Topological Data Analysis and Convolutional Neural Networks for Gravitational Wave Detection

Felipe de Jesús Felix Arredondo[3], Sofia Alvarez Sandoval[3], David Matamoros[3], Ana Almeida-Perez[3], Valeria María Serna Salazar[3], Raúl Alejandro Pérez Saucedo[3], Alejandro Ucan-Puc[1(✉)], and Rodrigo Davila Figueroa[2]

[1] Tecnologico de Monterrey, Eugenio Garza Sada 2501, 64849 Monterrey, Mexico
alejandro.ucan-puc@tec.mx
[2] Department of Science and Mathematics, Ajman University, University Street, Al Jurf1, P.O. Box 346, Ajman, UAE
r.figueroa@ajman.ac.ae
[3] Monterrey, Nuevo León, Mexico

Abstract. This paper presents a methodology that combines topological data analysis and convolutional neural networks (CNNs) to improve gravitational wave detection. Using persistent homology, topological features are extracted and integrated into a CNN model, improving its ability to distinguish relevant signals in the presence of noise. Experiments show that this approach not only increases signal detection accuracy, but also reduces the computational complexity involved in letting the neural network learn without the aid of persistent homology. These findings open up possibilities for efficient gravitational wave detection in noisy signal analysis.

Keywords: Gravitational waves · topological data analysis (TDA) · persistent homology · convolutional neural networks (CNN) · signal detection · signal-to-noise ratio (SNR)

1 Introduction

The detection of gravitational waves is an area of physics with theoretical foundations dating back more than a century. Einstein predicted the shape of these waves in his theory of general relativity, but this field has only recently emerged as a research area.

In September 2015, physicists at the Laser Interferometer Gravitational-Wave Observatory (LIGO) detected a collision of two black holes 1.3 billion light-years from Earth. One black hole was about 36 times the mass of the Sun, and the other was about 29 solar masses. As they spun toward each other, they merged into a single, more massive gravitational sink in space-time that weighed 62 solar masses, the LIGO team estimates.

L. Martínez-Villaseñor et al. (Eds.): MICAI 2025, LNAI 16222, pp. 143–164, 2026.
https://doi.org/10.1007/978-3-032-09044-7_11

As physicist Stephen Hawking pointed out, the significance of these observations lies in the fact that the detected waveform coincided with Einstein's predictions, and thus became a confirmation of much theoretical work, including Einstein's general theory of relativity, which never even believed in the existence of black holes [7]. Such was the importance of this discovery that the 2017 Nobel Prize in Physics was awarded to the collaboration of the LIGO and VIRGO observatories (specifically to the scientists Rainer Weiss, Barry C. Barish, and Kip S. Thorne). [19].

Despite the numerous analogous observations that have been made since then, the detection of such waves has a significant challenge factor due to the impact of intrinsic and environmental noise on measurement instruments. In 2005, P. Jaranowski and A. Królak ([15]) introduced the first statistical and data analysis methods for gravitational waves' detection, on signals buried in stationary and Gaussian Noise. Later, in 2006, S. Chatterji et al. ([8]), proposed methods to detect gravitational waves in signals that have non-stationary noise; this case is where most of the techniques fail.

With the popularization of artificial intelligence and neural networks, the field of gravitational wave detection has seen a significant shift towards machine learning techniques. In 2018, H. Gabbard et al. ([13]) introduced a novel approach to gravitational wave detection using convolutional neural networks (CNNs) adapted to the findings in [8]. This work marked a significant milestone in the application of machine learning to gravitational wave astronomy, demonstrating the potential of CNNs to enhance the sensitivity and accuracy of GW detection. The authors employed CNNs to analyze time-frequency representations of gravitational wave signals, achieving impressive results in detecting and classifying various types of gravitational wave events. Other related work, in (?), the authors successfully integrated CNNs with conventional methods to detect gravitational waves, showcasing the efficacy of machine learning in this field. Similarly, [16] demonstrated the capability of CNNs to classify gravitational wave signals amidst noisy data, highlighting their utility in real-world scenarios. Nevertheless, the raw signals that are used to detect gravitational waves has a significant presence of noise. Topological descriptores are key in this task, because topology studies features that are invariant under small continuous perturbations.

Topological data analysis (TDA) is a recently developed field that applies techniques from algebraic topology to extract meaningful information from complex datasets. In F. Chazal et al. ([9–11]), the authors studied the stability of *persistence modules* and introduced the concept of *persistence diagrams* to capture the topological features of data. The work of F. Chazal et al. implies that the robustness of persistent diagrams to noise and their ability to capture essential topological features make them a powerful tool for analyzing complex datasets. The authors demonstrated the effectiveness of TDA in various applications, including LIGO glitch detection and medical imaging, showcasing its potential to uncover hidden structures and patterns in data that traditional methods may overlook.

Recent works on gravitational wave detection introduce the use of TDA. For example, in [21], the authors propose the use of Roy-Kesselman Diagrams in addition to persistent homology applied, particularly, in GWs detection. The Roy-Kesselman diagrams are an *topological event graph*, meaning that these graphs depend on the topological features of the data, and they are centered on a particular event or phenomenon. The authors propose this methodology to classify GW signals with physical characteristics. In this context, data science and topological data analysis have emerged as fields with powerful tools for extracting meaningful features from signals and distinguishing patterns in noisy data. A notable example of the joint application of these fields was presented in Christopher Bresten and Jae-Hun Jung's paper [6], where the use of convolutional neural networks (CNNs) is implemented with an additional process involving feature extraction with persistent homology. This additional step has been shown to enhance the system's resilience to noise and improve the ability to detect signals with reduced training. Their analysis yielded a substantial enhancement in efficiency, with reduced computational demands. This is notable given the prior observation that the exclusive use of CNNs had been shown to be computationally intensive and that their accuracy exhibited a significant decline when the signal-to-noise ratio was low.

In this paper, these techniques are used to develop a model that integrates topological properties into the signal processing architecture, improving the ability to identify and reconstruct signals in the presence of noise. The incorporation of topological information in the model is expected to provide a more accurate and differentiated representation of the intrinsic characteristics of the signals, which could have significant applications in various fields ranging from physics to data analysis in engineering and natural sciences.

2 Theoretical Foundations

2.1 Gravitational Waves

In 1916, Albert Einstein predicted the existence of gravitational waves within his theory of general relativity. This theory interprets gravity as a consequence of distortions in space-time caused by mass. In accordance with this theory, Einstein predicted that any event in the cosmos would cause "ripples" in space-time that would propagate in all directions. However, these ripples would be so minuscule that they would be nearly imperceptible with any technology envisioned at the time. However, the advent of technology capable of detecting such minute fluctuations would enable the observation of cataclysmic events, such as black hole collisions, which generate waves of considerable magnitude [23]. This assertion is evidenced by the observation of the GW150914 event, which exhibited a greater emission of energy (in the form of gravitational waves) than the collective emission of light by all stars within the observable universe over the same time span.

At present, two methods are employed for the detection of gravitational waves: indirect observation and direct observation. The former approach entails the analysis of astronomical phenomena influenced by these waves, a method pioneered by Russell Hulse and Joseph Taylor, who were awarded the 1993 Nobel Prize in Physics for their contributions. Their research centered on the analysis of the effects of a pulsar on a neutron star within the system PSR B1913+16, which emits electromagnetic pulses at precise and regular intervals. They observed that the frequency of these pulses and the position of both bodies approached a spiral shape, resulting in the loss of energy below the values established by Einstein for the gravitational waves that would be acting on the star [18].

Conversely, direct observation proved challenging for an extended period due to the imperceptible nature of the effects to be detected and the necessity of discerning them from concomitant vibrations within the Earth [23]. In the 1960 s, the utilization of interferometry, a family of observational techniques based on the interference of light waves (and subsequently, lasers), was proposed. This interferometer functions by fusing light sources to generate an interference pattern, which can subsequently be measured and analyzed due to its inherent value in providing information about the object or phenomenon under study. This method is primarily employed to make measurements of minimal size that cannot be achieved through alternative means. Its implementation enables enhanced precision at short wavelengths [20].

The fundamental principle underlying this technique asserts that the amplification of light waves coinciding in phase is contingent upon their synchrony, while those in phase opposition (i.e., exhibiting equivalent magnitude but opposite velocity) undergo cancellation. The LIGO interferometer comprises a laser beam that is divided into two segments, with each segment traversing an arm perpendicular to the other. At the terminal point, each arm reflects off a mirror and subsequently reunites. The degree of cancellation or amplification generated by their junction is indicative of the passage of a gravitational wave through the LIGO interferometer [1]. The sensitivity of the device allows it to detect minimal variations in the space-time continuum, comparable to the width of a hair (or approximately one ten-thousandth the width of a proton), over distances of 4,000 light-years (the separation distance between the Sun and Alpha Centauri, the nearest star to the solar system) [7].

Given that gravitational waves can be modeled as a function of finite energy, they are elements of the Hilbert space $L^2([-\pi,\pi],\mathbb{R})$, which is the space of all square integrable functions in the interval $[-\pi,\pi] \in \mathbb{R}$.

The following theorem is useful to describe a gravitational wave $G(t) \in L^2([-\pi,\pi])$:

Theorem 1 (Chap. 2 of (?)). *The sequence of functions*

$$\mathcal{B} = \left\{ \frac{1}{\sqrt{2}}, \sin(x), \cos(x), \sin(2x), \cos(2x), \ldots, \sin(nx), \cos(nx), \ldots \right\}$$

form an infinite orthonormal sequence in the space $L^2([-\pi,\pi])$.

Actually the set $\mathcal{B}$ is an orthonormal basis of $L^2([-\pi, \pi])$ and therefore we can represent the gravitational wave $G(t)$ as a linear combination of the elements of $\mathcal{B}$

$$G(t) = \frac{a_0}{\sqrt{2}} + \sum_{n=1}^{\infty} (a_n \cos(nx) + b_n \sin(nx))$$

where a_n and b_n are the Fourier coefficients given by

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} G(t) \cos(nt) dt$$

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} G(t) \sin(nt) dt$$

2.2 Persistent Homology

Persistent homology, first introduced in [28], is a tool which aims to extract and quantify significant topological features from discrete data sets at different levels of resolution. In the context of analyzing complex data signals such as gravitational waves, this technique allows for the identification of structures such as connected components, cycles, and voids, whose persistence across scales provides robust information against noise and deformations.

To apply persistent homology to discrete data (such as the embedded points of a signal in phase space), it is necessary to construct a simplicial complex that captures the proximity relationships between the points. One of the most commonly used complexes in practice is the Vietoris-Rips complex.

Let $X \subset \mathbb{R}^d$ be a finite set of points and $\epsilon > 0$ a scale parameter. The Vietoris-Rips simplicial complex $VR_\epsilon(X)$ is defined as the set of all finite simplices whose vertices are at most ϵ apart from each other. That is,

$$VR_\epsilon(X) = \{\sigma \subset X | dist(x_i, x_j) \leq \epsilon \text{ for all } x_i, x_j \in \sigma\}$$

By varying the parameter ϵ, a filtered family of simplicial complexes is obtained,

$$VR_{\epsilon_1}(X) \subset VR_{\epsilon_2}(X) \subset \dots$$

which serves as a basis for calculating the persistent homology.

Persistent homology tracks how homology classes $H_k(X)$ (for $k = 0, 1, 2, \cdots$) evolve along the complex filtration. Formally, for each dimension k, the persistent homology compute the persistent modules

$$H_k(VR_{\epsilon_1}(X)) \to H_k(VR_{\epsilon_2}(X)) \to \cdots,$$

and record the values ϵ_b and ϵ_d in which a homology class of dimension k is born and dies, respectively. These ordered pairs (ϵ_b, ϵ_d) are represented in persistence diagrams where indicates the duration of a topological feature in the filtration, or alternatively in persistence vectors that can be treated computationally.

The interpretation of these homology classes varies according to the dimension k. In dimension $k = 0$, the group H_0 describes the connected components

of the space; each individual point in the data set starts as its own component, and these are merged as ϵ grows. In dimension $k = 1$, the group H_1 describe the non-trivial cycles, i.e., the occurrence of closed loops. In dimension $k = 2$, the group H_2 corresponds to three-dimensional cavities or voids.

In the present study, the gravitational wave signal is embedded in a phase space using the Takens embedding, which preserves the topological characteristics of the underlying dynamics due to its diffeomorphism nature. The Vietoris-Rips complex is built on this embedding, and the persistent homology in dimensions 0 and 1 is calculated.

Due to the interest in cyclic signal structures, the H_1 homology group analysis is emphasized. Whose information is translated into a persistence vector that captures the duration and prominence of the cycles present in the signal. The Vietoris-Rips algorithm [4], of complexity $O(n^2)$, will be applied on subsets of $n = 200$ points, generating persistence diagrams for H_0 and H_1, although only the vectors associated with H_1 will be used for the final classification.

These vector representations will be processed by a convolutional neural network, concatenated with other outputs extracted from the raw signal, with the goal of automatically detecting sections of the signal containing noise versus those with significant cyclic information, as described in [5]. The number of persistent cycles is thus used as a robust metric to identify structured patterns in the gravitational wave signal.

2.3 Sliding Window Embedding

The sliding window embedding provides a framework to transform a time series into a trajectory in a higher-dimensional feature space, see [25]. It is achieved by sliding a fixed-size window over the signal and mapping the values within each window into a single vector in a higher-dimensional space. Let $\{x(t)\}_{t=1}^{N}$ be a time series of length N. The sliding window embedding with window size m and delay parameter τ yields a sequence of vectors $\{X_i\}_{i=1}^{N-(m-1)\tau}$, where each embedded vector X_i is defined as:

$$X_i = (x(i), x(i+\tau), x(i+2\tau), \ldots, x(i+(m-1)\tau)) \in \mathbb{R}^m. \tag{1}$$

Here, $m \in \mathbb{N}$ is the embedding dimension (also referred to as the window size), and $\tau \in \mathbb{N}$ is the lag or time delay between consecutive elements within the window. The resulting collection of vectors defines a point cloud in $\mathbb{R}^m$, capturing the temporal structure of the original signal.

In this work, we adopt the parameter values used in [5], where the authors empirically determined suitable values for the application to gravitational wave detection. Specifically, we use $m = 200$ and a stride of $s = 1$ to ensure a dense sampling of the signal. This choice of parameters produces an embedding that is sufficiently expressive to reveal topological features such as loops and voids when analyzed using persistent homology.

This sliding window point cloud serves as the input to the subsequent construction of the Vietoris–Rips complex, from which we extract topological

invariants such as persistent H_1 homology classes. These features are later used for signal classification tasks.

2.4 Convolutional Neural Networks

When working with one-dimensional neural networks, we can model gravitational waves using discrete version of convolution. In this framework, we define a one-dimensional convolutional neural network with N input channels and M output channels as follows:

$$\text{Conv}(x)_m = b_m + \sum_{n=0}^{N-1} w_{m,n} * x_n \tag{2}$$

Here, x represents the input vector, w is the weight vector, b is the bias vector, and $*$ denotes the convolution operator. This convolutional neural network acts as a linear function that applies convolution to the input data.

Since the signal of the gravitational waves are not continuous smooth functions but rather discrete time-series data, we can efficiently model them using this approach. Each point of the wave, given by $G(t)$ (where t is constant) corresponds to an input value x_i of the neural network. If we assume the weight matrix $w_{m,n}$ approximates points of a sinusoidal wave $\sin(at)$, this implies the convolutional network can extract gravitational waves features as Fourier coefficients.

This means that given a gravitational wave $G(t)$ and convolutional weights $w_{C,k}$ we can obtain the coefficients:

$$a_m = \{a_{0,m}, a_{1,m}, a_{2,m}, \cdots a_{k,m}\} = \sum_{m=0}^{N-1} w_{n,m} * G(t) \tag{3}$$

where a_m represents the output vector of coefficients found by the convolutional network, and k is the resulting convolutional size.

This relationship holds specifically when the weights $w_{m,n}$ correspond to points of sinusoidal waves $\sin(nt)$ (with $n = 1, 2, 3, \ldots$), defined as:

$$w_{n,m} = \sin(\frac{2\pi n}{N} mt) \tag{4}$$

Thus, we establish a connection between the neural network's channels and Fourier series coefficients. By convolving these weights with segments of the gravitational waves, we can extract associated coefficients for each segments of the gravitational wave, we can extract the associated coefficients for each segment (as supported by Theorem 1). This mapping can be expressed as:

$$\text{Conv}(h(t)) = A = \{a_0, a_1, a_2, \cdots a_m\} \tag{5}$$

This process effectively decomposes the time series into its Fourier components, or at least into similar basis learned by the convolutional network at each layer.

The application of multiple convolutional layers allows us to progressively increase the effective wavelength captured by the weights. This enables analysis of gravitational waves segments at different time scales, with each layer specializing in a particular temporal resolution, as demonstrated in [22].

For these reasons, convolutional neural networks prove particularly valuable for gravitational wave analysis, as they can effectively extract waves features in a manner analogous to Fourier series coefficients.

2.5 Cleaning Neural Network

Convolutional networks identify features in gravitational wave signals but do not inherently remove noise. When noise is present, many coefficients become internally inconsistent, necessitating a dedicated denoising module to filter these artifacts.

Following the approach of Bresten and Jung [5], we employ a multi layer perceptron (MLP) to process the coefficients. In their work, the MLP extracts characteristic wave patterns to perform binary classification (gravitational wave vs non-gravitational wave). Here, we adapt it to distinguish meaningful coefficients from noise, retaining only the former.

Unlike their diminishing architecture (with larger initial layers), we propose an MLP of constant width, except for the input and output layers, which must match to enable signal reconstruction from cleaned coefficients. This design draws inspiration from autoencoders [3], where the outputs replicates the input, under the assumption that gravitational waves can be embedded in a reduced space $\mathbb{R}^k$ within a higher dimensional space $\mathbb{R}^n$ (with $k < L$, where L is the wave's length).

The module of the proposed architecture consist of 3-layer network:

1. **Input:** Flattened convolutional coefficients $c \in \mathbb{R}^C$, where C is the convolution size.
2. **Hidden layers:** 128 neurons ($k = 128$), with ReLU activation and normalization layers, optimized for noise removal [27].
3. **Output:** Matches the input dimension, enabling reconstruction via transposed convolutions [26] (originally termed "deconvolutions").

Dropout is incorporated to prevent overfitting. This configuration leverage the universal approximation theorem, ensuring the model can learn the nonlinear mapping between noisy and clean coefficients.

2.6 Deconvolution Network for Gravitational Wave Reconstruction

Like convolution, deconvolution is a technique used to extract features from a gravitational wave. Its operation bears some similarities to a Fourier series [26], but its main purpose is not to describe the process, but to assign output weights to the coefficients filtered by the multilayer perceptron. Like in Eq. 4, the deconvolution layers can assign weights to the gravitational wave coefficients. Since

these weights resemble segments of sinusoidal functions -due to their analogy to Fourier series- it is possible to reconstruct the gravitational wave from the clean coefficients.

Much like a discrete Fourier series, if we have the coefficients generated by the network $c \in \mathbb{R}^{C \times H}$, where C is the size of the convolution that produced the multilayer netwrok's output and H represents the channels previously generated in the convolutional network, then, because this process maps the gravitational wave coefficients and their location in the compressed waveform, the signal can be reconstructed using the following formula:

$$h(t) = \sum_{i=0}^{C} \sum_{j=0}^{H} w_{i,j} * c_{i,j} \tag{6}$$

where $w_{i,j}$ are the convolutional network's weights, which ideally correspond to points of a sinusoidal function $\sin(at)$ (with $a = 1, 2, 3, \ldots$) and $c_{i,j}$ are the cleaned gravitational wave coefficients.

This method allows for the reconstruction of the gravitational wave from the processed coefficients, leveraging its similarity to a discrete Fourier series to recover the noise-free original signal.

3 Methodology

3.1 Data Generation

In this work we use the Giotto-TDA library's documentation [14] an artificial dataset that contains gravitational wave signals and Gaussian noise, following the method developed by Bresten and Jung [5]. The model simulates the merger of a non-spinning binary black hole system, with noise behaving as a Gaussian distribution. Each time series has a 50% probability of containing an injected gravitational wave at a random time index, and otherwise consists solely of Gaussian noise. The resulting time series follows the expression:

$$s = g + \epsilon \frac{1}{R} \xi$$

where g is the reference gravitational wave form, ξ is Gaussian noise, $\epsilon = 10^{-19}$ is a scaling factor for noise amplitude and $R \in (0.075, 0.65)$ is the signal-to-noise ratio (SNR) parameter, which controls noise intensity. Unlike the original study [5], this work generates 10000 independent time series with 100 distinct SNR values uniformly sampled within the interval. A higher value of R corresponds to less noise in the signal, while lower values indicate more noise. This approach increase variability in the dataset and enhances the neural network ability to detect signals, particularly in low-SNR regimes.

3.2 Exploratory Data Analysis

Before performing topological data analysis on the generated gravitational waves and noise, a comparative statistical analysis was conducted of their amplitudes and power spectral density. The results reveal significant differences between both signals: while the noise displays a broad normal distribution (left Fig. 1), gravitational waves show a pronounced amplitude concentration with a peak near zero. This contrast becomes more evident when examining the power spectral density (right Fig. 1), where gravitational waves concentrate their energy at extremely low frequencies (near $0hz$), confirming the dominance of their low-frequency components. In contrast, noise primarily affects higher frequency ranges. These preliminary findings suggest that topological data analysis could effectively separate signals from noise, though signal discrimination will first be evaluated using conventional metrics like the signal-to-noise ratio.

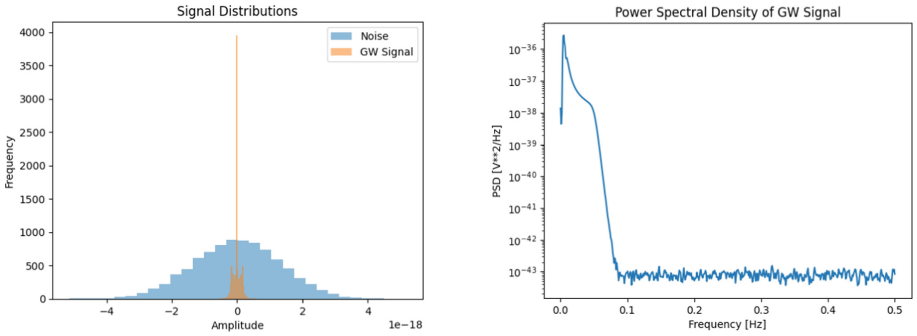


Fig. 1. Left: Amplitude distribution of noise signals and gravitational wave, Right: Spectral density of a gravitational wave signal power

SNR Evaluation. Building on both our exploratory analysis results and the original findings from the reference paper [5], we evaluate the signa-to-noise ratio (SNR), which quantifies the proportion between the transmitted signal and background noise [24]. This ratio can be calculated in decibels using:

$$SNR_{dB} = 20 \log_{10} \left(\frac{A_s}{A_r} \right) \tag{7}$$

where A_s and A_r correspond to the amplitudes of the pure signal and noise, respectively. We computed this value for each observation to precisely determine the SNR in our dataset. While theoretically it should fall within the $(2.097, 17.98)$ range (as we use the same parameters as the original study), the data generation process incorporated 100 random SNR values within this interval. Therefore, explicit calculation was necessary to determine the exact quality of each analyzed date.

Fig. 2. Top: SNR values by index signal, Bottom: Cleaned signal amplitude

Figure 2 shows that most observations exhibit a SNR above 10, indicating generally good signal quality sufficient for clear distinction from background noise [24]. However, the highest quality signals (most easily identifiable) are concentrated at very low amplitudes. For higher amplitudes, few observations show optimal SNR values; most present acceptable levels that, while sufficient for signal identification, cannot guarantee error-free processing.

3.3 Topological Data Analysis

Mapper. We employed the Mapper algorithm to analyze the separation between gravitational wave signals and noise, a technique grounded in persistent homology principles. This topological approach identifies stable structural features in the data while demonstrating remarkable resilience to noise-induced distortion. Persistent homology, as the core component of this method, uncover the essential architecture of the signals, thereby enabling accurate characterization of their intrinsic properties.

Figure 3 shows the visualization of the Mapper. The parameters used are described below:

- **Projection:** PCA with 4 principal components (`PCA(n_components=4)`).
- **Number of cubes:** 8 (`N_CUBES=8`).
- **Overlap percentage:** 0.4 (`PERC_OVERLAP=0.4`).
- **Clusterer:** KMeans with 3 clusters (`KMeans(n_clusters=3)`).
- **Scaling:** MinMaxScaler (`MinMaxScaler()`).

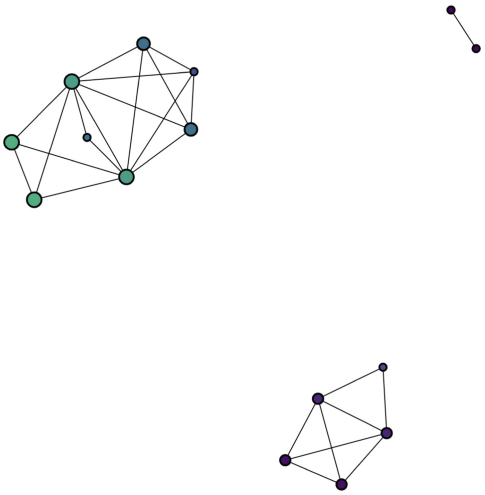


Fig. 3. Generated Mapper

- **Nodes:** 15.
- **Edges:** 26.
- **Total samples:** 215.
- **Unique samples:** 55.
- **Color function:** Row number (`Row number`).
- **Node color function:** Mean (`mean`).

The Mapper analysis produces a visual representation that confirms our previous findings regarding signal-nose separation. The cluster distribution clearly reflects this discrimination capability: while purple and blue clusters (labeled 0) predominantly represent noise, green clusters (labeled 1) mainly correspond to pure signals. These results demonstrate the promise of our approach, confirming that with the available data we can effectively distinguish between pure signals and noise, and aspect we will explore in greater depth next.

Persistent Homology. Our decision to use persistent homology in this study steams from critical insight gained by analyzing the persistence diagrams of both clean and noise-contaminated signals. The diagrams presented below provide a clear visualization of the topological features that emerge and vanish across different scales in our data, reveling fundamental differences between these two types of signals.

The persistence diagram of the noisy signal (Fig. 4) shows a distinct clustering of points near the diagonal, indicating few significantly persistent features. This characteristic pattern demonstrates how noise introduces randomness into the data, effectively eliminating stable topological structure [12].

In contrast, the clean signal's diagram (Fig. 5) displays numerous points farther from the diagonal, revealing persistent topological features. These stable

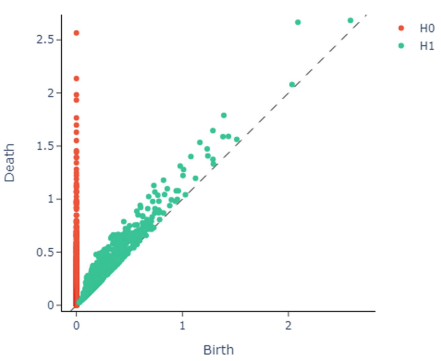


Fig. 4. Persistence diagram with noise

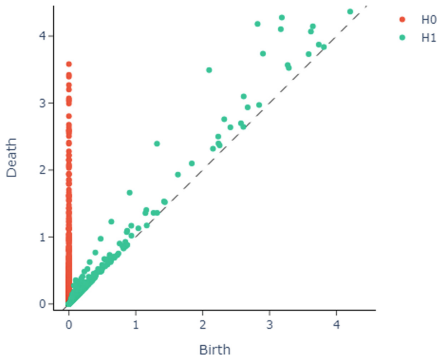


Fig. 5. Persistence diagram without noise

patterns confirm the existence of well-defined underlying structures in the original signal [17].

This striking persistence contrast not only validates our ability to distinguish signals from noise but also underscores persistent homology's potential to identify stable signal features despite interference. These persistent properties represent ideal candidates for training CNNs in signal-noise separation tasks.

3.4 Convolutional Neural Network Model

The topology-enhanced model adds a third persistent homology block to the base autoencoder architecture. This additional module analyze the signal to determine which topological features persist and which disappear across scales. The generated information -including the sizes and statistical significance of detected cycles- combines with convolutionally processed data before entering the linear layer, functioning as an intelligent topological filter.

Unlike conventional approaches where the network must independently learn noise discrimination, our model actively uses patterns identified by persistent homology to distinguish between significant cycles (real signal structures) and minor artifacts (noise). This strategy substantially reduce computational complexity by focusing analysis only on relevant features, enabling the network to process signal essentials more efficiently and achieve more accurate reconstructions.

The final architecture for gravitational wave processing integrates:

$$\text{Conv}(h(t)) \rightarrow \text{Cleaning network} \rightarrow \text{DeConv}(c(t)) \rightarrow g(t) \tag{8}$$

Or equivalently to:

$$\text{Conv}(h(t)) \rightarrow \mathbb{R}^C = R^{l,h}$$

$$\text{Proj}(\mathbb{R}^C) \rightarrow \mathbb{R}^k$$

$$\text{Proj}^{-1}(\mathbb{R}^k) \rightarrow \mathbb{R}^C$$

$$\text{DeConv}(R^C) \rightarrow g(t)$$

Where the projection Proj is a neural network that adjusts the dimensionality of cleaned coefficients, and the anti-projection Proj^{-1} restores these coefficients to their original dimensionality.

By incorporating the persistent homology module, we obtain this enhanced architecture:

$$(\text{Conv}(h(t)) + \text{Conv}(H_1)) \rightarrow \text{Cleaning network} \rightarrow \text{DeConv}(c(t)) \rightarrow g(t) \quad (9)$$

The objective is to minimize $d(h(t), g(t))$, where d represents a distance function between original and processed gravitational waveforms.

3.5 Chosing the Neural Network Parameters

The neural network parameters were selected following established conventions in specialized literature, particularly similar works like [2]. The initial CNN uses 32 channels ($c = 32$), doubling at each stage, and is optimized to extract key gravitational waves features. Each convolutional layer contains: (1)ReLU activation, (2) $MaxPooling$ operation, (3) layer normalization, and (4) Dropout ($p = 0.5$) to prevent overfitting.

The persistent homology CNN replicates this structure, maintaining the initial 32 channels and power-of-2 growth pattern. Before the cleaning stage, we implement a projection network that reduce dimensionality to $k = 128$ neurons using ReLU. This network merges outputs from both the topological module and convolutional extractor, preparing data for subsequent processing.

The cleaning network maintains constant $k = 128$ neurons in its three hidden layers, preserving the ReLU-normalization-Dropout ($p = 0.5$) structure. The final layer performs the inverse transformation, projecting from k back to c neurons. For the deconvolutional network, we start with 4×32, progressively reducing in powers of 2. The first three layers repeat the ReLU-normalization-Dropout scheme, while the final layer directly produce the 1-neuron output that reconstructs the gravitational waveform.

Figure 7 details this complete configuration.

The architectures can be found specifically in the following table containing each of the blocks shown in the previews Figs. 7:

3.6 Training

The neural network training will use the Adam optimizer, following the successful implementation reported in [5]. This choice is based on Adam's proven superior performance in neural network optimization. We configure the algorithm with a learning rate of 0.001 and employ the Huber loss function for its advantages in robust regression problems.

The training process uses a dataset of 10,000 independently generated samples of gravitational waves with artificial Gaussian noise. These are divided into 80% for training (8,000 samples) and 20% for validation (2,000 samples). The split is performed at the level of complete time series, ensuring that each injected gravitational wave and its noisy realization appear exclusively in one subset.

3.7 Evaluation

To minimize noise interference, we employ the Huber loss function as our comparative metric, defined as (Table 1):

$$L = d(h(t), g(t)) = \begin{cases} \frac{1}{2}(h(t) - g(t))^2 & \text{if } |h(t) - g(t)| \leq 1 \\ |h(t) - g(t)| - \frac{1}{2} & \text{if } |h(t) - g(t)| > 1 \end{cases} \tag{10}$$

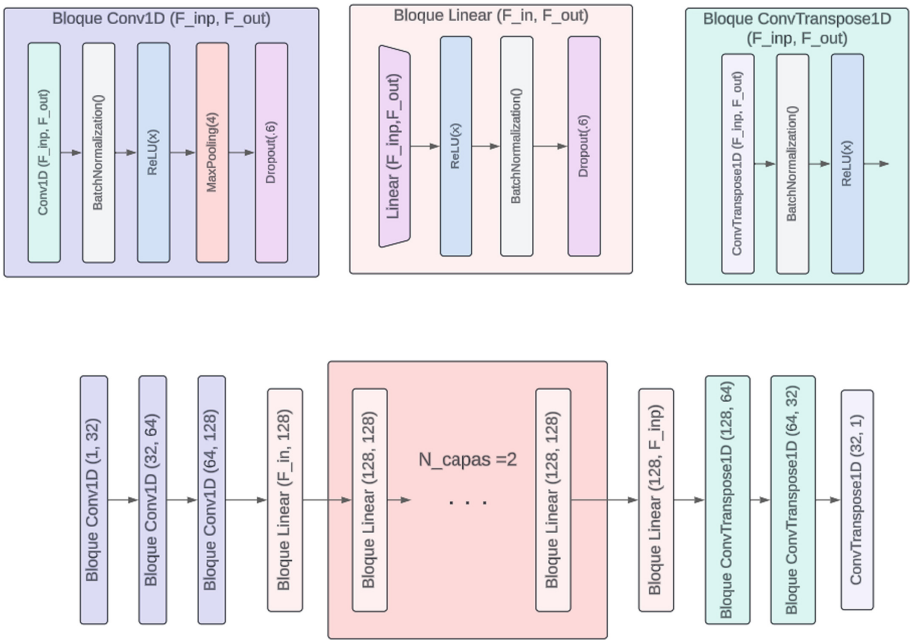


Fig. 6. Homology free neural network architecture, based on [3]

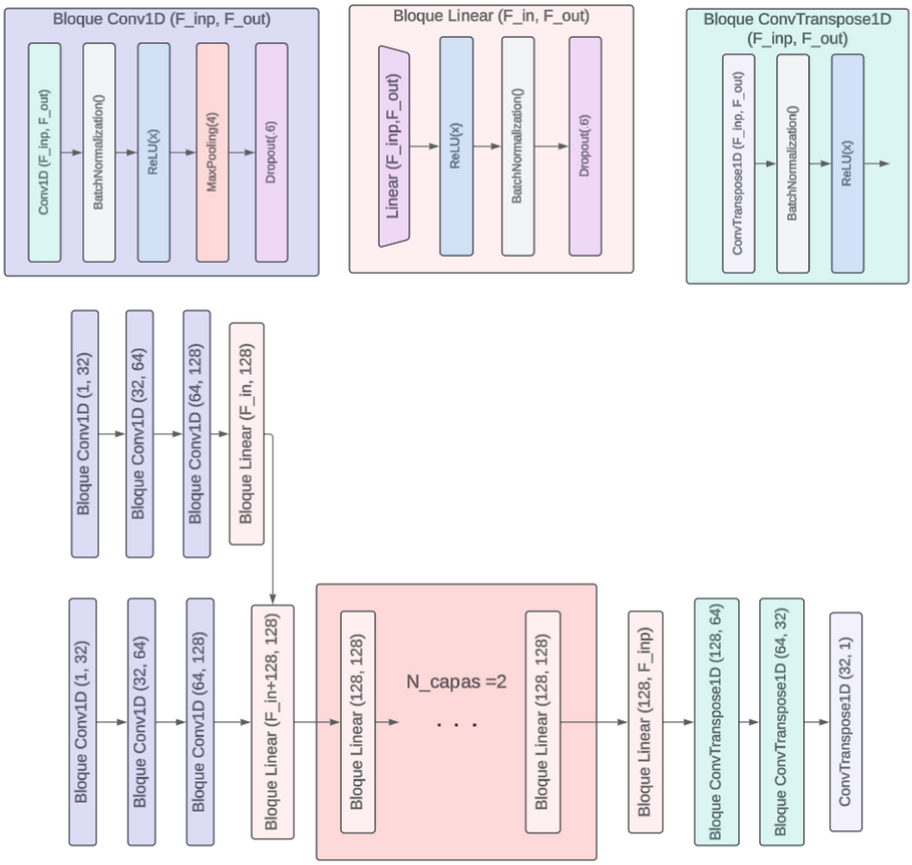


Fig. 7. Neural network architecture with homology

This function ensures balanced training by preventing extreme errors (common in scientific notation) from dominating the optimization process. The gradient of the loss with respect to the weight w is:

$$\frac{\partial L}{\partial w} = (h(t) - g(t))\frac{\partial g(t)}{\partial w} \tag{11}$$

When $h(t) \approx g(t)$, both the error and gradient diminish. The Huber function operates as follows:

- For small errors (≤ 1): Quadratic penalization accelerates convergence.
- For large errors (> 1): Linear penalization mitigates outlier influence.

To assess the persistent homology module's effectiveness, we compare our main network (Fig. 6) against a base line version without this module. Performance is evaluated using the average loss over the last 10 validation batches, quantifying whether topological features enhance model accuracy.

Table 1. Neural network architecture

Red	Layer	Neurons	Activation function	Normalization	Dropout
Convolutional	1	(1,32)	ReLU	BatchNorm	0.5
Convolutional	2	(32,64)	ReLU	BatchNorm	0.5
Convolutional	3	(64,128)	ReLU	BatchNorm	0.5
Homology	1	(1,32)	ReLU	BatchNorm	0.5
Homology	2	(32,64)	ReLU	BatchNorm	0.5
Homology	3	(64,128)	ReLU	BatchNorm	0.5
Projection	1	(C_{out},128)	ReLU	BatchNorm	0.5
Cleaning	1	(128,128)	ReLU	BatchNorm	0.5
Cleaning	2	(128,128)	ReLU	BatchNorm	0.5
Cleaning	3	(128,128)	ReLU	BatchNorm	0.5
Projection	1	(128,C_{out})	ReLU	BatchNorm	0.5
Desconvolutional	1	(128,64)	ReLU	BatchNorm	0.5
Desconvolutional	2	(64,32)	ReLU	BatchNorm	0.5
Desconvolutional	3	(32,32)	ReLU	BatchNorm	0.5
Desconvolutional	4	(32,1)	ReLU	BatchNorm	0.5

4 Implementation

The neural network was developed in Python using PyTorch, and open-source deep learning framework designed for efficient implementation and training of neural models. The system ran on hardware with the following specifications:

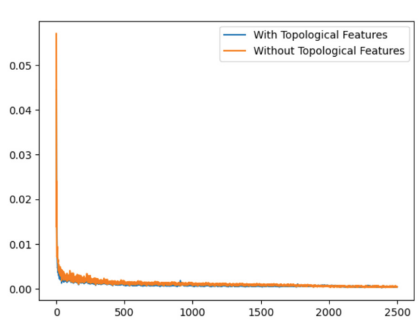
- CPU: Intel Core i7 (8 cores, 16 threads)
- RAM: 16 GB
- GPU: NVIDIA GeForce RTX 2080 (8 GB VRAM)

5 Results and Analysis

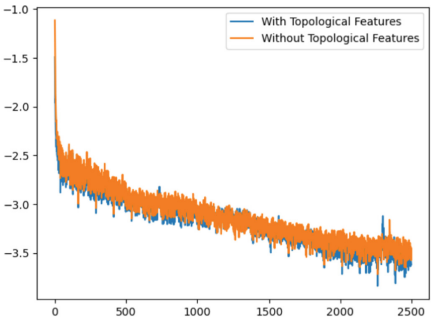
We assessed performance by comparing the average loss of the last 10 validation batches between networks with and without persistent homology. Results (Fig. 8) demonstrate that the persistent homology-enhanced network achieves lower loss values in both training and validation, consistently outperforming the conventional version. Notably, its validation loss is even lower than training loss, confirming superior gravitational wave denoising capability.

Since loss values were extremely small (in scientific notation), we applied logarithms for clearer comparison. The 0.35 average difference favors the persistent homology network, translating to a 70% efficiency improvement

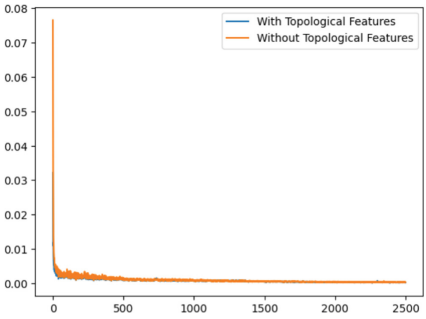
$$\frac{L_n}{L_h} = \frac{e^{l_n}}{e^{l_h}} = e^{-0.35} \approx 0.70 \approx 70\%$$



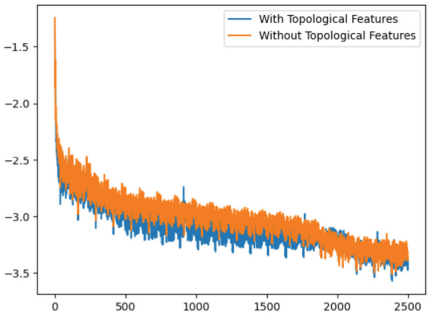
(a) Loss in the training set



(b) Logarithm of the loss in the training set

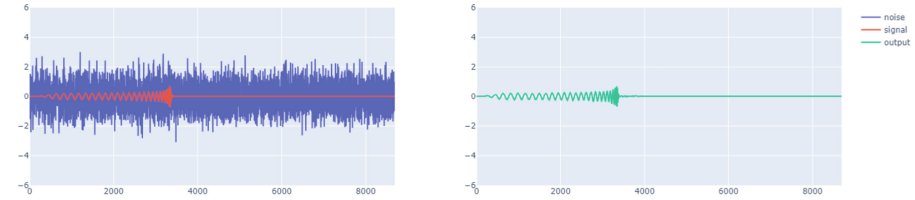


(c) Loss in the validation set

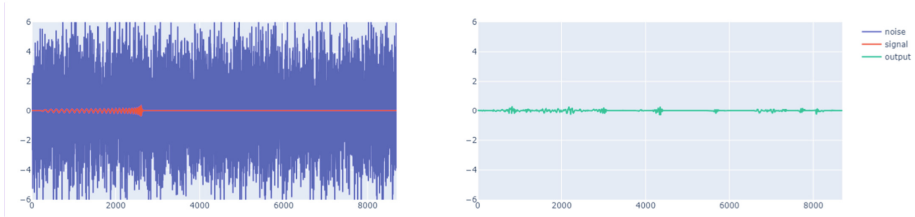


(d) Logarithm of loss in the validation set

Fig. 8. Neural network result with persistent homology and without persistent homology



(a) Gravitational wave with homology



(b) Gravitational wave without homology

Fig. 9. Neural network result with persistent homology

This advantage manifests not only in final results but also in faster training convergence. Reconstruction analysis (Fig. 9a) shows the persistent homology network effectively removes noise while preserving the gravitational wave's original shape, frequency, and amplitude, unlike its conventional counterpart achieving only partial recovery.

The following table quantitatively summarizes these comparative results (Table 2):

Table 2. Neural network results

Net	Loss	LossLog	TestLoss	TestLossLog
With homology	$0.4e^{-4}$	−3.4	$0.3e^{-5}$	−3.4
Without homology	$0.6e^{-4}$	−3.3	$0.4e^{-5}$	−3.2

5.1 Analysis

While the persistent homology network demonstrates superior performance, its main drawback lies in the computational time required to generate persistence vectors, which significantly increases training time compared to the conventional network. Although persistence vectors are precomputed to reduce cost, the training process remains substantially slower. This creates a critical trade-off between accuracy improvement and computational cost, necessitating careful evaluation based on specific application requirements.

6 Conclusions

The model significantly improved convergence, denoising and reconstruction of gravitational waves, enabling not just signal detection in noise but full recovery of original waveform through clean coefficients.

6.1 Limitations

Although the persistent homology network shows superior denoising and reconstruction performance, it has several important limitations. First, the approach relies on synthetic gravitational wave signals generated under idealized conditions. While these simulations are useful for testing the concept, they do not fully capture the complexity of real detector environments, such as those of LIGO and Virgo, where instrumental and environmental noise is highly non-Gaussian and time-dependent. Generalization to real-world data therefore requires transfer learning or the use of hybrid datasets combining synthetic and experimental signals.

A second limitation is the computational cost of generating persistence vectors. Although these vectors are precomputed to optimize training, calculating persistent homology significantly increases runtime compared to conventional convolutional neural networks (CNNs). The trade-off between accuracy and scalability must be carefully considered when applying the method to very large datasets.

Third, the current method requires prior knowledge of the general structure of gravitational waveforms to generate noisy data for training. Without this knowledge, the network's denoising capability diminishes, limiting its applicability to scenarios where signal morphology is unknown.

Finally, compressing topological information into fixed-size persistence vectors may eliminate long-range or fine-grained temporal patterns. Future research should explore architectures, such as recurrent or transformer-based neural networks, that can process persistent homology as a sequence rather than a static vector. These architectures could enable the model to capture richer dynamics and adapt better to variable-length signals.

6.2 Future Research

The current persistent homology analysis method has notable limitations. By processing data through convolutional layers and then sorting by persistence, we might lose essential patterns from the original behavior. Furthermore, compressing the data into a fixed-size vector of N_1 points could discard valuable information. To address these issues, we propose that future research should implement recurrent neural networks (RNNs) capable of treating persistent homology as a time series rather than a static vector. This approach would not only improve analysis efficiency but also removes the fixed-size constraint, allowing more natural handling of variable-length sequences.

7 Algorithm

The algorithm implements an end-to-end process covering data processing to final wave reconstruction, including comparative evaluation between models with and without persistent homology. The following pseudocode details its structure:

$X, Y \leftarrow$ Noise gravitational wave data
$X_{train}, X_{test}, Y_{train}, Y_{test} \leftarrow$ Separate data in training and validation
$H_1 \leftarrow$ Calculate persistent homology
$X_{train} \leftarrow$ Concatenate X_{train} with H_1
$X_{test} \leftarrow$ ConcatenateX_{test} with H_1
$model \leftarrow$ Neural network with persistent homology
$model_{no} \leftarrow$ Neural network without persistent homology
for $epoch \leftarrow 1, 20$ **do**
  for $batch \leftarrow 1, 128$ **do**
    $X_{batch}, Y_{batch} \leftarrow$ Next training batch

$loss \leftarrow$ Huber loss function
$loss_{no} \leftarrow$ Huber loss function
$model$.backward()
$model_{no}$.backward()
$model$.step()
$model_{no}$.step()
end for
end for
$loss \leftarrow$ Huber loss function
$loss_{no} \leftarrow$ Huber loss function
$X_{test} \leftarrow$ Concatenate X_{test} with H_1
$Y_{pred} \leftarrow model(X_{test})$
$Y_{pred_{no}} \leftarrow model_{no}(X_{test})$
$loss \leftarrow$ Huber loss function
$loss_{no} \leftarrow$ Huber loss function
$Y_{pred} \leftarrow$ Reconstruct gravitational wave
$Y_{pred_{no}} \leftarrow$ Reconstruct gravitational wave

Acknowledgments. We would like to thank Fondo de Apoyo a Publicaciones del Tecnologico de Monterrey for supporting part of this work.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. What is an interferometer? https://www.ligo.caltech.edu/page/what-is-interferometer (2024)
2. Alom, M.Z., et al.: The history began from AlexNet: A comprehensive survey on deep learning approaches (2018)
3. Bank, D., Koenigstein, N., Giryes, R.: Autoencoders (2021)
4. Bauer, U.: Ripser: efficient computation of vietoris–rips persistence barcodes. J. Appl. Comput. Topology **5**(3), 391–423 (2021)
5. Bresten, C., Jung, J.-H.: Detection of gravitational waves using topological data analysis and convolutional neural network: An improved approach (2019)
6. Bresten, C., Jung, J.-H.: Detection of gravitational waves using topological data analysis and convolutional neural network. J. Korean Soc. Ind. Appl. Math. **28**(4), 243–255 (2024)
7. Castelvecchi, D., Witze, A.: Einstein's gravitational waves found at last. Nature (2016)
8. Chatterji, S., Lazzarini, A., Stein, L., Sutton, P.J., Searle, A., Tinto, M.: Coherent network analysis technique for discriminating gravitational-wave bursts from instrumental noise. Phys. Rev. D—Part. Fields Gravit. Cosmol. **74**(8), 082005 (2006)
9. Chazal, F., Cohen-Steiner, D., Guibas, L.J., Oudot, S.: The Stability of Persistence Diagrams Revisited. Research Report RR-6568 (2008)

10. Chazal, F., De Silva, V., Glisse, M., Oudot, S.: The structure and stability of persistence modules, vol. 10. Springer (2016)
11. Chazal, F., Michel, B.: An introduction to topological data analysis: fundamental and practical aspects for data scientists. Front. Artif. Intell. **4**, 9 (2021)
12. Edelsbrunner, H., Cohen-Steiner, J.H.D.: Stability of persistence diagrams. Discrete Comput. Geom. **1**, 12 (2007)
13. Gabbard, H., Williams, M., Hayes, F., Messenger, C.: Matching matched filtering with deep networks for gravitational-wave astronomy. Phys. Rev. Lett. **120**(14), 141103 (2018)
14. Giotto-AI. Gravitational waves detection. https://giotto-ai.github.io/gtda-docs/0.3.0/notebooks/gravitational_waves_detection.html (2020)
15. Jaranowski, P., Królak, A.: Gravitational-wave data analysis. formalism and sample applications: the gaussian case. Living Rev. Relativ. **8**(1), 3 (2005)
16. Li, X.-R., Yu, W.-L., Fan, X.-L., Babu, G.J.: Some optimizations on detecting gravitational wave using convolutional neural network. Front. Phys. **15**(5), 1–11 (2020). https://doi.org/10.1007/s11467-020-0966-4
17. Obayashi, I., Hiraoka, Y., Kimura, M.: Persistence diagrams with linear machine learning models. J. Appl. Comput. Topology **1** (2018)
18. Royal Swedish Academy of Sciences. The Nobel prize in physics 1993 - press release. https://www.nobelprize.org/prizes/physics/1993/press-release/ (1993)
19. Royal Swedish Academy of Sciences. The Nobel prize in physics 1993 - press release. https://www.nobelprize.org/prizes/physics/2017/press-release/ (2017)
20. Renishaw. Interferometry explained. https://www.renishaw.com/en/interferometry-explained--7854 (2022)
21. Roy, A., Kesselman, A.: A novel approach to topological graph theory with rk diagrams and gravitational wave analysis. arXiv preprint arXiv:2201.06923 (2021)
22. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition (2015)
23. Max Planck Society. The long road towards evidence. https://www.mpg.de/9966773/background (2016)
24. Cadence Design Systems. What is signal-to-noise ratio and how to calculate it. https://resources.pcb.cadence.com/blog/2020-what-is-signal-to-noise-ratio-and-how-to-calculate-it (2020)
25. Takens, F.: Detecting strange attractors in turbulence. In: Dynamical Systems and Turbulence, Warwick 1980: proceedings of a symposium held at the University of Warwick 1979/80, pp. 366–381. Springer (2006)
26. Zeiler, M.D., Krishnan, D., Taylor, G.W., Fergus, R.: Deconvolutional networks. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2528–2535 (2010)
27. Zhang, Z., Li, X., Li, Y., Dong, Y., Wang, D., Xiong, S.: Neural noise embedding for end-to-end speech enhancement with conditional layer normalization. In: ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 7113–7117 (2021)
28. Zomorodian, A., Carlsson, G.: Computing persistent homology. In: Proceedings of the Twentieth Annual Symposium on Computational Geometry, pp. 347–356 (2004)