3 (latidos de fusión) tiene 641 muestras, y la clase 4 (latidos no clasificables) tiene 6,431 muestras.

Esta distribución evidencia una fuerte predominancia de la clase 0 (latidos normales), lo que podría influir negativamente en el desempeño del modelo. Por ello, se exploraron diferentes enfoques para abordar el desbalance en las clases, sin embargo, los resultados obtenidos no mostraron una mejora significativa en el rendimiento del modelo. Por esta razón, se optó por mantener el desbalance en los datos.

Posteriormente, se graficaron señales ECG representativas para cada clase con el objetivo de visualizar las diferencias morfológicas entre los distintos tipos de latidos (fig. 2), tomando como base el código de Mollenhauer, 2018. Esto permitió tener una primera aproximación al comportamiento característico de cada categoría, identificando que las clases 0 y 3 son más fáciles de distinguir por su regularidad, mientras que las 1 y 2 podrían requerir técnicas más robustas de extracción de características o aprendizaje automático para ser clasificadas correctamente.

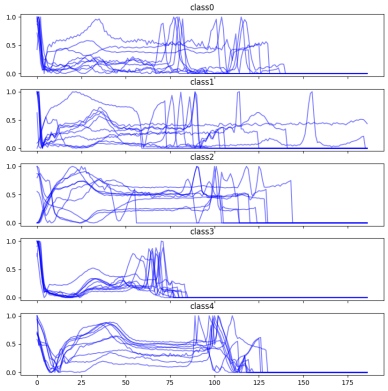

Figura 2: Visualización de series temporales para cada clase con 10 ejemplos representativos

Además, se construyó una matriz de correlación entre las señales con el fin de explorar posibles relaciones lineales entre las clases e identificar similitudes estructurales (Anexo A).

Tras este análisis exploratorio, se realizó un análisis de homología persistente. Para cada clase del conjunto de datos, se seleccionó una señal y se transformó en una nube de puntos en un espacio tridimensional mediante un embebido de retardo temporal, utilizando una dimensión de incrustación de 3 y un retardo de 2. Esta representación permitió analizar la forma geométrica de la señal en un espacio de fase. Luego, se aplicó la técnica de homología persistente mediante el complejo de Rips para extraer características topológicas, visualizando el diagrama de persistencia, el cual muestra la aparición y desaparición de componentes topológicos a lo largo del proceso de filtración (fig.3). Se puede consultar Anexo B para ver los diagramas correspondientes al resto de las clases.

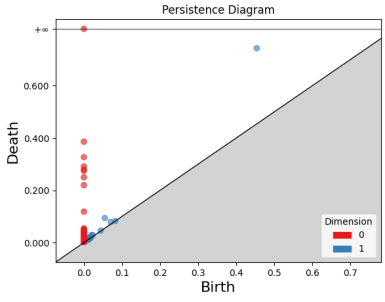

Figura 3: Diagrama de persistencia de la clase 0

Además, se construyó la imagen de persistencia para cada señal (Anexo C), una representación matricial que captura la relevancia topológica de las características extraídas y facilita su uso en modelos de aprendizaje automático. A continuación, se muestra un ejemplo de la clase 0 (fig.4).

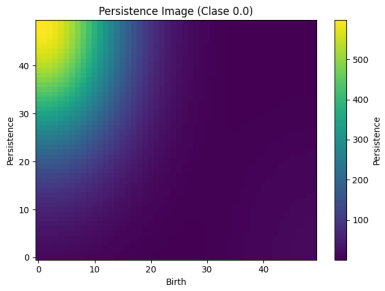

Figura 4: Imagen de persistencia de la clase 0

Finalmente, se graficó el paisaje de persistencia (Anexo D), una función continua que resume la información del diagrama en capas superpuestas, proporcionando una vista alternativa de la estructura topológica de cada clase. Estas visualizaciones permiten comparar la complejidad topológica entre las clases de latidos y detectar patrones estructurales distintivos (fig.5).

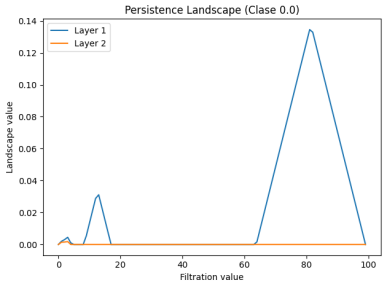

Figura 5: Persistence Landscape de la clase 0

Para continuar el análisis topológico, se trabajó con las curvas de Betti. Primero, las señales unidimensionales fueron representadas como nubes de puntos en un espacio

multidimensional utilizando una técnica de reconstrucción de la señal basada en un modelo de retardo (delay embedding). Específicamente, para cada señal, se construyó un conjunto de puntos en un espacio tridimensional ($\mathbb{R}^3$) a partir de los valores de la señal y su desplazamiento temporal, lo que proporcionó una representación embebida que capturaba la dinámica de la señal en el tiempo. Una vez que las señales fueron convertidas en nubes de puntos, se calcularon las curvas de Betti (Anexo E) utilizando ripser, un algoritmo eficiente para la construcción de diagramas de persistencia en espacios de alta dimensión. Para la implementación de este proceso se adaptó el código de Anticdimi, 2020, que sirvió como base para el cálculo de las curvas de Betti de orden 0 (β_0) y 1 (β_1), utilizadas para capturar la conectividad y los ciclos presentes en los datos, respectivamente, proporcionando información sobre la estructura topológica de las señales.

La figura 6 muestra cómo evoluciona la estructura topológica de los datos de la clase 0 a medida que se incrementa el valor de filtración. Al inicio, hay muchas componentes desconectadas (más de 120), lo que indica que los puntos están aislados entre sí. No obstante, conforme aumenta la filtración, estas componentes comienzan a conectarse, y su número disminuye rápidamente hasta llegar a una sola componente. Por otro lado, se observa la aparición de algunos ciclos en valores bajos de filtración, pero estos son pocos y desaparecen rápidamente. En conjunto, esto sugiere que los datos de esta clase tienen una estructura dispersa al inicio, con escasa presencia de formas cíclicas persistentes.

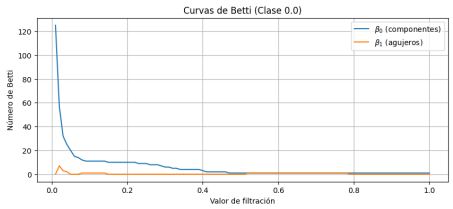

Figura 6: Curvas de Betti (β_0 y y β_1 para clase 0)

A partir de las curvas de Betti obtenidas, se extrajeron diversas características numéricas, como el área bajo la curva (AUC), el valor máximo, la media y la desviación estándar. Estas características fueron utilizadas para entrenar un modelo de RandomForestClassifier.

Además, se construyó un modelo sin aplicar técnicas de TDA, basado en una red neuronal profunda implementada con la API Sequential de Keras, el cual fue obtenido de Sa3dola, 2025. La arquitectura consistió en varias capas densas acompañadas de mecanismos de regularización para mejorar la estabilidad y evitar el sobreajuste.

El modelo incluyó una capa de entrada con 512 neuronas, seguida por tres bloques ocultos de 256, 128 y 64 neuronas, respectivamente. Cada capa fue seguida de Batch Normalization, activación LeakyReLU con ($\alpha = 0{,}3$) y una capa de Dropout con una tasa de 0.5. La salida consistió en una capa softmax con cinco neuronas, correspondiente a las clases del problema.

Se utilizó el optimizador Adam y la función de pérdida sparse categorical crossentropy, empleando la precisión como métrica principal. El entrenamiento se realizó por un máximo de 100 épocas, con un tamaño de lote de 64, validación por época y detención temprana basada en la pérdida de validación con una paciencia de 5 épocas.

Posteriormente, se desarrolló un segundo modelo de red neuronal profunda con la misma arquitectura descrita previamente, pero en este caso empleando características topológicas extraídas mediante técnicas de análisis topológico de datos (TDA). Específicamente, las señales originales fueron transformadas en nubes de puntos utilizando retraso temporal (embedding en $\mathbb{R}^3$), sobre las cuales se calcularon curvas de Betti de primer orden mediante el algoritmo Ripser. A partir de estas curvas, se extrajeron características numéricas relevantes (como el área bajo la curva, el valor máximo, la media y la desviación estándar), que fueron utilizadas como entrada al modelo.

La arquitectura de la red incluyó capas densas de 128, 64 y 32 neuronas, cada una seguida de normalización por lotes (Batch Normalization), activación LeakyReLU con $\alpha = 0{,}3$, y Dropout con tasa de 0.5. La capa de salida fue una capa softmax con un número de neuronas igual al número de clases.

El modelo fue compilado utilizando el optimizador Adam y la función de pérdida sparse categorical crossentropy, y se entrenó utilizando early stopping para prevenir el sobreajuste. Finalmente, se evaluó su desempeño en un conjunto de validación a través de métricas como la precisión y el reporte de clasificación.

Finalmente, se desarrolló un modelo de clasificación basado en aprendizaje automático tradicional utilizando técnicas del Análisis Topológico de Datos (TDA) como mecanismo principal de extracción de características. Primero, se construyó un subconjunto estratificado del conjunto de entrenamiento, compuesto por 800 observaciones, para asegurar una representación equilibrada de todas las clases. A partir de cada señal de este subconjunto, se generó una nube de puntos mediante técnicas de embedding retardado. Posteriormente, sobre estas representaciones se calcularon diagramas de persistencia mediante el algoritmo Ripser, considerando invariantes topológicos hasta dimensión 1 (es decir, componentes conexas y ciclos).

De los diagramas de persistencia obtenidos, se generaron landscapes de persistencia (funciones paisaje) de grado de homología 1 utilizando la clase PersLandscapeApprox. Estos paisajes fueron discretizados y vectorizados para transformarlos en vectores de características de longitud fija. Para estandarizar la representación, todos los vectores se rellenaron con ceros hasta alcanzar la longitud máxima observada entre ellos.

Dado el carácter de alta dimensionalidad de estas representaciones, se aplicó un análisis de componentes principales (PCA), conservando el 95 % de la varianza expli-

cada. Finalmente, se entrenó un modelo de clasificación basado en máquinas de vectores de soporte (SVM) con núcleo radial (RBF), utilizando los vectores reducidos por PCA como entrada.

El modelo fue evaluado sobre el mismo conjunto de entrenamiento y su desempeño fue reportado mediante el informe de clasificación, lo cual permitió identificar su capacidad para distinguir entre las clases en base a las características topológicas extraídas.

Resultados

En este proyecto se evaluaron diferentes enfoques para la clasificación de señales con el objetivo de identificar las características más relevantes de las señales. Para evaluar los modelos generados, se utilizó la métrica de precisión. A continuación se presentan los resultados obtenidos.

Modelo de Red Neuronal sin Características Topológicas

El primer enfoque utilizado fue una red neuronal que no incorporó características topológicas. Este modelo alcanzó una precisión de 0.97, lo que indica un rendimiento excepcional en la clasificación de las señales. Este resultado respalda la hipótesis inicial de que, sin la inclusión de características topológicas, el modelo podría aún lograr una alta precisión debido a la capacidad de las redes neuronales para aprender patrones complejos de los datos.

Red Neuronal con Características Topológicas

El segundo enfoque incorporó características topológicas en el proceso de clasificación a través de la extracción de características adicionales. En este caso, el modelo alcanzó una precisión de 0.86. Aunque el desempeño fue algo inferior al modelo sin características topológicas, aún muestra resultados competitivos. Este descenso en la precisión sugiere que las características topológicas, aunque pueden enriquecer la representación de los datos, no necesariamente contribuyen a una mejora significativa en la clasificación para este caso particular. Este resultado es relevante para evaluar si las características topológicas son verdaderamente útiles para el problema planteado, alineándose con la hipótesis inicial de que su valor podría ser limitado en ciertos contextos.

Modelo Basado en Curvas de Betti

El modelo evaluado con curvas de Betti implementó un RandomForest. Este obtuvo una precisión de 0.86, similar a la red neuronal con características topológicas. Los resultados sugieren que, aunque las curvas de Betti pueden ser efectivas para capturar la topología de los datos, su impacto en el rendimiento de la clasificación es comparable al de las características topológicas tradicionales.

Modelo PCA-SVM con TDA

Otro enfoque explorado fue la combinación de PCA (Análisis de Componentes Principales) con un clasificador SVM, al que se le añadieron características topológicas obtenidas mediante TDA (Análisis Topológico de Datos). Este modelo obtuvo una precisión de 0.83, mostrando un rendimiento similar al modelo con características topológicas. La relación entre el PCA y el SVM podría haber ayudado a reducir la dimensionalidad de los datos, pero la adición de características topológicas no resultó en una mejora clara, lo que refuerza la idea de que las características topológicas no necesariamente mejoran la precisión en todos los casos.

Análisis Comparativo y Relación con la Hipótesis

Los resultados obtenidos indican que la inclusión de características topológicas no mejora significativamente la precisión del modelo en comparación con un enfoque más simple, como la red neuronal sin características topológicas. Esto plantea una reflexión importante sobre nuestra hipótesis inicial, que sugería que las características topológicas serían clave para mejorar el rendimiento en la clasificación de señales. Si bien los modelos que incorporan TDA muestran un desempeño adecuado, el modelo sin características topológicas superó a la mayoría en términos de precisión.

En resumen, los resultados revelan que la complejidad añadida por las características topológicas no siempre contribuye de manera sustancial a la mejora del rendimiento de clasificación en este problema específico. Estos hallazgos sugieren que, en algunos contextos, modelos más simples pueden ser igualmente efectivos, lo cual es relevante tanto para la validación de nuestra hipótesis como para futuras investigaciones en el área.

Referencias

Anticdimi. (2020). tda-arrhythmia-detection/scripts at master · anticdimi/tda-arrhythmia-detection [Accessed: 2025-05-01].

Dindin, M., Umeda, Y., & Chazal, F. (2019). Topological Data Analysis for Arrhythmia Detection through Modular Neural Networks [HAL Id: hal-02155849]. *HAL Open Science*. https://doi.org/10.1000/182

Fazeli, S. (s.f.). ECG Heartbeat Categorization Dataset [Accessed: 2025-05-01].

Gobierno de México. (2023). *Cada año 220 mil personas fallecen debido a enfermedades del corazón*. %7B%5Curl%7Bhttps://www.gob.mx/salud/prensa/490-cada-ano-220-mil-personas-fallecen-debido-a-enfermedades-del-corazon%7D%7D

Hernández-Lemus, E., Miramontes, P., & Martínez-García, M. (2024). Topological Data Analysis in Cardiovascular Signals: An Overview. *Entropy*, *26*(1), 67. https://doi.org/10.3390/e26010067

Mollenhauer, M. (2018). ECG-heartbeat-classification [Accessed: 2025-05-01].

Sa3dola. (2025). ECG_heartbeat using NN [Kaggle Notebook, accessed 1 May 2025].

Anexo A: Matriz de correlación

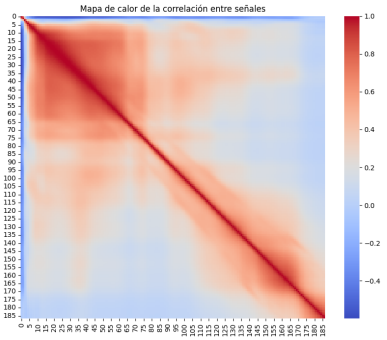

Figura A.1: Mapa de correlación entre señales

Anexo B: Diagramas de persistencia

Diagramas para un ejemplo de cada clase

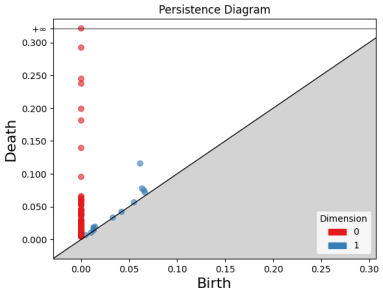

Figura A.1: Ejemplo de señal ECG clase 1

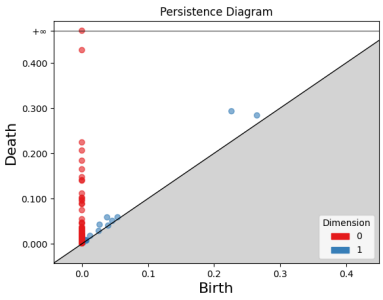

Figura A.2: Ejemplo de señal ECG clase 2

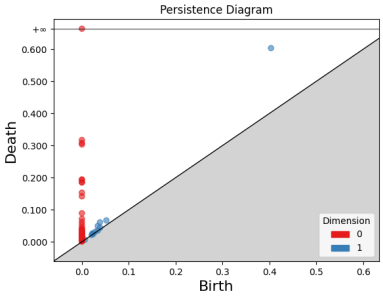

Figura A.3: Ejemplo de señal ECG clase 3

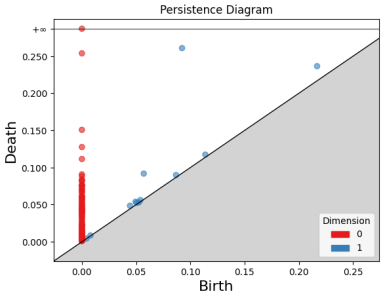

Figura A.4: Ejemplo de señal ECG clase 4

Anexo C: Imágenes de persistencia

Imágenes de persistencia para un ejemplo de cada clase

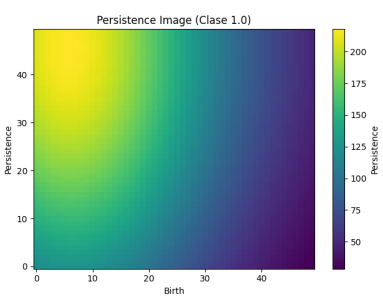

Figura B.1: Ejemplo de señal ECG clase 1

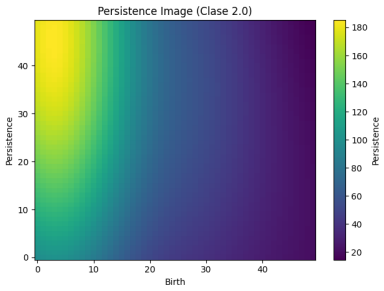

Figura B.2: Ejemplo de señal ECG clase 2

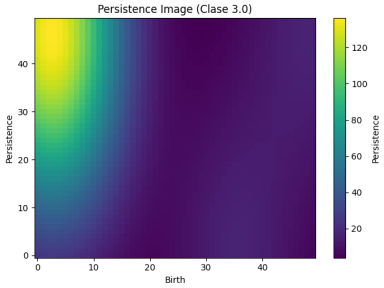

Figura B.3: Ejemplo de señal ECG clase 3

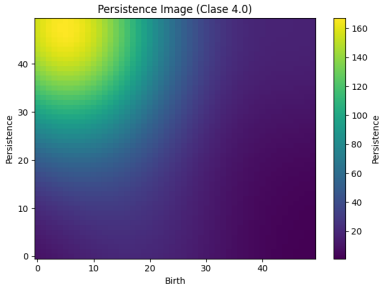

Figura B.4: Ejemplo de señal ECG clase 4

Anexo D: Persistence Landscape

Persistence Landscape para un ejemplo de cada clase

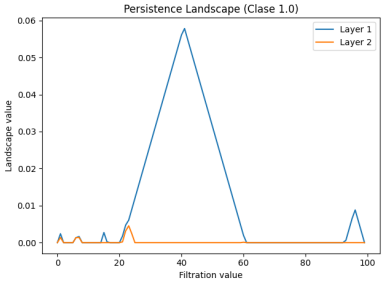

Figura C.1: Ejemplo de señal ECG clase 1

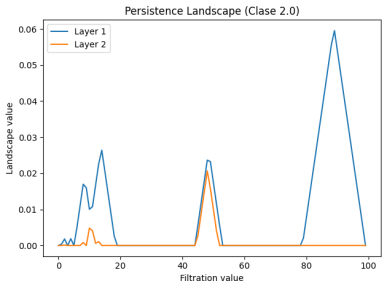

Figura C.2: Ejemplo de señal ECG clase 2

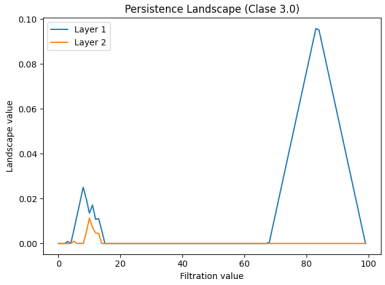

Figura C.3: Ejemplo de señal ECG clase 3

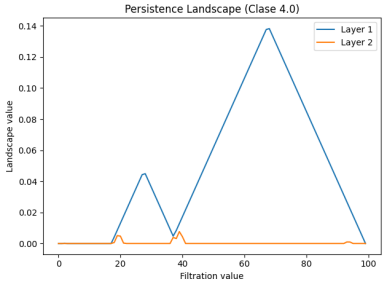

Figura C.4: Ejemplo de señal ECG clase 4

Anexo E: Curvas de Betti

Curvas de Betti para un ejemplo de cada clase

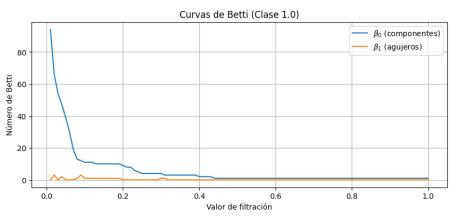

Figura D.1: Ejemplo de señal ECG clase 1

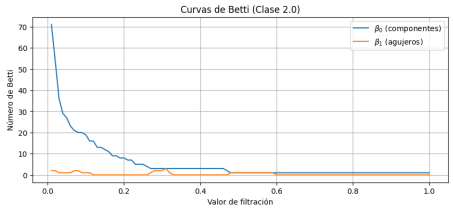

Figura D.2: Ejemplo de señal ECG clase 2

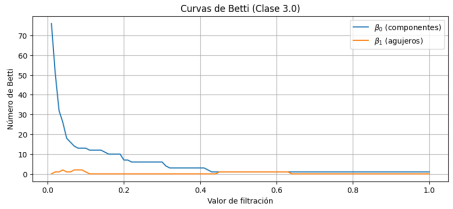

Figura D.3: Ejemplo de señal ECG clase 3

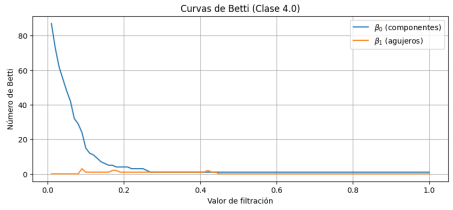

Figura D.4: Ejemplo de señal ECG clase 4