Mass Determination of Cosmological Objects from Gravitational Wave Data Using Neural Networks

Blenda Úlima R. C. Guedes
Dept. of Statistics and Computing
UFRPE
Recife, Brazil
blenda.guedes@ufrpe.br

Antonio de Pádua Santos
Dept. of Physics
UFRPE
Recife, Brazil
antonio.padua@ufrpe.br

Tiago A. E. Ferreira
Dept. of Statistics and Computing
UFRPE
Recife, Brazil
tiago.espinola@ufrpe.br

***Abstract*—Gravitational waves were predicted by Albert Einstein more than a century ago, but only in 2015, the Laser Interferometer Gravitational-Wave Observatory (LIGO) was able to detect them. The gravitational wave phenomenon can be compared to spreading water from a lake after a stone has been thrown into it. Here, gravitational wave generation comes from an astronomical binary system formed by black holes or neutron stars. However, unlike the water, the amplitude of those gravitational waves is on a scale smaller than a proton's size. Despite this, we can describe it with simple equations in a phenomenological way. We can model those waves on a regular computer using post Newtonian physics. Here we were able to generate gravitational waves from computational simulations and make its data analyze. When a gravitational wave is detected in the real-world problem, there is a great interest in establishing the physical features of the astronomical bodies involved in the process. In this way, we propose applying a simple neural network to receive the gravitational wave data and infer information about the astronomical bodies' mass. The experimental results show that a simple neural network can extract mass information from the gravitational wave data. The recognition process proposed is much more straightforward than the complex computation based on numerical relativity for gravitational wave data analysis.**

***Index Terms*—Neural Networks, gravitational waves.**

I. Introduction

Gravitational waves are space-time vibrations predictable from Einstein's relativity theory. With tremendous implications about the understanding of the universe structure, the gravitational waves were first detected in 2015 by LIGO (Laser Interferometer Gravitational-Wave Observatory) [1]. Gravitational waves have been a subject that arouses the interest of the scientific community. It brings the possibility of listening to the universe in a completely different way, previously limited to the forms of electromagnetic waves.

LIGO Scientific Collaboration, within its large collaborative network, was responsible for the development of an extraordinary physical experiment able to measure displacements less than the proton's sizes, something about 10^{-21} meter [1]. Before, gravitational waves had been indirectly detected through the Hulse-Taylor Pulsar [2], which led them to receive the 1993 Nobel Prize for Physics [3]. One hundred years after the prediction of gravitational waves by Einstein's General Theory of Relativity, published in 1915 [4], LIGO succeeds in proving its direct detection, and in 2017 this contribution was gratified with a Nobel Prize in Physics [5]. Since that LIGO has detected 50 gravitational waves [6], [7].

Although LIGO has made available its data and even a library in Python [8], making a characterization of these waves through their signal is still an arduous task that requires much computational power. Therefore, several academic works try to invest in the use of machine learning, either to detect if the signal is a gravitational wave or to characterize a wave signal [9], [10].

We propose with this article show that with a simple computer, it is possible to generate simulations of gravitational waves and process those simulations to extract information from them using Machine Learning (ML), more specifically Artificial Neural Networks (ANN). Here, we are considering that gravitational wave generation comes from an astronomical binary system. This astronomical binary system is formed by two very massive bodies, like black holes. In this way, the gravitational waves bring information about the binary system, such as information about the black holes' masses. We want to determine if using a simple ANN to process the gravitational wave signal could infer the mass of the objects generating this wave.

The article is drawn as follows. In the next section, we describe a simple gravitational wave phenomenological model. In Section III we present a brief explanation of the Artificial Neural Networks approach that we used here. After that, in Section IV we evidence the process we used to create the database and make the data analysis. Finally, in Section V are presented the relevant remarks of the work.

II. Gravitational waves

According to Einstein's General Theory of Relativity, space and time ceased to be two rigid and separate entities. They came to be called space-time, a continuum of four dimensions. The presence of matter and energy can alter it, so objects of great mass in the Universe, such as planets, stars, and black holes, are able to bend it [11].

General Relativity also predicted the existence of Gravitational Waves, which is a phenomenon produced by every accelerated and not spherical object. Therefore the magnitude of these waves is minimum, on a scale of $10^{-21}m$ [11],

[12]. To have a fair comparison the size of a proton is $0.8768 \times 10^{-15} m$. Those waves also become weaker as far as they are from the source.

The event of Gravitational Waves can be compared to the propagation of waves in a lake after a stone has struck it. However, gravitational waves do not need a medium to propagate. Another fact is that different from electromagnetic waves, which are vibrations of electromagnetic fields, gravitational waves are space-time vibrations propagating with the speed of light [13].

Compared with electromagnetic waves, gravitational waves are extremely weak [13]. Thus, this type of phenomenon can only be observed in environments with large amounts of mass and energy [11]. Nowadays, we do not have enough resources to reproduce gravitational waves experimentally, making us look at the sky, searching for astronomic events capable of producing them. Those events are a collision of black holes, collision of neutron stars, supernova explosion, and gravitational waves from the Big Bang. This last one would give us information about the Universe before the formation of the atoms. Knowing these waves can come from this type of phenomenon, we are able to understand them better and hear what electromagnet events could not tell us until now [11], [13].

A. Gravitational waves of a binary system

As send before, phenomena like neutron stars and black holes collision are cosmological events able to produce gravitational waves within a significant magnitude (amplitude bigger than 10^{-21} meters). Such events were the only source that the LIGO was able to detect [1].

A binary system is composed of two objects. They can be a combination of black holes and neutron stars. Those objects meet in the cosmos and stay orbiting a common barycenter[1], among time they become closer and closer until they collapse together and become only one object [14]. It is important to say that these two objects orbit around each other for an undetermined amount of time. We only have data to evidence when they are close enough and on a very high velocity[2], to the point of producing gravitational waves with a significant scale.

This phenomenon can be described in three phases (Figure 1). The first one is the spiral phase, when both objects are orbiting each other at a very high speed, with fractions of the speed light [1]. This phase can be described phenomenologically by Equation 1. Where it seeks to define gravitational waveform of simple and circular binary system [14].

$$h(t) = \mathcal{A}(t) cos \Phi(t) \tag{1}$$

[1]Center of mass.

[2]The GW150914 detection achieved a velocity higher than 0.3 speed of the light [1]

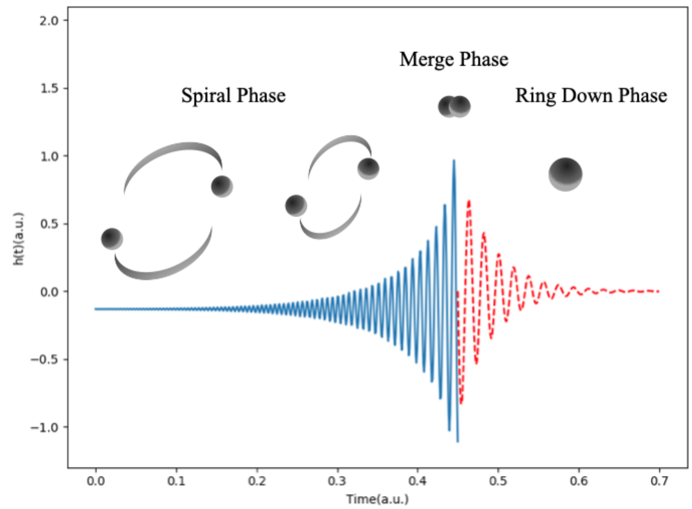

Fig. 1. Phases of a binary system. Blue continuous line - gravitational wave signal from the spiral phase Equation 1. Red dashed line - ring down stage Equation 8

Where $h(t)$ indicates the waveform, $\mathcal{A}(t)$ is the amplitude and $\Phi(t)$ is the wave phase. $A(t)$ and $\Phi(t)$ can be described as:

$$\mathcal{A}(t) = \frac{2(G\mathcal{M})^{5/3}}{c^4 r} \left(\frac{\pi}{P_{gw}(t)} \right)^{2/3} \tag{2}$$

Considering the Equation 1, it is possible to see that the fluctuation of the space-time metric depends only on time. Therefore $\Phi(t)$ can be written as the Equation 3.

$$\Phi(t) = \Phi_0 + 2\pi \int_0^t \frac{dt'}{P_{gw}(t')} \tag{3}$$

G is Newton's gravitational constant; c is the speed of light; r is the initial observation distance, in light-year; Φ_0 is the initial phase and P_{gw} is the period in the instant t. The value of $\mathcal{M}$ is called *"mass chirp"*, given this name, because it can be determined from the evolution of the signal received at the time *chirp* of the system's evolution [3]

$$\mathcal{M} \equiv (M_1 M_2)^{3/5} / (M_1 + M_2)^{1/5} \tag{4}$$

The period P_{gw} is easily related to the orbital period P_{orb}:

$$P_{orb}(t) = 2P_{gw}(t) \tag{5}$$

$$P_{orb}(t) = (P_0^{\frac{8}{3}} - \frac{8}{3} kt)^{\frac{3}{8}} \tag{6}$$

$$k \equiv \frac{96}{5} (2\pi)^{\frac{8}{3}} \left(\frac{G\mathcal{M}}{c^3} \right)^{\frac{5}{3}} \tag{7}$$

Observing the equations above, it is noticed that some input variables are determinants of the waveform. These are the

[3]Exact moment of approximation between the two objects and gradual increase of the frequency of the received signal.

initial orbital period (P_0), mass *chirp* ($\mathcal{M}$), the observation distance (r) and the initial phase of the wave (Φ_0).

The second phase is the moment of collapse, when objects unite and become one. The last one is the ring down phase, gravitational waves begin to fatigue until they return to low amplitude. At this part the waves can be represented phenomenologically as a damped harmonic oscillator [15], see bellow:

$$h(t) = Be^{\frac{-t}{\tau}} \sin(\omega t + \delta_0) + D \tag{8}$$

where B and D are constants, δ_0 is an initial phase, and τ is the damping constant. It can be seen as the red dashed line in Figure 1.

III. Neural Networks

Mathematically an Artificial Neural Network is a universal function approximator [16]. They work as a parallel composed of simple units that calculate a specific mathematical function. These units are distributed in one or more layers so that they produce a large number of connections [17].

Starting from the idea of biological neural network, in 1943 McCulloch and Pitts created a mathematical model of a Perceptron, which attempts to copy the structure of a neuron [18]. A perceptron can receive several *entries* $(x_1, ..., x_m)$ which the relevance of those entries are made by *weights* $(w_1, ..., w_m)$. In mathematical form, one can describe a neuron in a function of its index k (Equation 9).

$$u_k = \sum_{j=1}^{m} w_{kj} x_j \tag{9}$$

The Equation 9 also includes *bias* (b_k) which has the purpose of increasing or decreasing the value of the download function. That is, the value of the function is negative or positive.

$$y_k = \varphi(u_k + b_k) \tag{10}$$

On Equation 10 u_k is the *linear combination* of the input values (x). Already φ is an activation function and y_k is output from the function.

There is a variety of activation functions, the Equations 11, 12 and 13 features the rectified linear unit (ReLU) function, Hyperbolic Tangent function and Sigmoid function [19].

$$f(x) = x^+ = max(0, x) \tag{11}$$

$$f(x) = tanh(x) \tag{12}$$

$$f(x) = \frac{1}{1 + e^{-x}} \tag{13}$$

One type of ANN is a Multilayer Perceptron (MLP), it uses a set of perceptron and direct feed[4]. Its basic structure is

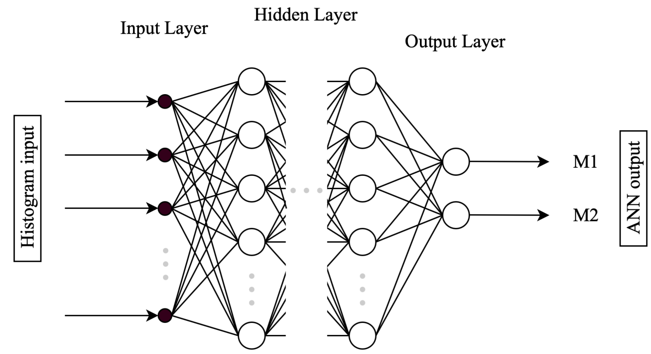

Fig. 2. Architecture of a Multilayer Perceptron Artificial Neural Network.

called in three points: each neuron has a function of activation nonlinear and differentiable; there are one or more layers (see Figure 2) intermediate or hidden for both input and output layers; and has a high number of connections where their amplitudes are calculated by the weight of each of the connections [20].

The input and output layers work similarly to a perceptron, which has their connections pointed at more than one neuron. The intermediate layers work with a network of perceptrons, having exits of a neuron in any layer with the entrance of the neurons of the following layer. Finally, the output layer delivers the result of the network [18].

To adjust the weights of the network connections, we use the backpropagation technique. This method uses as a parameter the network prediction error propagating this error retroactively, so it receives this name [18]. The function used to make this adjustment is called Optimizer. Among the existing ones, the Stochastic Gradient Descent (SGD) has evidenced itself as an efficient optimizer used in many machine learning problems. Furthermore, there's the Adam optimizer, which only requires first-order gradients and consumes less computing power [21].

IV. Methodology

In this section, we present the steps of the experiments developed in this research. The machine used in the development of the experiments was a MacBook Pro, with a Mac OS Catalina operating system, $500Gb$ SSD, and RAM $16Gb$. We used Python 3.6 and the PyTorch library to the ML experiments [22].

A. Building the database

Within the aim to obtain the mass values from the wave signal generated by the gravitational wave phenomenal, we generated our database using the method described on II-A. The next step for constructing the base was to determine the mass of the objects to be simulated. Therefore, only the masses were variable during the simulation, keeping all other input values quoted previously constant. Thus, the data set dependent only on these two input mass values (M_1 and M_2). Therefore, a collection of variants was created with a range

[4]The input layer values are projected only in their direct sense to the layer of exit, and never the other way around [20].

between $20M\odot$[5] and $40M\odot$ and a $0.1M\odot$ step. These values were chosen because they attend most of LIGO detection range of mass [6], [7].

It was also necessary to ensure that all waves were from different masses so that there was no duplication of waves in the database. The Algorithm 1 shows the exact implementation to check the no duplication of the *chirp mass*.

ALGORITHM 1. Generate *chirp mass*

```
sm = 1.989e30 #Sun mass

for m1 in range(20,40,0.1):
  for m2 in range(m1,40,0.1):
    chirp_mass = calculate_chirp_mass
                 (m1*sm, m2*sm)
    if chirp_mass not in chirp_mass_list:
      chirp_mass_list.append(chirp_mass)
```

Thus, it was possible to obtain $80,200$ masses relations, and we ensured that there was no repetition, which leads to the same quantitative on waves. As said in II-A the waves generated here are finished at the exact moment of the objects merge.

The pre-set values for the other variables were 0.1 for the initial period (P_0), 0 for the initial phase (Φ_0) and $410,000kcp$ for luminous distance (r), approaching from the distance of $GW140915$ [1]. The Algorithm 2 displays the Python code used to generate the waves.

ALGORITHM 2. Wave generator

```
def wave_generator(chirp_mass,
                   luminosity=1.0):

    c = 3.0e8 #light speed
    G = 6.674e-11 #Newton's gravitational constant
    k = (96/5)*(2*math.pi)**(8/3)*
        (G*chirp_mass/c**3)**(5/3)
    r = luminosity
    x = 2.0*(G*chirp_mass)**
        (5/3)/(r*c**4)
    P0 = 0.5 #initial orbital period
    fi = 0.0 #initial wave's phase
    dt = 0.001 #time frame interval
    t = 0.0
    h = 0.0
    strain = []

    while t < (3*P0**(8/3)/(8*k)):
        strain.append(h/1e23)
        pgw = 0.5*(P0^(8/3)-(8/3)*k*t)^(3/8)
        amp = x*(math.pi/pgw)^(2/3)
        phi = fi-(4.0/5.0)*math.pi*
              (3.0*P0^(8/3)-8.0*k*t)/
              (k*(P0^(3/8)-(8/3)*k*t)^(3/8))
        h = h + amp * math.cos(phi)
        t = t + dt
    return strain
```

After all waves have been generated, it was possible to see that the greater the mass was more points the wave had (Figure 3). Since the time step is fixed, the wave lasts longer when the mass has increased, which made the waves have more

[5] $M\odot$ represents the solar mass, $1.989 \times 10^{33}g$.

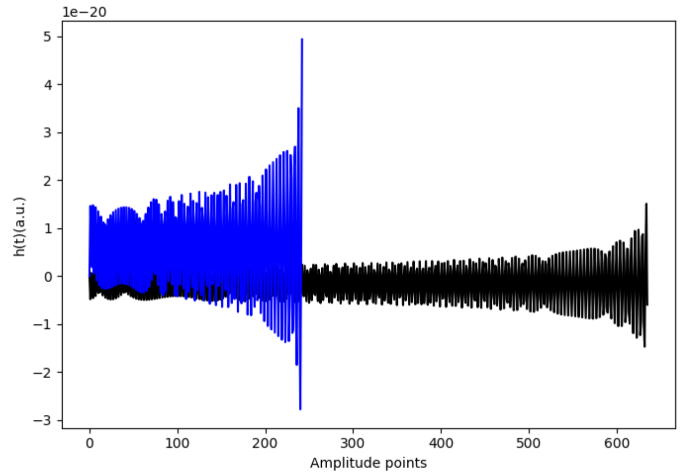

Fig. 3. Different sizes waveform. The smaller and blue was generated with $20M\odot$ and $22M\odot$ objects. The bigger and black was generated with $40M\odot$ and $35M\odot$ objects.

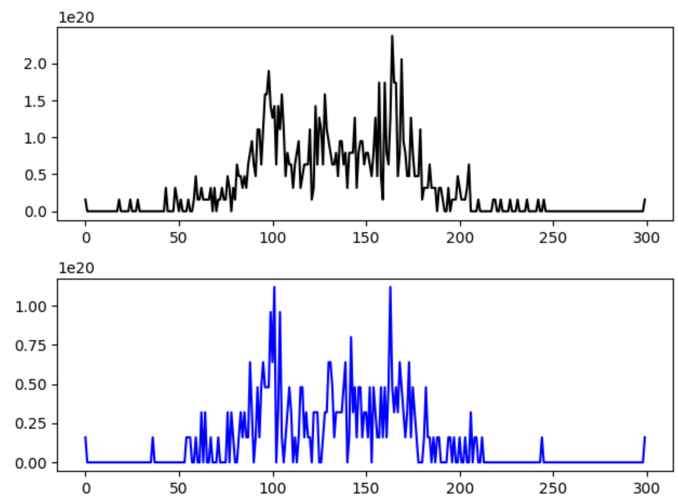

Fig. 4. Waves's Histogram. On the top on black is the histogram of the wave generated with $20M\odot$ and $22M\odot$ objects. On the bottom on blue is the histogram of the wave generated with $40M\odot$ and $35M\odot$ objects.

amplitude points. To manage this situation, the wave's histograms (Figure 4) were used as the network input. That way, all waves are maintained with an equivalent characterization, and the network can be trained because all the input data have the same size of 300 points.

B. Search for the best Neural Network

For analyze the generated data we use Neural Networks, described in III, first we tried to find the best combination of the network architecture (disposed in Table I) and its learning rate, where we used: $1e10^{-1}$, $1e10^{-2}$ and $1e10^{-3}$. We applied the Grid Search technique, where a set of hyperparameters is disposed and combined, then we try every possible combination to find the one that leads to the best performance of the algorithm. All the networks used have 300 nodes as input

TABLE I
NEURAL NETWORKS ARCHITECTURE CONFIGURATION

Number of hidden layer	2, 3, 5, 10
Number of node on the hidden layer	50, 100, 150
Activation function	ReLU, Sigmoid, Tahn

TABLE II
THE TOP 10 NEURAL NETWORK PERFORMANCE WITH 1000 EPOCHS

Hidden Layer	**Hidden Layer Units**	**Activation Function**	**Learning Rate**	**Test Score**	**σ of Difference**
3	150	ReLU	$1e10^{-2}$	4.437	4.438
2	150	ReLU	$1e10^{-2}$	4.438	4.438
3	100	ReLU	$1e10^{-2}$	4.438	4.438
2	100	ReLU	$1e10^{-2}$	4.441	4.441
5	100	ReLU	$1e10^{-2}$	4.443	4.443
2	50	ReLU	$1e10^{-2}$	4.443	4.443
3	50	ReLU	$1e10^{-2}$	4.443	4.448
5	100	ReLU	$1e10^{-3}$	4.449	4.449
2	50	ReLU	$1e10^{-3}$	5.458	5.458
3	150	ReLU	$1e10^{-3}$	5.515	5.515

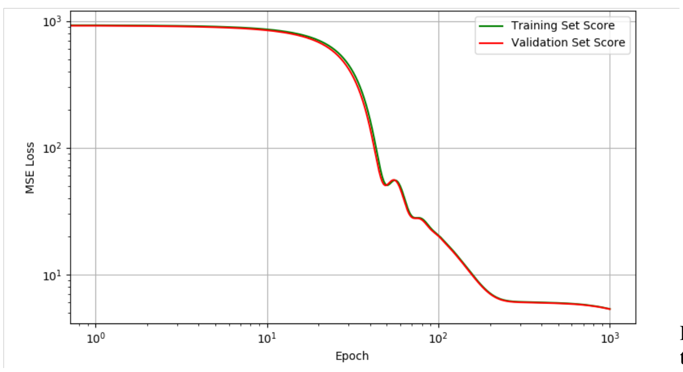

Fig. 5. Loss Graph of the 9^{th} neural network on the ranking, in a logarithmic view. The network has 2 hidden layers with 50 nodes each, used ReLU as activation function and $1e10^{-3}$ as the learning rate.

layer and 2 nodes as output layer to bring us the approximated mass of M_1 and M_2.

To train the networks, we also applied stationary configurations for all the networks searched. We used the Adam optimizer with the default values provided by the PyTorch library, and the mean squared error as loss function [22].

Proceeding with the experiment, we applied a fixed seed of value 2. We separated the database in a training set with 70%, validation and test each one with 15% of the data. We first ran the Grid Search with 1000 epochs for each combination and selected the best 10 networks according to the test set score (Table II).

After this phase of the experiments, we could notice that some loss graphs were still decaying (Figure 5). We stored the networks with their exact states and weight values. We proceed to more 4000 epochs only applied on the top 10 networks. They were having a total of 5000 epochs of training for each network.

With the new experiment, we also considered the training

TABLE III
THE TOP 10 NEURAL NETWORK PERFORMANCE WITH 5000 EPOCHS

Hidden Layer	**Hidden Layer Units**	**Activation Function**	**Learning Rate**	**Test Score**	**σ of Difference**
3	100	ReLU	$1e10^{-2}$	4.439	4.440
5	100	ReLU	$1e10^{-2}$	4.441	4.441
5	150	ReLU	$1e10^{-3}$	4.443	4.443
2	50	ReLU	$1e10^{-3}$	4.446	4.445
5	100	ReLU	$1e10^{-3}$	4.455	4.455
3	50	ReLU	$1e10^{-2}$	4.456	4.451
2	50	ReLU	$1e10^{-2}$	4.457	4.451
2	150	ReLU	$1e10^{-2}$	4.463	4.463
3	150	ReLU	$1e10^{-2}$	4.465	4.455
2	100	ReLU	$1e10^{-2}$	4.499	4.452

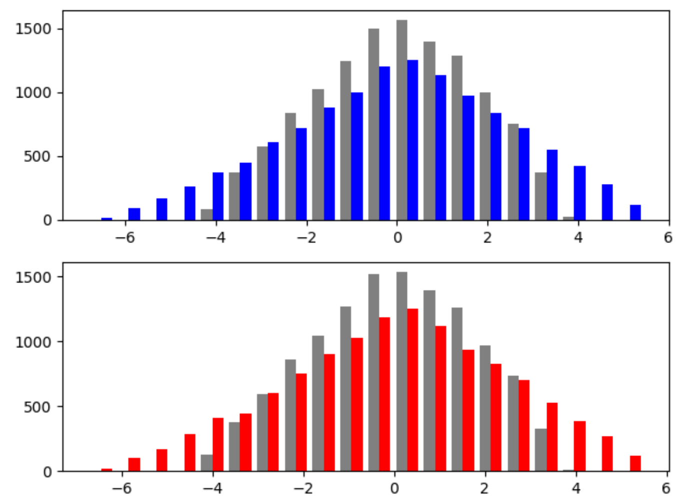

Fig. 6. Histogram of the difference between target and prediction. On the top, we can see the one trained with 1000 epochs, both neural network configurations differ on the number of hidden layers quantitative, the blue one has 150 nodes, the grey lines represent M_1, and blue are M_2. On the bottom we have a 100 nodes on the hidden layers, the grey lines are the representation of M_1, and red are M_2. Both distributions are very similar, in visualization and in metric values.

score to classify the best performance, which made the table change (Table III). We also noticed that all the scores are under 5 now. The distribution of the real and predicted values can be seen on Figure 6, where can visualize a flatness of the masses values through prediction process. Therefore, we had extra work to build this Table III, the networks that were on the top of their performance did not have an improvement.

V. CONCLUSIONS

Many efforts has been done for many research groups to understand the information about the universe. Laboratories like LIGO developed a huge experimental setup to be able to measure essential space-time vibrations, the gravitational waves. The LIGO computational data analyze required to extract physical information about the astronomical bodies involved in the gravitational wave observations is a very hard task. Many techniques based on numerical Relativity procedures are commonly employed to analyze of those LIGO data, where a large computational infra-structure is necessary [23]–[25].

The article proposed the use of a non-complexity neural network to characterize gravitational waves signals. We modeled our database and processed the signal. After analyzing the proposed methodology, we were able to have the results within neural networks and variance of 4.43, which leads us to a margin of error of 2.10. Our initial mass range is 20 since we vary between 20 and 40 solar masses. This brings us to about 10.5% error on the values of the predicted masses.

Therefore, even with reduced computational power, it was possible to model and analyze gravitational waves and reach a reasonable margin of error. At this article we modeled the data, furthermore the results achieved here can be expanded with real data from LIGO, which could bring a second phase of the project.

References

[1] B. P. Abbott, et. al., Observation of gravitational waves from a binary black hole merger, Phys. Rev. Lett. 116 (2016) 061102. doi:10.1103/PhysRevLett.116.061102.
URL https://link.aps.org/doi/10.1103/PhysRevLett.116.061102

[2] R. A. Hulse, J. H. Taylor, Discovery of a pulsar in a binary system., apjl 195 (1975) L51–L53. doi:10.1086/181708.

[3] N. Foundation, The nobel prize in physics (2020).
URL https://www.nobelprize.org/prizes/physics/1993/press-release

[4] T. Sauer, Albert Einstein's 1916 Review Article on General Relativity, arXiv e-prints (2004) physics/0405066arXiv:physics/0405066.

[5] N. Foundation, The nobel prize in physics (2020).
URL https://www.nobelprize.org/prizes/physics/2017/press-release/

[6] B. Abbott, R. Abbott, T. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, R. Adhikari, V. Adya, C. Affeldt, et al., Gwtc-1: A gravitational-wave transient catalog of compact binary mergers observed by ligo and virgo during the first and second observing runs, Physical Review X 9 (3) (Sep 2019). doi:10.1103/physrevx.9.031040.
URL http://dx.doi.org/10.1103/PhysRevX.9.031040

[7] R. Abbott, T. Abbott, S. Abraham, F. Acernese, K. Ackley, A. Adams, C. Adams, R. Adhikari, V. Adya, C. Affeldt, et al., Gwtc-2: Compact binary coalescences observed by ligo and virgo during the first half of the third observing run, Physical Review X 11 (2) (Jun 2021). doi:10.1103/physrevx.11.021053.
URL http://dx.doi.org/10.1103/PhysRevX.11.021053

[8] Rich Abbott, Thomas D. Abbott, Sheelu Abraham, Fausto Acernese, Kendall Ackley, Carl Adams, Rana X. Adhikari, Vaishali B. Adya, Christoph Affeldt, Michalis Agathos, et al., Open data from the first and second observing runs of advanced ligo and advanced virgo, SoftwareX 13 (2021) 100658. doi:10.1016/j.softx.2021.100658.
URL http://dx.doi.org/10.1016/j.softx.2021.100658

[9] G. R. Santos, M. P. Figueiredo, A. de Pádua Santos, P. Protopapas, T. A. E. Ferreira, Gravitational wave detection and information extraction via neural networks (2020). arXiv:2003.09995.

[10] L. Haegel, S. Husa, Predicting the properties of black-hole merger remnants with deep neural networks, Classical and Quantum Gravity 37 (13) (2020) 135005. doi:10.1088/1361-6382/ab905c.
URL http://dx.doi.org/10.1088/1361-6382/ab905c

[11] B. F. Schutz, A First Course in General Relativity, 2nd Edition, Cambridge University Press, 2009.

[12] M. P. Hobson, G. P. Efstathiou, A. N. Lasenby, General Relativity An Introduction for Physicists, 1st Edition, Cambridge University Press, 2006.

[13] LIGO Laboratory, Why detect them? (2015).
URL https://www.ligo.caltech.edu

[14] L. J. Rubbo, S. L. Larson, M. B. Larson, D. R. Ingram, Hands-on gravitational wave astronomy: Extracting astrophysical information from simulated signals, Utah State University DigitalCommons@USUs (Aug. 2007). doi:10.1119/1.272158.

[15] J. B. Marion, S. T. Thornton, Classical dynamics of particles and systems, 5th Edition, Cengage Learning, 2003.

[16] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016, http://www.deeplearningbook.org.

[17] A. Braga, T. Ludemir, A. Carvalho, Redes Neurais Artificiais - Teoria e aplicação, 1st Edition, Livros técnicos e científicos editora, 2000.

[18] C. M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, Inc., New York, NY, USA, 1995.

[19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[20] S. Haykin, Neural Networks and Learning Machines, 3rd Edition, Pearson Education, Inc., 1999.

[21] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization (2017). arXiv:1412.6980.

[22] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, Pytorch: An imperative style, high-performance deep learning library, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems 32, Curran Associates, Inc., 2019, pp. 8024–8035, http://papers.neurips.cc/paper/9015-pytorch- an-imperative-style-high-performance-deep-learnin g-library.pdf.

[23] G. R. Santos, M. P. Figueiredo, A. de Pádua Santos, P. Protopapas, T. A. E. Ferreira, Gravitational wave detection and information extraction via neural networks (2020). arXiv:2003.09995.

[24] R. De Pietri, A. Feo, J. A. Font, F. Löffler, M. Pasquali, N. Stergioulas, Numerical-relativity simulations of long-lived remnants of binary neutron star mergers, Phys. Rev. D 101 (2020) 064052. doi:10.1103/PhysRevD.101.064052.
URL https://link.aps.org/doi/10.1103/PhysRevD.101.064052

[25] N. E. M. Rifat, S. E. Field, G. Khanna, V. Varma, Surrogate model for gravitational wave signals from comparable and large-mass-ratio black hole binaries, Phys. Rev. D 101 (2020) 081502. doi:10.1103/PhysRevD.101.081502.
URL https://link.aps.org/doi/10.1103/PhysRevD.101.081502