

Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

Neural network method to search for long transient gravitational waves

This is the final peer-reviewed author's accepted manuscript (postprint) of the following publication:

Published Version:

Attadio, F., Ricca, L., Serra, M., Palomba, C., Astone, P., Dall'Osso, S., et al. (2024). Neural network method to search for long transient gravitational waves. PHYSICAL REVIEW D, 110(10), 1-15 [10.1103/physrevd.110.103047].

Availability:

This version is available at: <https://hdl.handle.net/11585/1029753> since: 2025-11-23

Published:

DOI: <http://doi.org/10.1103/physrevd.110.103047>

Terms of use:

Some rights reserved. The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (<https://cris.unibo.it/>).
When citing, please refer to the published version.

(Article begins on next page)

A neural network method to search for long transient gravitational waves

Francesca Attadio^{1,2}, Leonardo Ricca¹, Marco Serra², Cristiano Palomba², Pia Astone²,
 Simone Dall’Osso^{2,1}, Stefano Dal Pra³, Sabrina D’Antonio⁴, Matteo Di Giovanni^{1,2}, Luca
 D’Onofrio², Paola Leaci^{1,2}, Federico Muciaccia^{1,2}, Lorenzo Pierini², and Francesco Safai Tehrani²

¹*Università di Roma La Sapienza, 00185 Roma, Italy*

²*INFN, Sezione di Roma, 00185 Roma, Italy*

³*INFN, CNAF, 40127 Bologna, Italy*

⁴*INFN, Sezione di Roma Tor Vergata, 00133 Roma, Italy*

We present a new method to search for long transient gravitational waves signals, like those expected from fast spinning newborn magnetars, in interferometric detector data. Standard search techniques are computationally unfeasible (matched filtering) or very demanding (sub-optimal semi-coherent methods). We explored a different approach by means of machine learning paradigms, to define a fast and inexpensive procedure. We used convolutional neural networks to develop a classifier that is able to discriminate between the presence or the absence of a signal. To complement the classification and enhance its effectiveness, we also developed a denoiser. We studied the performance of both networks with simulated colored noise, according to the design noise curve of LIGO interferometers. We show that the combination of the two models is crucial to increase the chance of detection. Indeed, as we decreased the signal initial amplitude (from 10^{-22} down to 10^{-23}) the classification task became more difficult. In particular, we could not correctly tag signals with an initial amplitude of 2×10^{-23} without using the denoiser. By studying the performance of the combined networks, we found a good compromise between the search false alarm rate (2%) and efficiency (90%) for a single interferometer. In addition, we demonstrated that our method is robust with respect to changes in the power law describing the time evolution of the signal frequency. Our results highlight the computationally low cost of this method to generate triggers for long transient signals. The study carried out in this work lays the foundations for further improvements, with the purpose of developing a pipeline able to perform systematic searches of long transient signals.

I. INTRODUCTION

The first gravitational wave (GW) event, GW150914, was detected in 2015 [1], and was produced by the coalescence of two black holes with masses $\sim 29 M_{\odot}$ and $36 M_{\odot}$, respectively. Since then, more than 90 events have been detected, all due to the inspiral and coalescence of neutron star (NS) or black hole binaries [2]. Such signals are classified as short transients, with a duration lasting from a fraction of a second to a few tens of seconds in the band of LIGO [3], Virgo [4] and KAGRA [5] ground-based detectors.

Several other source classes are searched for in GW data [6], and have not been detected yet. Among them, there are sources of continuous quasi-periodic GWs (CW), e.g., asymmetric spinning neutron stars [7]. For such sources, the signal frequency is linked to the star’s rotational frequency by a proportionality factor and slowly decreases in time at a rate (i.e. spin-down) $\lesssim (10^{-10} \text{ Hz/s})$, due to the emission of GW itself and likely other energy losses, like the emission of electromagnetic (EM) radiation. So far, interesting upper limits on the amplitude of possible CW signals have been placed (see e.g., [8–19]).

A variety of models (e.g. [20–23]) have been proposed for millisecond spinning, highly asymmetric NSs (likely, newborn magnetars) emitting quasi-periodic GW signals over a limited time interval (in the detector’s band), from minutes to several hours, with a spin-down $\sim \mathcal{O}(\text{Hz/s})$. Such signals are generically called *long tran-*

sients (or transient continuous waves, tCWs)[24]. As we currently do not know exactly how matter behaves at supra-nuclear densities, the detection of this kind of signals will be a powerful additional tool in the study of the NS interior structure and behavior. Typical search techniques used for tCWs, based on semi-coherent methods (see e.g., [25–28], applied for instance in [29, 30]) or on matched filter (see e.g., [31, 32]), are computationally demanding.

Here we propose a machine learning (ML) technique to perform a robust and computationally feasible search and produce a first list of candidates. With the exception of the initial studies done in [33, 34] there are no ML procedures implemented for this kind of signal. In general, data analysis using neural networks (NNs) requires less computing power than more classical algorithms. Indeed, once a model¹ is trained, it can be applied to different cases, as we will see in the subsection IV E.

The aim of this work is to develop ML techniques to analyze ground-based interferometric detector data. This paper presents the implementation of a NN *classifier* capable of identifying the presence or absence of tCW signals. Data are represented using time-frequency maps, inputs to the classifier model built with convolutional neural networks (CNN) [35]. An additional neural network model has been placed before the classifier itself

¹ Model aka neural network in this paper.

to improve its performance, specifically a NN *denoiser* model exploiting residual learning [36].

The paper is organized as follows. In Sec. II, we describe the main features of the tCW expected from a newly born magnetar. In Sec. III we explain how we build time-frequency maps, input to our artificial neural networks, and how we choose the parameter space to simulate the signal. In Sec. IV, we report the results of our study, describing first the denoiser and classifier performances and subsequently the robustness of our models. Finally, we draw conclusions in Sec. V.

II. SIGNAL MODEL

The rotational energy of an isolated NS is dissipated through the emission of EM and/or gravitational radiation, implying a time variation of the rotational frequency described by the spin-down equation (e.g., [37]):

$$\dot{f}_{\text{rot}} = -k f_{\text{rot}}^n \quad (1)$$

where the spin-down constant k and the braking index n are both determined by the emission mechanism. The value of the braking index is related to the emission process, and the constant contains information about the cause of the emission. For example, in the case of magnetic dipole emission, $n = 3$, and k depends on the strength of the dipolar magnetic field (and on the NS moment of inertia, I).

In general, an isolated NS spinning with frequency f_{rot} can emit GWs if it presents an asymmetry with respect to the spin axis. GW emission due to the NS time-varying mass quadrupole corresponds to $n = 5$ (e.g., [38]), although some instabilities, e.g. r -modes, lead to GW emission with $n = 7$. If, in addition, the rotational energy is radiated only via GWs, then the spin-down equation (1) becomes

$$\dot{f}_{\text{rot}} = -k_{\text{GW}} f_{\text{rot}}^5, \quad k_{\text{GW}} = \frac{32}{5} \frac{G I \epsilon^2 \pi^4}{c^5} \quad (2)$$

where G is Newton's gravitational constant, c is the speed of light, I is the moment of inertia along the rotational axis, and ϵ is the ellipticity, a measure of the star asymmetry defined as

$$\epsilon \equiv \frac{|I_1 - I_2|}{I}, \quad (3)$$

where I_1 and I_2 are the moments of inertia around the other two principal axes.

Magnetars are a particular class of NSs, characterized by a strong external magnetic field, $10^{14} - 10^{15}$ G, and an even stronger interior component [22]. They originate from a (significant) fraction, at least $\sim 10\%$ [39] of stellar collapses, and are also expected to be formed as remnants in (a minority of) binary NS mergers, depending on details of the equation of state (EoS) of NS matter [40–43]. Their ellipticity, at formation, is expected to be

$\sim 10^{-4} - 10^{-3}$, the order of the magnetic-to-binding energy ratio (e.g. [38]), when the super-strong magnetic field is the main source of anisotropic stresses².

The solution of Eq. (2) is

$$f_{\text{rot}}(t) = f_{0,\text{rot}} \left(1 + \frac{t}{\tau}\right)^{-\frac{1}{4}} \quad \tau = \frac{1}{4 f_{0,\text{rot}}^4 k_{\text{GW}}}, \quad (4)$$

where $f_{0,\text{rot}}$ is the initial frequency and $\tau = f_{0,\text{rot}}/4\dot{f}_{0,\text{rot}}$ the characteristic spin-down time.

The source was modelled as an ellipsoid that rotates along one of its principal axes for which the GW frequency $f(t)$ is linked to $f(t)_{\text{rot}}$ by [45]

$$f(t) = 2f(t)_{\text{rot}}. \quad (5)$$

The GW signal received by an observer at a distance d , whose line of sight makes an angle ι with the direction of the spin of the star, is described in terms of the two wave polarization modes:

$$\begin{aligned} h_+(t) &= h_0(t) \frac{1 + \cos^2 \iota}{2} \cos(2\pi t f(t)) \\ h_x(t) &= h_0(t) \cos \iota \sin(2\pi t f(t)), \end{aligned} \quad (6)$$

where

$$h_0(t) = \frac{4\pi^2 G}{c^4} \frac{I f(t)^2}{d} \epsilon. \quad (7)$$

III. METHODS

The method we present involves the implementation of an NN that properly initialized allows the presence of a tGW to be distinguished from the noise present in the data produced by an interferometer. The data format we use and the NNs implemented are described here.

A. Introductory concepts of machine learning

Machine learning is a type of Artificial Intelligence (AI) that allows computers to learn without being explicitly programmed [46]. It involves feeding data into algorithms (implemented in a network *model*) that can then identify patterns and make predictions on new data. ML implementations are classified into four major categories, depending on the nature of the learning “signal” or “response” available to a learning system. Among these we use *Supervised Learning*, that is the ML task of learning a function that maps an input to an output based on

² Hydrodynamical instabilities like, e.g., the secular bar-mode instability may lead to an even larger degree of asymmetry, although shorter-lived ([44] and references therein)

example input-output pairs. Both "classification" and "denoising" problems are supervised learning problems, solved with specific models (*classifiers*, *denoisers*). A *classifier* is a model that tries to predict the correct category between distinct classes ("labels") of a given input data. The *denoiser* task involves removing noise from an image (from a data representation in which signal and noise are present). In our task, the models are fully initialized to use ("trained") using the labeled training data (*signal vs absence of signal*). Before we get to the implementation of our models we introduce the data format we use in the next paragraph.

B. Time-frequency maps

A time-frequency representation of data is a well suited starting point for the search of tCW signals. We have developed a MATLAB code to generate various kinds of time-frequency maps, in particular spectrograms, exploiting the SNAG package [47]. The code starts from the data time series, divides it into multiple segments, and then, for each chunk, using the Fast Fourier Transform (FFT) algorithm, computes an estimator of the power spectrum, namely the periodogram, which is the square modulus of the Fourier transform.

Figure 1 shows examples of spectrograms, with time on the horizontal axis and frequency on the vertical axis, computed from simulated colored noise [48] following the LIGO O4 design sensitivity curve [49]. The color bar corresponds to amplitude spectral density (ASD), i.e. the square root of the spectrum, associated with each pair of time and frequency values.

The spectrogram can be represented as a matrix where each column corresponds to the modulus of the FFT of a time segment of duration Δt . The size of the frequency bin is

$$\delta f = \frac{1}{\Delta t}. \quad (8)$$

Given that we are computing our FFT interlaced by half, the number of FFTs in a time interval T is

$$N_t = \text{floor} \left(\frac{2T}{\Delta t} \right) \quad (9)$$

and, hence, our time bin δt , i.e. our time resolution, is

$$\delta t = \frac{\Delta t}{2}. \quad (10)$$

The choice of the time bin δt is based on the characteristics of the signals. Ideally, in order to maximize the signal-to-noise ratio, one would choose the longest duration, which still allows the signal power to remain confined in a single frequency bin, i.e. ($\dot{f}\delta t \leq \delta f = 1/(2\delta t)$). However, different signals can have widely different values of $\dot{f}$, i.e. different slopes in the time-frequency plane, depending on the signal parameters. Given the parameter space we explore (see Eq. (12) and related discussion),

after various tests, we have concluded that a reasonable choice is to use a segment duration $\delta t = 2$ seconds. In addition, in order to build squared maps, we computed each spectrogram over a time interval of 1200 s and a frequency band of 150 Hz. In this way, we have maps of 600×600 pixels, where each pixel has a single value assigned. So a time-frequency map can be represented as a gray-scale image. The choice of this representation was suggested by the properties of the signal we are searching for, as discussed later in this section. The size (600×600) was also chosen also taking into account the maximum memory that can be used on the GPU in NN training and the layers we used to implement the model.

Since we are computing the FFT of segments of finite duration, we have to account for spectral leakage, which consists of a spread of the power content of a frequency bin among neighboring bins. To reduce spectral leakage, we adopted a flat-top window, defined as a sum of cosine functions:

$$f_{\cos}(x_i) = \frac{1}{2} \left(1 - \cos \left(\frac{4\pi x_i}{N} \right) \right) \quad i \leq \frac{N}{4}, \quad i \geq \frac{N}{4} \quad (11)$$

$$f_{\cos}(x_i) = 1 \quad \frac{N}{4} < i < \frac{N}{4}$$

where N is the number of samples in the segment duration considered and x_i are the samples ($i = 1, \dots, N$).

To train the NN needed for the analysis, we created three types of maps containing, respectively, only noise, only the signal and the sum of noise plus signal, as shown in Fig. 1, from left to right. To simulate the signal, according to Eq. (6), we fixed a fiducial value for the NS momentum of inertia, $I = 1.4 \times 10^{38} \text{ kg m}^2$. As for the angle of sight, we used $\iota = 55^\circ$, representative of the average inclination. Finally, we set the following ranges for the initial frequency (f_0) and NS ϵ , respectively:

$$f_0 \in [1.25, 2.00] \text{ kHz} \quad \epsilon \in [3, 30] \times 10^{-4}, \quad (12)$$

which resulted from a combination of computational considerations, memory usage, signal shape and astrophysical arguments (see, e.g., [50]).

As shown in Eq. (4), different values of f_0 and ϵ lead to widely different signal shapes in the time-frequency plane. In particular, a higher initial frequency and/or larger ellipticity imply a much faster frequency evolution, which, in some cases, may decrease by over 150 Hz in less than 1200 s. Such signals would therefore cross multiple maps. To augment the chances of retrieving them, maps were interlaced by 75 Hz. As an example, Fig. 2 depicts a signal with $f_0 = 1985 \text{ Hz}$ and $\epsilon = 2.8 \times 10^{-3}$ covering a range wider than 300 Hz in 1200 s, thus appearing in 6 interlaced maps.

The last degree of freedom is the distance to the magnetar or, equivalently, the initial GW amplitude. We simulated signals at different fixed initial amplitudes, and here we report only the case with 2×10^{-23} , our current limit, which is the minimum amplitude for which the use of the denoiser turned out to be beneficial to train

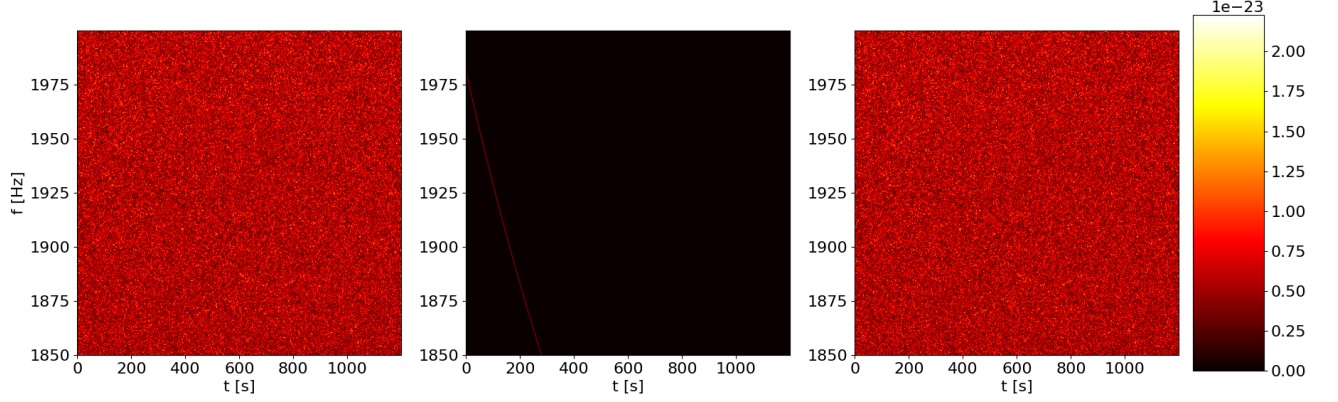


FIG. 1: Example of spectrograms of noise plus signal, only signal and only noise. The physical parameters for the signal are $f_0=1982$ Hz, $\epsilon=0.0028$, $h_0 = 2 \times 10^{-23}$. Each map has a frequency range between 1850 Hz and 2000 Hz and covers a time interval of 1200 s. The noise is simulated according to the O4 (begin: May 24, 2023, end: planned June 2025) design sensitivity curve of LIGO interferometers.

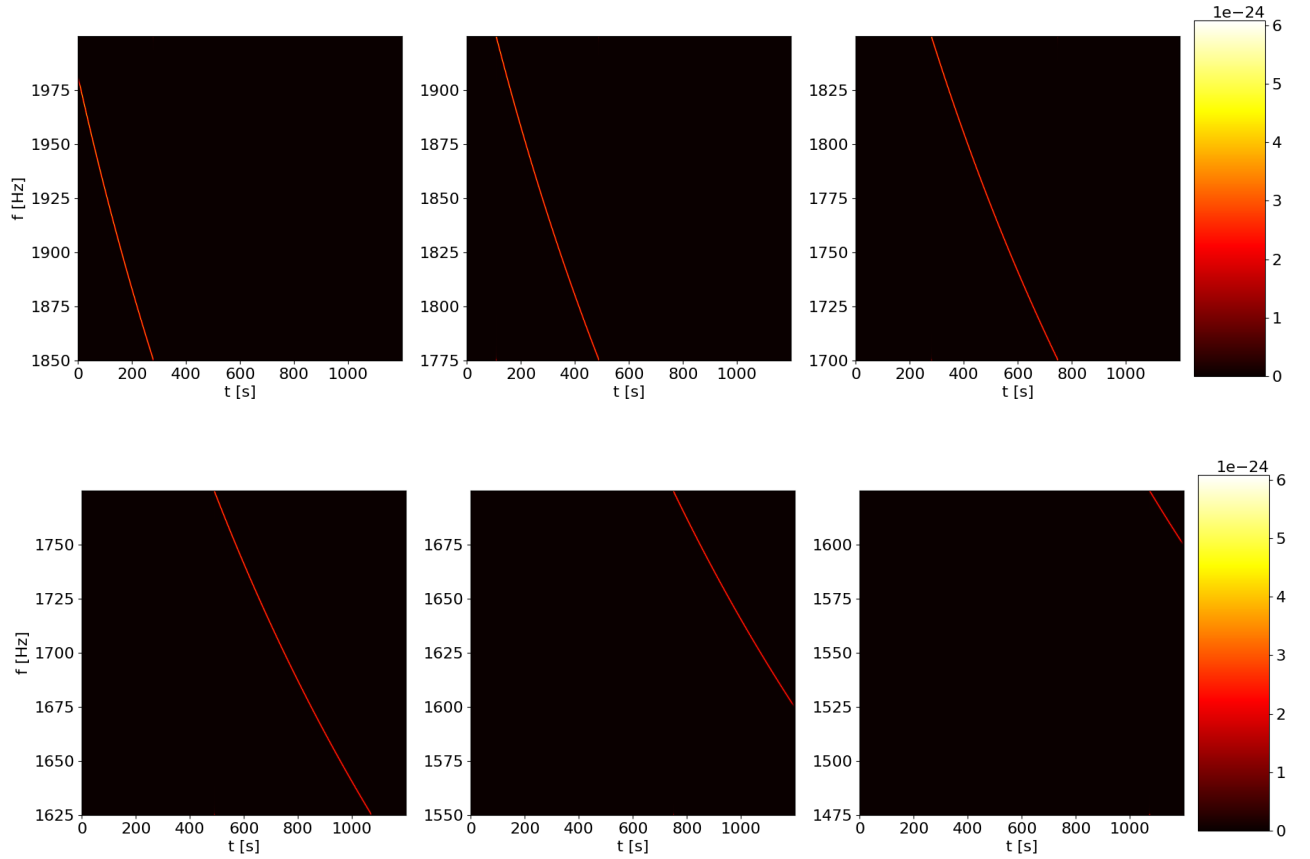


FIG. 2: Signal simulated with $f_0 = 1982$ Hz and $\epsilon = 0.0028$. Different frequency intervals, same time interval. The first, third and fifth maps cover the frequency range from 2075 Hz to 1625 Hz, while the second, fourth and sixth maps cover the frequency range from 2000 Hz to 1550 Hz. We count the maps starting with the one in the upper left angle and we move subsequently to the right.

the classifier. At lower amplitudes (using the O4 noise curve), our NNs are not able to discriminate between noise and signal.

To quantify the signal strength relative to noise, we compute the pixel-signal-to-noise ratio (pr_{sn}) as

$$pr_{\text{sn}} = \frac{1}{N_{\text{pix}}} \sum_{N_{tf} \neq 0} \frac{S_{tf}}{N_{tf}}, \quad (13)$$

where N_{pix} is the number of pixels in the map, N_{tf} is the noise amplitude spectral density at each map pixel, with time t and frequency f , and S_{tf} is the signal amplitude spectral density at the same map pixel.

C. Machine learning models

ML provides different approaches to solving complex problems in numerous areas [46]. The characteristics of the data and the specific problem define the best class of models for accomplishing the desired task. Deep learning (DL) is a specific technique to implement ML models that focuses on training multi-layered artificial neural networks (ANNs) to learn complex patterns and representations from data. DL is particularly well suited to tasks involving large amounts of data, such as “denoising” and “image classification”, because of its ability to automatically discover nonlinear patterns. In particular, we exploited CNNs [51] to build our deep NNs, specifically a signal denoiser and a classifier. The layer structure of the CNN is not densely connected, i.e., not all input nodes affect all output nodes. As a result, the number of parameters (weights) needed to characterize a layer is smaller than in a typical dense architecture, [52], which helps with high-dimensionality inputs such as the image data we use to represent the signal. For this reason, CNNs require less computational power to learn the characteristics of the data than other artificial neural networks. To implement our NN models, we used the Pytorch framework [53].

1. Denoiser

The goal of a denoiser, a specialized ANN, is to retrieve a clean image y from a noisy observation $x = y + \nu$, where ν is the noise. We adopted the residual learning approach (see [54] and Appendix A for details on our modified architecture). The model is closed by a skip connection³ that links the output to the input. The idea is to teach the model how to predict noise and then subtract it from noisy data images in order to produce a denoised map, i.e. the input for the classifier.

A supervised training approach was followed. As a loss function, i.e., a mapping from real events into a real number, we choose the mean square error:

$$MSE = \frac{1}{N_{\text{samples}}} \sum_{i=1}^N (\hat{y}_i - y_i)^2, \quad (14)$$

where N_{samples} is the number of samples, y_i is the predicted value and $\hat{y}_i$ the ground truth. Our goal is to minimize this quantity to find the best set of parameters.

An optimal denoiser should be able to reduce the noise level of the image without degrading the signal. To evaluate its ability to preserve the signal, we extended to two dimensions the waveform overlap used in [56]:

$$\mathcal{O} = \sqrt{\sum_{tf} h_{tf} h_{tf}^d \left(\sum_{tf} h_{tf} h_{tf} \right)^{-1}} \quad (15)$$

where $h_{tf} h_{tf}^d$ is the multiplication pixel by pixel of the signal map (the ground truth, h_{tf}) and the output of the denoiser (h_{tf}^d) while $h_{tf} h_{tf}$ is the multiplication pixel per pixel of the signal map with itself. The overlap values range from 0, when no track of the signal has been preserved, to 1, when the entire signal has been retrieved.

2. Classifier

An image classifier is a model that can distinguish different images that have different labels [57]. In particular, we are interested in a binary classifier: the two classes are the presence of a signal (positive class) and the absence of a signal (negative class). So to train our model, we used the binary cross entropy as a loss function (see [58] for more details).

The receptive field in the first layer, as shown in Appendix B, is bigger for the classifier than for the denoiser because we want to identify a correlation between distant pixels on the scale of the whole map. The signal differs from the noise because of this correlation, so this is a key feature that we want to extract to classify the maps.

The output of the NN is interpreted as the probability of the presence/absence of a signal. The samples that have been accurately predicted are either true positives (TP) or true negatives (TN). On the contrary, mismatched samples are known as false positives (FP) and false negatives (FN).

We studied the performance of the classifier, in particular the efficiency of detection (Eff) and the false alarm probability (FAP), adopting different probability thresholds to discriminate between the presence or absence of a signal in a given map. In principle, the Eff should be computed as

$$Eff = \frac{TP}{TP + FN}. \quad (16)$$

³ Technique used to link network input data directly to the last stages by skipping some branches ([55]).

Nevertheless, as discussed in subsection III B, a signal can cross multiple maps when its frequency varies very rapidly. We define a "positive trigger" to select a GW candidate when at least one map is correctly tagged. So, from here on, we will report the efficiency per signal (Eff), defined as the number of positive triggers over the total number of injected signals. For what it concerns the FAP, we computed it as

$$FAP = \frac{FP}{FP + TN + TP + FN}, \quad (17)$$

where the denominator is the sum of all the samples considered. Eventually, to obtain a good balance between Eff and FAP, we used the F1 score, [59], defined as:

$$F_1 = \frac{2}{\frac{1}{P} + \frac{1}{R}} = 2 \times \frac{P \times R}{P + R} = \frac{TP}{TP + \frac{FN + FP}{2}}. \quad (18)$$

where the *precision* P and the *recall* R are defined as

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN}. \quad (19)$$

F1 is a harmonic mean, so it gives more weight to lower values of P and R, and it is large only if both P and R are large. In particular, it is 0 if there are no true positives, i.e. $\text{Eff} \sim 0$ and $FAP \sim 1$, and 1 if there are no false predictions, i.e. $\text{Eff} \sim 1$ and $FAP \sim 0$. So we need to maximize the F1 score.

We use ROC curves to provide a visual representation of our classifier's performance.

IV. RESULTS

A. Dataset preparation

In order to train the two models, we built two sets of maps. Given that we were using two NNs at a time, the testing set of the first one (the denoiser) has been split to become the training and testing set of the second one (the classifier).

To train the denoiser, we simulated 1000 signals in the range described by Eq. (12) and 200 more with initial frequency f_0 in the frequency interval [1.8, 2.0] kHz. To test the denoiser and later train and test the classifier, we built another set of maps, with 2400 signals in the same intervals as Eq. (12). We trained and tested our models for different initial amplitudes, and we report here the results for $h_0(t_0) = 2 \times 10^{-23}$. For more information, see the Appendix C.

The two plots in Fig. 3 summarize the main properties of the denoiser training set. On the left panel, we report the number of maps crossed by each signal, and on the right panel, the pr_{sn} (Eq.(13)) of each signal versus the NS ellipticity (ϵ) and initial spin frequency (f_0). Each point represents a different signal. Notice that the

higher the initial frequency and the ellipticity, the higher the number of maps crossed by the signal. Indeed, as described in Eq. (4), the frequency in this case varies more rapidly. As a result, the signal loses energy faster, and pr_{sn} is lower (see the upper right corner of the parameter space). As discussed in the next subsection, the region where both f_0 and ϵ are large is where both denoising and classification become more challenging. The test set exhibits the same trend.

B. Denoising

We trained the denoiser as described in Sec. III C 1, and then evaluated the model's performance in reducing the noise and in preserving the signal separately. In particular, while the training was done with "noise plus signal" maps, the tests were carried out separately on "only noise" maps and on maps containing both noise *and* a signal.

We first focus on the noise reduction capability and summarize our results in Fig 4. The histograms on the left panel show the mean value of all pixels in each map of the test set ("noise only" maps) before (μ_{noise} , orange) and after (μ_{den} , blue) the denoiser, i.e.

$$\mu_{\text{noise}} = \sum_{t,f} \frac{1}{N_{pix}} N_{tf}^{\text{noise}} \quad \mu_{\text{den}} = \sum_{t,f} \frac{1}{N_{pix}} N_{tf}^{\text{den}} \quad (20)$$

where N_{tf} is the same as in Eq. (13). On the right panel, we show the ratio of the means, defined as

$$\mu_{\text{ratio}} = \frac{\mu_{\text{noise}}}{\mu_{\text{den}}} \quad (21)$$

as a function of f_{max} , i.e. the highest frequency of the time-frequency map.

We observe that after denoising, the mean values are reduced by up to two orders of magnitude and note a decreasing trend with frequency in the ratio. The reason for the latter is the dependence of the noise ASD on frequency. In fact, in the upper part of our frequency range, the noise ASD is higher (see, e.g., [60]), and denoising becomes more challenging.

We then turned to evaluating the effectiveness of the denoiser in preserving the signal by testing it on "noise+signal" maps. Given that, as discussed above, individual signals may appear in multiple maps, we report in Fig. 5 the best value of the overlap, $\mathcal{O}$ (Eq. (15)), in the $f_0 - \epsilon$ plane, i.e., the value of the overlap computed on the map where the signal has been denoised the best. Indeed, we stress that, in order to have a valid "signal trigger," it is sufficient to be able to tag right one map. We note that there is a correlation between the parameters of the signals and the ability of the denoiser to retrieve them. Indeed, the worst values of overlap are found in the upper right corner, the same region where pr_{sn} is lowest (right panel of Fig. 3), and the signal reaches the noise level more rapidly.

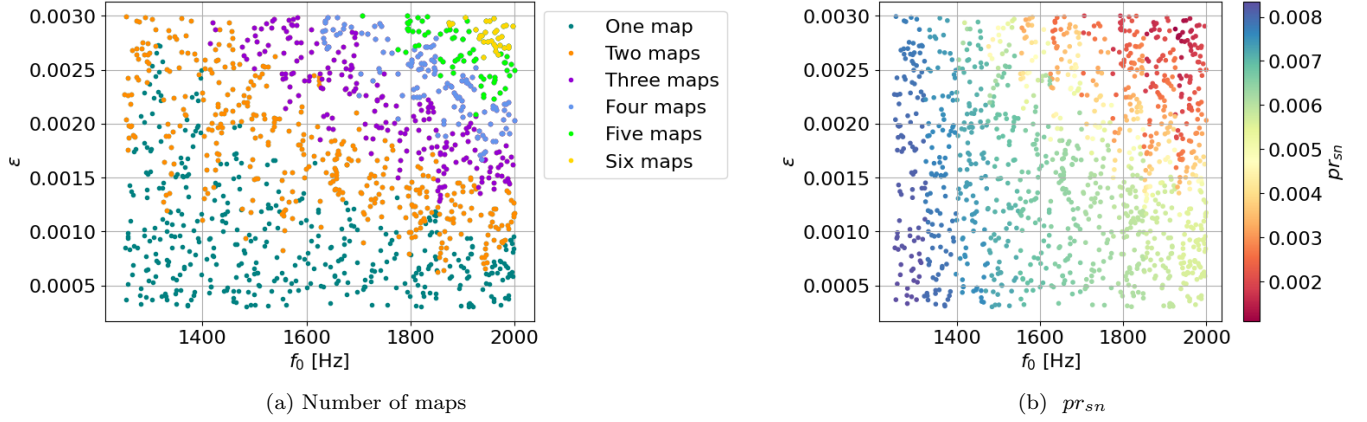


FIG. 3: For the denoiser training set, we plot, as a function of initial frequency and ellipticity, the number of maps crossed by the same signal (left panel), and the pixel-signal-to-noise-ratio (right panel).

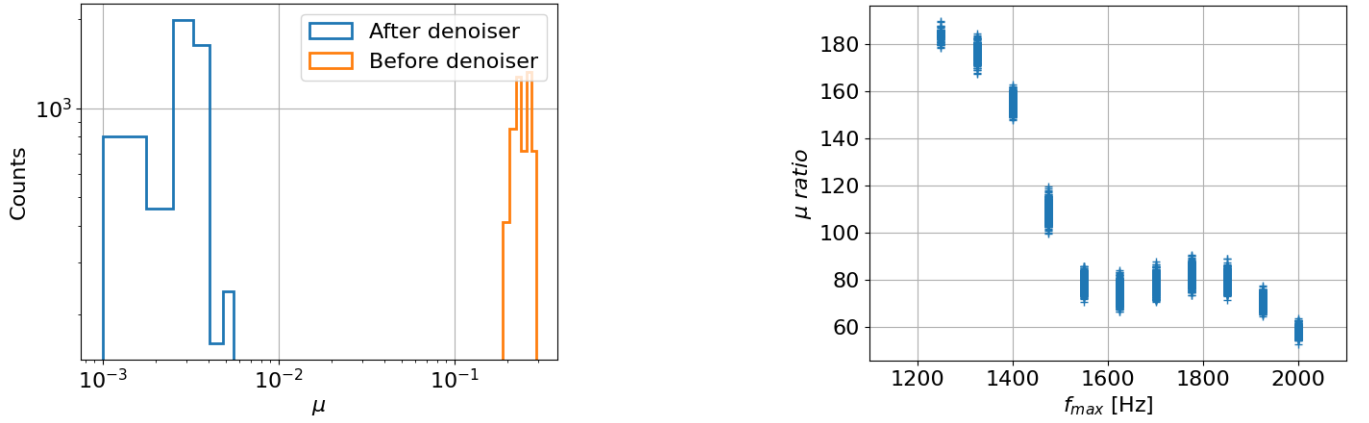


FIG. 4: Noise reduction evaluation. On the left-hand panel, we report the histograms of the ASD pixels mean for the testing set, before (orange) and after (blue) the denoiser. On the right-hand panel, we report the ratio between the mean of the maps that have not been passed through the denoiser to the mean of the denoised maps as a function of the highest frequency of each time-frequency map.

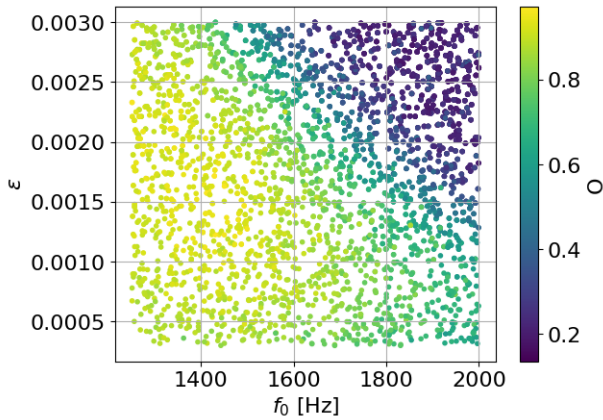


FIG. 5: Best overlap values for each set of maps, as a function of the value of frequency and ellipticity.

Despite this, the denoiser is capable of retrieving at least half of the signal for the majority of the parameter space. Overall, 76% of the signals have an overlap higher than 0.5.

C. Classification

Our goal is to distinguish maps where the signal is present from those where it is absent. The training procedure is applied to our classifier, described in Sec. III C 2, both on maps that were passed through the denoiser (clean maps) and maps that were not passed through the denoiser (noisy maps). In the latter case, the training of the classifier fails (for an initial amplitude of $2E-23$), i.e., the model does not find a set of parameters that minimize the training and validation losses. As

a consequence, the model does not learn the difference between maps with signals and maps that contain only noise. This demonstrates the crucial role of the denoiser in enhancing the chances of detection. We thus went on to only classify clean maps, using the F1 score (Eq. (18)) to find the best compromise between FAP and Eff.

In Fig. 6, we show the FAP, Eff and F1 score for different values of the probability threshold of the classifier. The fuchsia straight line is the value that maximizes the F1 score and corresponds to

$$\begin{aligned} \text{FAP} &= 4\% & \text{FAP}_{1800} &= 4\% \\ \text{Eff} &= 87\% & \text{Eff}_{1800} &= 94\% \end{aligned} \quad (22)$$

where FAP_{1800} is the FAP computed for noise-only maps whose frequency range is under 1800 Hz (blue crosses) and Eff_{1800} is the efficiency computed on signals with $f_0 < 1800$ (red crosses). As we would expect, Eff_{1800} is higher than Eff and decreases slower. Indeed, as shown in Fig. 5, as the frequency increases, it is more difficult to retrieve the signal. On the other hand, the FAP is not sensitive to the frequency and does not differ from FAP_{1800} .

In Fig. 7, we show the distance of the source associated with the retrieved signals as a function of f_0 and ϵ . As already stated, we consider a signal retrieved when we correctly tag at least one map that it crosses. Fewer points in the upper right corner of the maps show that these signals are more difficult to correctly label. These signals correspond to more distant sources. We expect this result, given the values the overlap assumes in that part of the parameter space (see Fig. 5). Anyway, we are able to classify sources distant up to 0.8 Mpc. This is still not enough, given the LIGO O4 sensitivity curve, because according to [61], at this distance, the supernova rate is lower than 0.1 per year. In addition, we expect magnetars to account for approximately 10% of the NS population [39], so the chance of having an event within these distances is low.

D. Denoiser improvements: masked loss

We need to increase the denoiser performance to reach better FAP or Eff. In particular, we need to improve our ability to retrieve the signal. During the training phase, we noticed that the loss decreased smoothly to a plateau. So in order to obtain better results, a change is needed, either in the NN structure or in the training procedure. A study of a diverse neural network pipeline will be the subject of future work. Here we report how we tried to highlight the structure of the signal and help the model learn it, using a masked loss. We note that the number of pixels in a single map (360 000) is three orders of magnitude higher than the number of pixels

that form the signal ($\mathcal{O}(600)$)⁴. This suggests modifying the training scheme through the use of a mask in order to improve the learning process. Initially, each map was multiplied by a matrix that had 1 on the signal pixels and 0 otherwise. As the training proceeded, using a gaussian blurring function, we enlarged the area of the map that was not set to zero, increasing the difficulty of the task. A more detailed explanation of visual attention methods, such as masked loss, can be found in [62]. In Fig. 8, we report an example of the output of the denoiser. In particular, on the left-hand panel there is a map containing noise plus a signal; on the center panel there is the signal we want to retrieve; and finally, on the right hand panel there is the output of the denoiser.

In Fig. 9, we report the ROC curves, obtained by the previous training technique. On the x-axis, there is the FAP, while on the y-axis, there is the Eff as a function of classifier threshold. The blue curve refers to learning without masked loss, i.e., what we have illustrated in Sec. IV C. The orange one is the best result that we have obtained after different trials with the masked loss, changing the parameters of the blurring. In Fig. 10 we report the FAP and Eff in the masked loss case (brown and cornflower blue x), compared with the same quantities as in Fig. 6. We observe that the values are better:

$$\text{FAP} = 2\% \quad \text{Eff} = 90\% \quad (23)$$

So the FAP decreases while the efficiency increases. At the same time, in Fig. 11 we report the distance of the source associated with the retrieved signals as a function of the parameter values. We see that even if the classifier identifies more signals, it still has problems in the upper right corner.

To compare with Eq. (22), we can choose a FAP of 4%. So adopting the masked loss training, we obtain a higher Eff, i.e.:

$$\text{FAP} = 4\% \quad \text{Eff} = 92\% \quad (24)$$

as shown in Fig. 10, in correspondence of the green vertical line. It is important to observe that we are conducting our analysis using data from a single interferometer. By combining data from a network of detectors, we expect to be able to lower the FAP. In fact, if we train our models separately on different interferometers data and then combine the results, we should lower our FAP down to the product of the different FAPs in the best-case scenario. This would allow us to choose a less stringent threshold to classify a map as having a signal.

⁴ As we have already stated in Sec. IIIB, we chose the size of the maps taking into consideration the properties of the signal we are interested in.

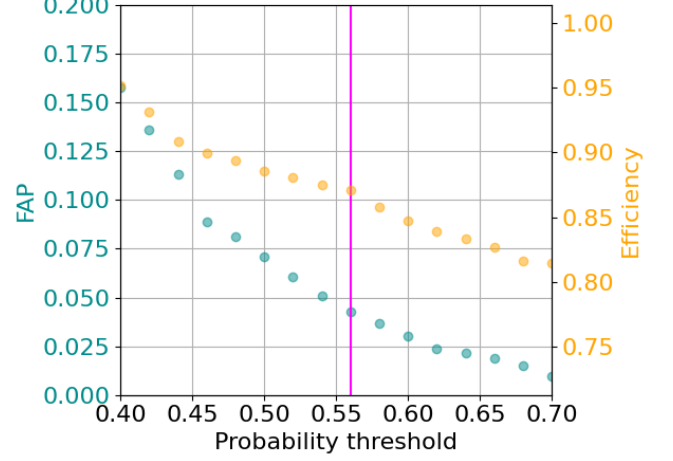
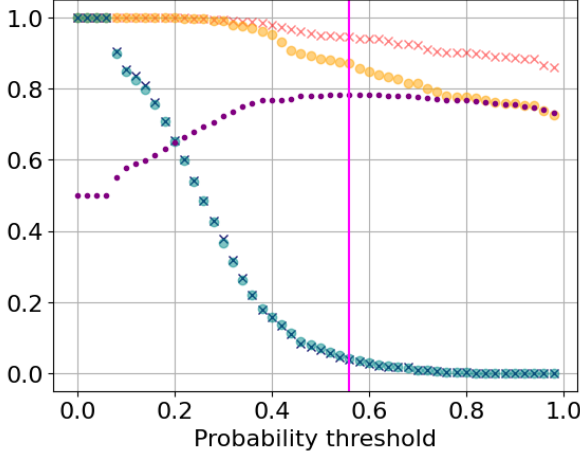


FIG. 6: On the left panel, efficiency per signal (orange), efficiency per signal with a maximum frequency of 1800 Hz (red x), FAP (cyan), FAP with a maximum frequency of 1800 Hz (blue x) and F1 score (purple) are shown for different values of the threshold to discriminate, in each map, among the presence or absence of a signal. The straight fuchsia line corresponds to the maximum of the F1 score. On the right panel, a zoom in the range of thresholds between 0.4 and 0.7 is shown, with the FAP on the left y-axis, and the efficiency on the right y-axis.

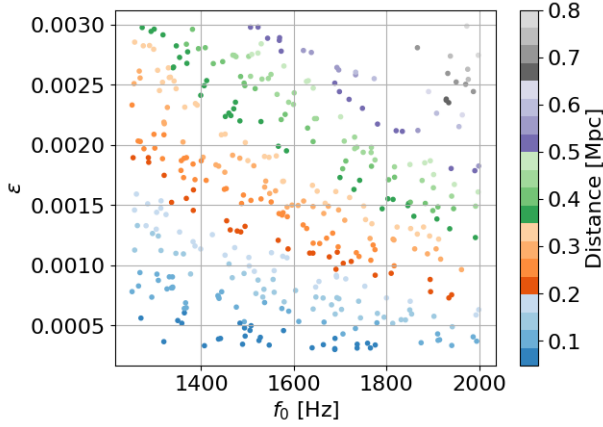


FIG. 7: Distance of the source associated to the retrieved signals as a function of frequency and ellipticity.

E. Changing the braking index

At the beginning of our work we trained our model assuming a braking index $n = 5$ (see equation (1)), that is, without taking into account the electromagnetic contribution to the spin-down (which is described, for instance, in [63]). To assess the robustness of the model, we tested its performance by injecting signals with different values of n , since our goal in the future is to generalize our technique by combining electromagnetic and gravitational torques.

So using the models already trained, we tried to test the denoiser and later train and test the classifier on

signals, with the same f_0 and ϵ , with n randomly taken in the range $[3.5, 5]$. It is important to be able to retrieve signals with different braking indexes in order to take into account the different emission processes of the star. In particular, we would like to have the capability of finding signals in regimes that differ from purely gravitational emissions. We did not train the denoiser again; rather we have used the best model of the previous subsection, trained with $n = 5$. Then, with the new denoised maps, we trained the classifier again. The idea is to show that even if we make a restrictive assumption, the model is able to retrieve different power laws from the one we trained it with. In Fig. 12, we show the new parameter space, where f_0 and ϵ are the same as the previous section, while we changed n . For what it concerns the fraction of the pixels of the signal which is preserved, in Fig. 13, we report the best value of the overlap for each set of maps as a function of initial frequency, ellipticity and braking index.

From the right-hand panel, we can notice that there is no trend in the braking index, in fact, the values of the overlap do not seem to depend on n . Nevertheless, on the left-hand panel, we notice the usual trend in f_0 and ϵ . However, the different braking index makes the classification procedure easier in the upper right corner of the parameter space while worsening the situation in the middle, as we can notice comparing with Fig. 5.

As a consequence, the classification task is more difficult. In Fig. 14, we report the values of efficiency, FAP and F1 score for different thresholds; the maximum of the F1 score (fuchsia line) corresponds to

$$\begin{aligned} \text{FAP} &= 1\% & \text{FAP}_{1800} &= 1\% \\ \text{Eff} &= 71\% & \text{Eff}_{1800} &= 67\%. \end{aligned} \quad (25)$$

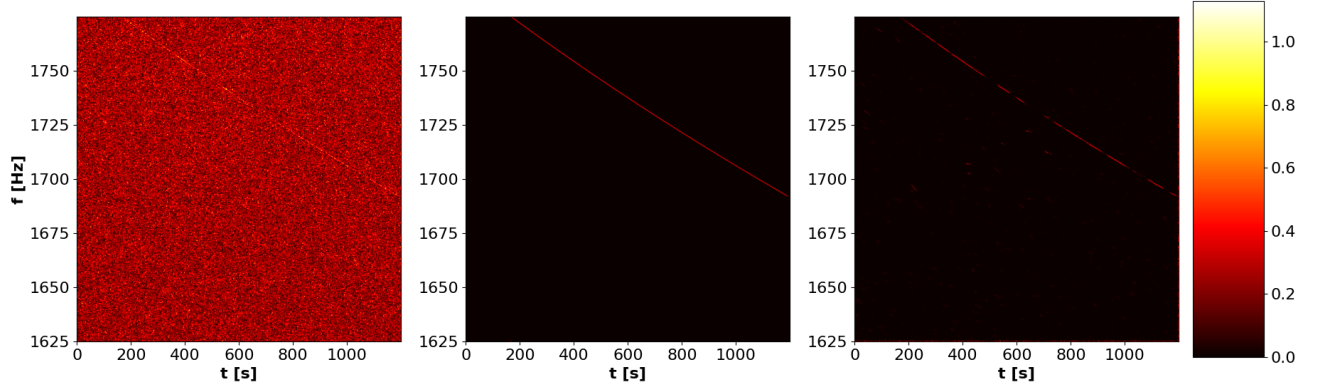


FIG. 8: Example of map containing noise plus signal, only signal and denoised noise plus signal ($\mathcal{O} = 0.58$). The physical parameters of the signal are $f_0=1791$ Hz, $\epsilon=0.001$, $h_0 = 2 \times 10^{-23}$. Each map has a frequency range between 1775 and 1625 Hz and covers a time interval of 1200 s. The noise is simulated according to the O4 design sensitivity curve of LIGO interferometers.

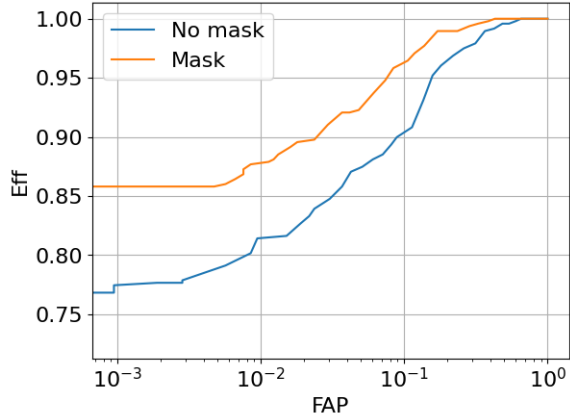


FIG. 9: ROC curves to compare the classifier performance on maps that have been denoised with a masked loss (blue) and the ones without it (orange).

in this case, the efficiency with a threshold on the initial frequency is slightly worse.

The efficiency is worse than (22, 23), but we gain a better FAP, so fixing the FAP at 4%, as in the first case Eq. (22), we obtain:

$$\begin{aligned} \text{FAP} &= 4\% & \text{FAP}_{1800} &= 4\% \\ \text{Eff} &= 73\% & \text{Eff}_{1800} &= 70\%. \end{aligned} \quad (26)$$

As in the previous case, Eq. (24), the efficiency increased.

We demonstrated that the method presented in this paper is robust to different values of n , ie. different power laws, even if we trained the denoiser with a fixed braking index.

F. Computational load

The total computing time for all the tasks described in this paper has been of the order of several hours. In particular, the creation of all the maps ($\mathcal{O}(7000)$) took about 6 hours. In addition, we needed 3 hours to train the denoiser with $\mathcal{O}(2000)$ maps and 20 minutes to denoise $\mathcal{O}(5000)$ maps. To train the classifier, 30 minutes were needed on $\mathcal{O}(4000)$ maps, and the classification task took less than 5 minutes on $\mathcal{O}(1000)$ maps. In other words, to train both models, we needed $\mathcal{O}(6000)$ maps of 1200 s that correspond up to 40 days of data. For more information, see Appendix C.

G. Discussion

In [30] a search for a tCW from a possible newly formed magnetar was conducted following the discovery of GW170817. In that case, different parameters were used for the simulation of the signal, in particular a moment of inertia that was three times bigger than the value we have used (at the limit of the mass ranges described in literature). As it is possible to notice in Eq. (7), the higher the moment of inertia, the higher the amplitude, and the easier it is to classify signals.

A fair comparison of our method to the methods employed in [30] is not straightforward because the considered parameter space is different in the two cases, and there are also differences in the assumed NS moment of inertia. However, we ran a code to estimate the optimal sensitivity of the *generalized Frequency Hough (GFH)* pipeline (see [25], Eq. (34)). We used the same signals, under the same simulated O4 noise. We obtain a maximum distance that is slightly less than twice that obtained by our method. It should be kept in mind, that this result is obtained under ideal conditions, in which the entire signal is fully integrated into the GFH calcula-

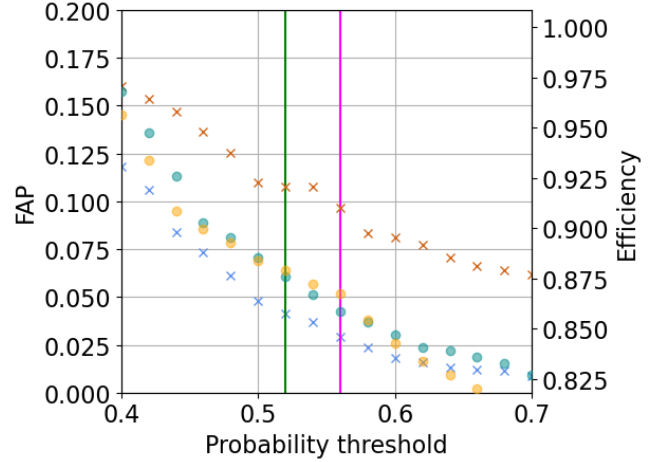
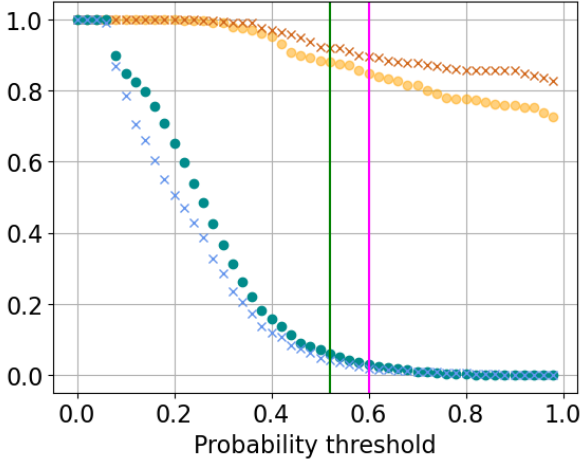


FIG. 10: On the left panel, efficiency per signal without masked loss (orange, as in Fig. 6), efficiency per signal with masked loss (brown x), FAP without masked loss (cyan, as in Fig. 6) and FAP with masked loss (cornflower blue x) are shown for different values of the threshold to discriminate, in each map, among the presence or absence of a signal. The straight fuchsia line corresponds to the maximum of the F1 score. The straight green line corresponds to FAP= 4 %. On the right panel, a zoom in the range of thresholds between 0.4 and 0.7 is shown, with the FAP on the left y-axis and the efficiency on the right y-axis.

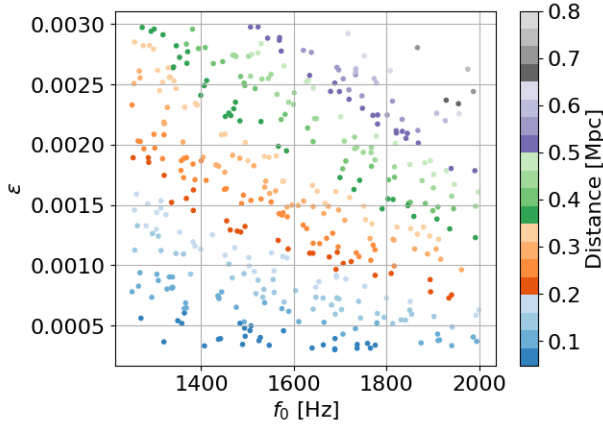


FIG. 11: Distance of the source associated to the retrieved signals as a function of frequency and ellipticity, masked loss.

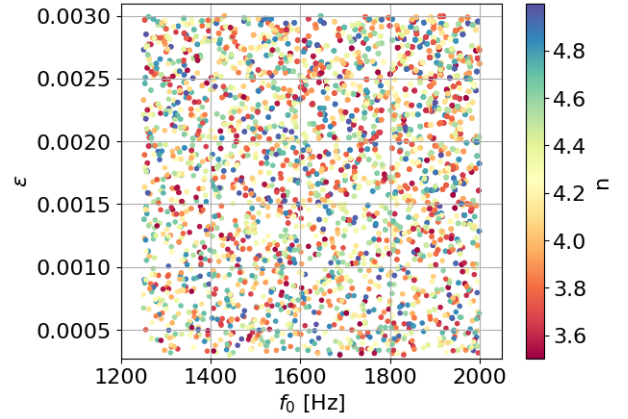


FIG. 12: Parameter space of the signals we have simulated, on the y-axis there is ϵ and on the x-axis f_0 , while on the colorbar there is n .

V. CONCLUSIONS

tion and the optimal data segment duration is used in all the parameter space. This approach is not realistically applicable for a general analysis in which signal parameters are not known a priori. A systematic comparison of these two methods sensitivities is beyond the scope of this article. Overall, we are confident that our technique will allow us to improve future searches to identify tCW candidates.

The goal of this study was to develop machine learning techniques for the search of tCW signals, such as those emitted by newborn magnetars, in interferometric data. We built, using a machine learning approach, a classifier to split time-frequency maps into two categories: presence or absence of signals. To help with the classification task, we built a denoiser.

To test the performance of our method, we simulated time-frequency maps containing noise plus signals. The noise was simulated according to the LIGO O4 design sensitivity curve, as explained in [48], and the signal

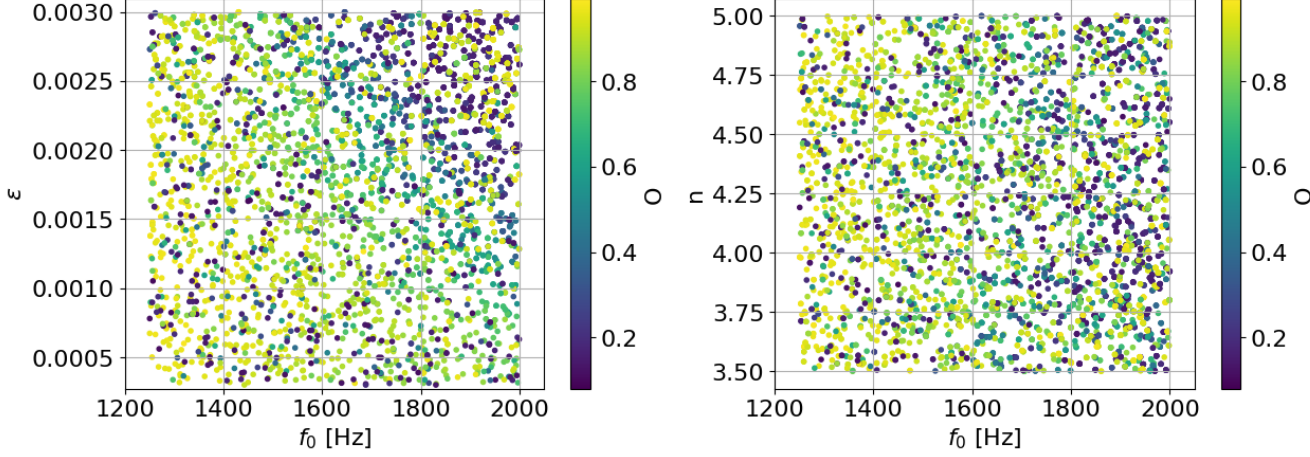


FIG. 13: Best overlap values, on the left-hand panel, as a function of the value of f_0 and ϵ and on the right-hand panel, as a function of f_0 and n .

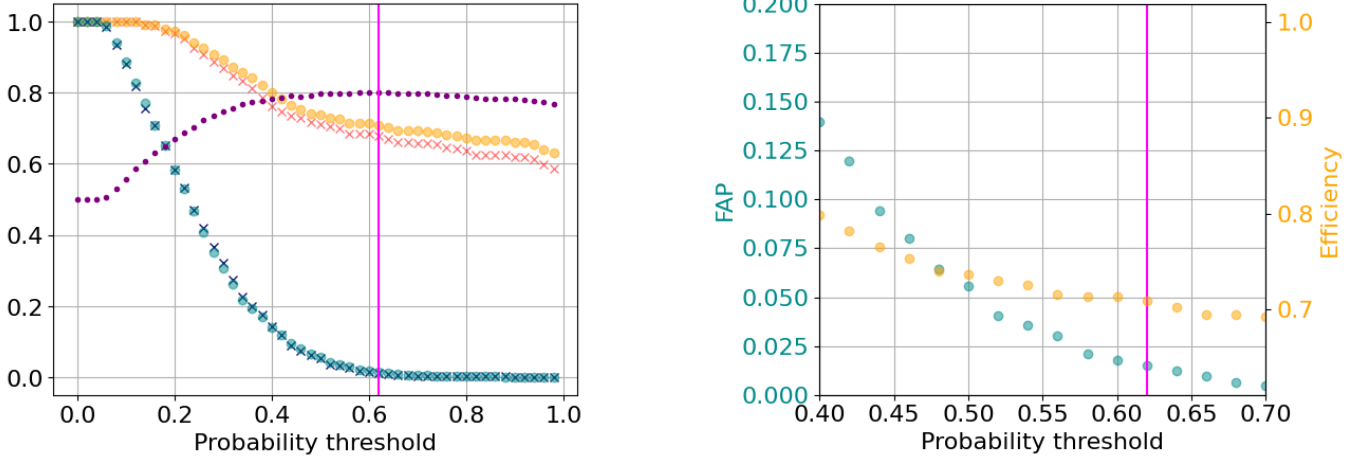


FIG. 14: Same quantities as in Fig. 6. The denoiser has been trained with the masked loss on signal with $n = 5$. The classified signals have $n \in [3.5, 5]$.

according to the rotating ellipsoid model (e.g. [45]) with $n = 5$.

Here we report our results for signals having an initial amplitude of 2×10^{-23} , the minimum value at which the use of the denoiser was beneficial to train the classifier.

We illustrate how this procedure can be used to search for tCWs emitted by newly born magnetars, and show that the denoiser plays a crucial role in the successful operation of the classifier. Indeed, we are not able to classify maps that have not been passed through the denoiser at an initial amplitude of 2×10^{-23} . Moreover, the procedure proved to be robust to different power laws with $n \in [3.5, 5]$, even if the denoiser was trained with $n = 5$.

An extremely relevant point is the rapidity of this

technique and the limited requirements in terms of computing power. In fact, we are able to analyze the equivalent of $\mathcal{O}(14)$ days of data in a few minutes.

In addition, 40 days of interferometer data should be sufficient to train the whole NN pipeline, and it is possible to use the rest of a scientific run to make inferences. Our method needs a number of maps to train the model that is about one order of magnitude smaller than in [33].

Overall, the proposed method has a sensitivity which is comparable with the already existing methods. In general, more sensitive and model-dependent methods can be devised to perform a follow-up of signal candidates and measure signal parameters.

In the future, we plan to do some further studies. In particular, we would like to find an efficient approach to combining the data of different interferometers and

conduct a deeper study on cases with a different braking index. At the same time, we aim at improving our denoiser by trying different NN architectures. Our goal, by combining these improvements with the new generation of detectors and the new observing run, is to be able to reach distances at which the probability of seeing this kind of event is significant.

ACKNOWLEDGMENTS

We thank our Sapienza Physics Department colleagues, S. Giagu and A. Ciardiello, for the useful discussion and suggestions. We also thank INFN and the Amaldi Research Center for the clusters hosted in the INFN Rome infrastructure, where we have stored the data used in this work and run the present analysis. We thank the INFN-CNAF computing staff for the resources we have used in this analysis and for their constant support. SD acknowledges funding by the European Union’s Horizon2020 research and innovation programme under the Marie Skłodowska-Curie (grant agreement No.754496).

Appendix A: Denoiser architecture

- First layer: 64 convolutional filters of size $3 \times 3 \times 1$ and ReLU as activation function, i.e. 1 input channel and 64 output channels.
- Second group of layers: six layers, each one with 64 convolutional filters of size $3 \times 3 \times 64$, i.e. 64 input channels and 64 output channels. Later, we have batch normalization and finally ReLU as an activation function.
- Last layer: 1 convolutional filter of size $3 \times 3 \times 64$, i.e. 64 input channels and one output channel.

Appendix B: Classifier architecture

- First layer: 5 convolutional filters of size $10 \times 10 \times 1$ and ReLU as activation function, i.e. 1 input channel and 5 output channels. The filters move with a stride of 5.
- Second layer: 10 convolutional filters of size $6 \times 6 \times 1$ and ReLU as activation function, i.e. 5 input channels and 10 output channels. The filters move with a stride of 3.
- Third layer: 1 max pooling layer with a kernel of size $5 \times 5 \times 1$.
- Fourth layer: one linear layer that reduces the dimension from 1690 to 84 and ReLU as an activation function.

- Last layer: a linear layer that passes from 84 numbers to 2 and softmax as activation function.

Appendix C: GPU usage and NN training

To speedup the NN training and inference times, we used an NVIDIA L40S GPU with 48GB of RAM, installed on the Virgo INFN Rome cluster. The denoiser for the first train, Sec. IV B, was trained for fifty epochs, with a batch size of 8. Beyond that, the loss did not show significant improvement. After the implementation of the masked loss, Sec. IV D, we trained the model for 200 epochs. In the first 160, we used masked loss, changing the width of the blurring function every 16 epochs. In the last 40 epochs, we trained it using the whole map while computing the loss function. As in the previous case, we used as batch size 8.

Concerning the classifier, we trained for 40 epochs with a batch size of 40 in the first two scenarios, Secs. IV C, IV D, and 60 epochs in the last one, Sec. IV E.

-
- [1] B. P. Abbott *et al.* (LIGO Scientific, Virgo), Observation of gravitational waves from a binary black hole merger, *Phys. Rev. Lett.* **116**, 061102 (2016).
- [2] R. Abbott *et al.* (KAGRA, VIRGO, LIGO Scientific), Gwtc-3: Compact binary coalescences observed by ligo and virgo during the second part of the third observing run, *Phys. Rev. X* **13**, 10.1103/physrevx.13.041039 (2023).
- [3] J. Aasi *et al.*, Advanced ligo, *Classical and Quantum Gravity* **32**, 074001 (2015).
- [4] F. Acernese *et al.*, Advanced virgo: a second-generation interferometric gravitational wave detector, *Classical and Quantum Gravity* **32**, 024001 (2014).
- [5] T. Akutsu *et al.*, Overview of KAGRA: Detector design and construction history, *Progress of Theoretical and Experimental Physics* **2021**, 05A101 (2020), <https://academic.oup.com/ptep/article-pdf/2021/5/05A101/37974994/ptaa125.pdf>.
- [6] M. Bailes *et al.*, Gravitational-wave physics and astronomy in the 2020s and 2030s, *Nature Rev. Phys.* **3**, 344 (2021).
- [7] O. J. Piccinni, Status and perspectives of continuous gravitational wave searches, *Galaxies* **10**, 10.3390/galaxies10030072 (2022).
- [8] V. Dergachev and M. A. Papa, Early release of the expanded atlas of the sky in continuous gravitational waves, *Phys. Rev. D* **109**, 022007 (2024), arXiv:2401.13173 [gr-qc].
- [9] L. D’Onofrio *et al.*, Search for gravitational wave signals from known pulsars in LIGO-Virgo O3 data using the 5n-vector ensemble method, *Phys. Rev. D* **108**, 122002 (2023), arXiv:2311.08229 [gr-qc].
- [10] R. Abbott *et al.* (LIGO Scientific, KAGRA, VIRGO), Model-based Cross-correlation Search for Gravitational Waves from the Low-mass X-Ray Binary Scorpius X-1 in LIGO O3 Data, *Astrophys. J. Lett.* **941**, L30 (2022), arXiv:2209.02863 [astro-ph.HE].
- [11] R. Abbott *et al.* (LIGO Scientific, KAGRA, VIRGO), Narrowband Searches for Continuous and Long-duration Transient Gravitational Waves from Known Pulsars in the LIGO-Virgo Third Observing Run, *Astrophys. J.* **932**, 133 (2022), arXiv:2112.10990 [gr-qc].
- [12] R. Abbott *et al.* (LIGO Scientific, Virgo, and KAGRA), All-sky search for continuous gravitational waves from isolated neutron stars using advanced ligo and advanced virgo o3 data, *Phys. Rev. D* **106**, 102008 (2022).
- [13] R. Abbott *et al.* (LIGO Scientific, Virgo, and KAGRA), Search for continuous gravitational wave emission from the milky way center in o3 ligo-virgo data, *Phys. Rev. D* **106**, 042003 (2022).
- [14] B. Steltner, M. A. Papa, H. B. Eggenstein, R. Prix, M. Bensch, B. Allen, and B. Machenschalk, Deep Einstein@Home All-sky Search for Continuous Gravitational Waves in LIGO O3 Public Data, *Astrophys. J.* **952**, 55 (2023), arXiv:2303.04109 [gr-qc].
- [15] V. Dergachev and M. A. Papa, Frequency-Resolved Atlas of the Sky in Continuous Gravitational Waves, *Phys. Rev. X* **13**, 021020 (2023), arXiv:2202.10598 [gr-qc].
- [16] R. Abbott *et al.* (LIGO Scientific, VIRGO), Search of the early O3 LIGO data for continuous gravitational waves from the Cassiopeia A and Vela Jr. supernova remnants, *Phys. Rev. D* **105**, 082005 (2022), arXiv:2111.15116 [gr-qc].
- [17] R. Abbott *et al.* (LIGO Scientific, VIRGO, KAGRA), Searches for Gravitational Waves from Known Pulsars at Two Harmonics in the Second and Third LIGO-Virgo Observing Runs, *Astrophys. J.* **935**, 1 (2022), arXiv:2111.13106 [astro-ph.HE].
- [18] R. Abbott *et al.* (LIGO Scientific, VIRGO, KAGRA), Search for continuous gravitational waves from 20 accreting millisecond x-ray pulsars in O3 LIGO data, *Phys. Rev. D* **105**, 022002 (2022), arXiv:2109.09255 [astro-ph.HE].
- [19] R. Abbott *et al.* (LIGO Scientific, Virgo, KAGRA), Constraints from LIGO O3 Data on Gravitational-wave Emission Due to R-modes in the Glitching Pulsar PSR J0537–6910, *Astrophys. J.* **922**, 71 (2021), arXiv:2104.14417 [astro-ph.HE].
- [20] Palomba, C., Gravitational radiation from young magnetars: Preliminary results, *A&A* **367**, 525 (2001).
- [21] C. Cutler, Gravitational waves from neutron stars with large toroidal b fields, *Phys. Rev. D* **66**, 084025 (2002).
- [22] S. Dall’Osso and L. Stella, *Millisecond Pulsars* (Springer International Publishing, 2021) p. 245–280.
- [23] A. Sur and B. Haskell, Gravitational waves from mountains in newly born millisecond magnetars, *Monthly Notices of the Royal Astronomical Society* **502**, 4680 (2021), <https://academic.oup.com/mnras/article-pdf/502/4/4680/36392225/stab307.pdf>.
- [24] K. Riles, Searches for continuous-wave gravitational radiation, *Living Reviews in Relativity* **26**, 10.1007/s41114-023-00044-3 (2023).
- [25] A. Miller *et al.*, Method to search for long duration gravitational wave transients from isolated neutron stars using the generalized frequency-hough transform, *Phys. Rev. D* **98**, 102004 (2018).
- [26] E. Thrane *et al.*, Long gravitational-wave transients and associated detection strategies for a network of terrestrial interferometers, *Phys. Rev. D* **83**, 083004 (2011).
- [27] L. Sun, A. Melatos, S. Suvorova, W. Moran, and R. J. Evans, Hidden markov model tracking of continuous gravitational waves from young supernova remnants, *Phys. Rev. D* **97**, 043013 (2018).
- [28] M. Oliver, D. Keitel, and A. M. Sintes, Adaptive transient hough method for long-duration gravitational wave transients, *Phys. Rev. D* **99**, 104067 (2019).
- [29] B. P. Abbott *et al.*, Search for post-merger gravitational waves from the remnant of the binary neutron star merger gw170817, *The Astrophysical Journal Letters* **851**, L16 (2017).
- [30] B. P. Abbott *et al.* (LIGO Scientific and Virgo), Search for gravitational waves from a long-lived remnant of the binary neutron star merger gw170817, *The Astrophysical Journal* **875**, 160 (2019).
- [31] B. Grace, K. Wette, and S. Scott, Gravitational wave searches for post-merger remnants of gw170817 and gw190425 (2024), arXiv:2403.11392 [gr-qc].
- [32] B. Grace, K. Wette, S. M. Scott, and L. Sun, Piecewise frequency model for searches for long-transient gravitational waves from young neutron stars, *Physical Review D* **108**, 123045 (2023).
- [33] A. L. Miller *et al.*, How effective is machine learning to

- detect long transient gravitational waves from neutron stars in a real search?, *Phys. Rev. D* **100**, 062005 (2019).
- [34] L. M. Modafferi, R. Tenorio, and D. Keitel, Convolutional neural network search for long-duration transient gravitational waves from glitching pulsars, *Physical Review D* **108**, 023005 (2023).
- [35] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, A survey of convolutional neural networks: Analysis, applications, and prospects, *IEEE Transactions on Neural Networks and Learning Systems* **33**, 6999 (2022).
- [36] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016) pp. 770–778.
- [37] O. Hamil, J. R. Stone, M. Urbanec, and G. Urbanová, Braking index of isolated pulsars, *Phys. Rev. D* **91**, 063007 (2015).
- [38] Z. F. Gao, X.-D. Li, N. Wang, J. P. Yuan, P. Wang, Q. H. Peng, and Y. J. Du, Constraining the braking indices of magnetars, *Monthly Notices of the Royal Astronomical Society* **456**, 55–65 (2015).
- [39] V. M. Kaspi and A. M. Beloborodov, Magnetars, *Annual Review of Astronomy and Astrophysics* **55**, 261–301 (2017).
- [40] B. Giacomazzo, J. Zrake, P. C. Duffell, A. I. MacFadyen, and R. Perna, Producing magnetar magnetic fields in the merger of binary neutron stars, *The Astrophysical Journal* **809**, 39 (2015).
- [41] B. Giacomazzo and R. Perna, Formation of stable magnetars from binary neutron star mergers, *The Astrophysical Journal Letters* **771**, L26 (2013).
- [42] P. Beniamini, K. Hotokezaka, A. van der Horst, and C. Kouveliotou, Formation rates and evolution histories of magnetars, *Monthly Notices of the Royal Astronomical Society* **487**, 1426 (2019), <https://academic.oup.com/mnras/article-pdf/487/1/1426/28755049/stz1391.pdf>.
- [43] P. D. Lasky, N. Sarin, and G. Ashton, Neutron star merger remnants: Braking indices, gravitational waves, and the equation of state, in *Xiamen-Custipen Workshop on the equation of state of dense neutron-rich matter in the era of gravitational wave astronomy* (AIP Publishing, 2019).
- [44] A. Corsi and P. Mészáros, GRB afterglow plateaus and Gravitational Waves: multi-messenger signature of a millisecond magnetar?, *Astrophys. J.* **702**, 1171 (2009), arXiv:0907.2290 [astro-ph.CO].
- [45] M. Maggiore, *Gravitational Waves. Vol. 1: Theory and Experiments* (Oxford University Press, 2007).
- [46] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).
- [47] S. Frasca, C. Palomba, R. Ruffato, and E. Majorana, Snag, a toolbox for gravitational wave data analysis, *International Journal of Modern Physics D* **09** (2012).
- [48] J. Timmer and M. Koenig, On generating power law noise., *Astronomy and Astrophysics* **300**, 707 (1995).
- [49] O4 design sensitivity curve - <https://dcc.ligo.org/ligo-t2000012/public>.
- [50] S. Dall’Osso, L. Stella, and C. Palomba, Neutron star bulk viscosity, ‘spin-flip’ and GW emission of newly born magnetars, *Monthly Notices of the Royal Astronomical Society* **480**, 1353 (2018).
- [51] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-dujaili, Y. Duan, O. Al-Shamma, J. I. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, Review of deep learning: concepts, cnn architectures, challenges, applications, future directions, *Journal of Big Data* **8** (2021).
- [52] C. M. Bishop and H. Bishop, *Deep Learning - Foundations and Concepts*, 1st ed., edited by S. Cham (2023).
- [53] Pytorch framework - pytorch.org.
- [54] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising, *IEEE Transactions on Image Processing* **26**, 3142 (2017).
- [55] G. Xu, X. Wang, X. Wu, X. Leng, and Y. Xu, Development of skip connection in deep neural networks for computer vision and medical image analysis: A survey, accessed: 2024-10-01.
- [56] P. Bacon, A. Trovato, and M. Beijer, Denoising gravitational-wave signals from binary black holes with a dilated convolutional autoencoder, *Machine Learning: Science and Technology* **4**, 035024 (2023).
- [57] D. Lu, A survey of image classification methods and techniques for improving classification performance, *International Journal of Remote Sensing* **28**, 823 (2007).
- [58] P. Shukla, How did binary cross-entropy loss come into existence?, accessed: 2023-05-01.
- [59] M. Sokolova and G. Lapalme, A systematic analysis of performance measures for classification tasks, *Information Processing & Management* **45**, 427 (2009).
- [60] B. P. Abbott *et al.*, A guide to ligo–virgo detector noise and extraction of transient gravitational-wave signals, *Classical and Quantum Gravity* **37**, 055002 (2020).
- [61] S. Ando, J. F. Beacom, and H. Yüksel, Detection of neutrinos from supernovae in nearby galaxies, *Physical Review Letters* **95**, 171101 (2005).
- [62] M. Hassanin, S. Anwar, I. Radwan, F. S. Khan, and A. Mian, Visual attention methods in deep learning: An in-depth survey, *Information Fusion* **108**, 102417 (2024).
- [63] S. Dall’Osso, B. Giacomazzo, R. Perna, and L. Stella, Gravitational waves from massive magnetars formed in binary neutron star mergers, *The Astrophysical Journal* **798**, 25 (2014).