Convolutional neural network regression to estimate the mass parameter of astrophysical binary black hole systems

Andres Antelis [*1] **Claudia Moreno** [*2]

Abstract

In this paper we propose the use of a deep learning based model for inferring astrophysical information of binary black hole (BBH) systems from observed gravitational wave (GW) signals. We focused in estimating the total mass of BBH systems M_T using a convolutional neural network regression (CNNR) model. We built a large dataset of 2D images representing the time-frequency evolution of BBH GW signals which are embedded in noise, where for each generated image the real total mass is known. A hold-out cross-validation procedure was performed to train and evaluate five architectures of CNNR models with different number and sizes of kernels The results indicate that the proposed deep neural network models for regression provide reliable point-parameter estimations with high accuracy. This estimation parameter approach can be easily extended to reconstruct more parameters from astrophysical sources directly from obseved GW events.

1. Introduction

Gravitational waves (GW) are ripples in the fabric of the space-time that were predicted more that one-hundred years ago by Albert Einstein in the general relativity theory.These waves are produced by exotic faraway astrophysical massive objects such as binary black holes (BBH) and core-collapse supernovae (CCSN) and carry novel and unique information of its source and the universe. For a century, GW were merely a theoretical aspect until very recently first detections were finally achieved. GW are detected in earth by the highly sensitive network of ground-based laser-interferometer observatories LIGO, VIRGO and KAGRA (LVK), which register the minuscule strain signals induced by traveling GW (Aasi et al., 2015; Acernese et al., 2015). The firt GW event detected in 2015, named GW150914 (Abbott et al., 2016), along with the around ninety GW detected so far (Abbott et al., 2021) from BBH and one from a binary neutron stars (BNS) system, are remarkable discoveries that open a new whole spectrum to understand of the universe.

There are three main aspect that allowed the discovery of GW and the beginning of a new astronomy. First, the theoretical solutions of Einstein's equations and the understanding of their physical meaning. Second, the LVK network of detectors, which involves sophisticated high-tech facilities that are sensible to the tiny GW signals and provide data to search for such signals. Third, the computational methods and the data analysis algorithms devoted to detect GW signals embedded in noise, to localize the sources in the sky, and to extract physical information of the source. This last aspect is essential for GW astronomy and there are still plenty of open problems such as detection of GW with stochastic or unknown signatures (i.e, detection problem), discrimination between noise artefacts and GW signals (i.e., classification problem), inferring physical information of the source (i.e., estimation problem), among others. Consequentially, computational intelligence methods can play a critical role to tackle those problems.

Currently, estimation of physical information from BBH given the data of detected GW is performed with Bayesian inference based methods, which employs stochastic sampling techniques to compute the joint posterior distribution of the system's parameters such as the masses and spins of the individual black holes for the case of BBH systems. However, sampling methods are known to be extremely expensive since they require highly computational processing times and associated costs. Moreover, application of such methods is a complex task that requires experience in its usage to correctly tune and run the algorithms. These issues are critical for GW astronomy since one is expected to perform rapid and reliable parameter estimation as soon as new data is available from new GW events. Machine learning (ML) and deep learning (DL) methods provide unique op-

[*]Equal contribution [1]Tecnologico de Monterrey, Escuela de Ingeniería y Ciencias, Av. Eugenio Garza Sada 2501 Sur, Colonia Tecnológico Monterrey, N.L., 64849, México [2]Departamento de Física, CUCEI, Universidad de Guadalajara, Blvd. Marcelino García Barragán 1421, C.P. 44430, Guadalajara, Jal., México. Correspondence to: Andres Antelis <A01637683@tec.mx>, Claudia Moreno <claudia.moreno@academico.udg.mx>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

portunities to address these issues, in particular, to perform rapid and reliable estimation of physical information.

In this paper we propose the novel use of DL, in particular convolutional neural network regression (CNNR) models, to estimate physical information of BBH systems from the data of GW signals detected with the LVK network of detectors. We built a large dataset of GW signals from BBH embedded in noise with known physical parameters. In particular, we focused in the estimation of the total mass of the BBH, $M_T = M_1 + M_2$, where M_1 and M_2 are the masses of the two black holes. The input to our CNNR model is a 2D image representing the time-frequency evolution of the observed data while the output is M_T. We implemented several architectures of CNNR consisting of convolutional layers with different number and sizes of the processing kernels and with different number of neurons and layers in the fully connected neural network. The CNNR model were evaluated with a hold-out cross-validation procedure and the results revealed reliable point-parameter estimations with very low errors.

2. Methods

2.1. Problem statement: inferring physical information of BBH from GW

The LVK network of GW detectors provides time-series of strain data which is firstly used to search for GW signals. Once a GW is detected, we want to infer information of source that generated such signal. For the case of BBH systems, examples of physical parameters are the masses and spins of the black holes, the sky localization, and the source distance. In this work, rather that estimating the two masses we focused in the total mass of the BBH system, $M_T = M_1 + M_2$ where M_1 and M_2 are the masses of the two black holes.

To define the GW parameter estimation problem, consider the observed data from a detector in the LVK network $x(t) = [x[0], x[1], ..., x[N-1]]^T$. This is a N-point time-series where there is a GW from a BBH embedded in additive observation noise. Hence, the observed data can be modelled as:

$$x[t] = w[t] + h[t; \boldsymbol{\theta}], \quad t = 0, 1, ..., N-1, \tag{1}$$

where $w[n]$ is the observation noise and $h[t; \boldsymbol{\theta}]$ is the GW signal of a BBH which depends on P parameters $\boldsymbol{\theta} = [\theta_1, \theta_2, ..., \theta_P]$. Here we are interested in a single parameter, the total mass M_T of the BBH system. Our goal is then to find the value of M_T based on the observed data $x(t)$.

2.2. Dataset description

To asses the CNNR model, we constructed a large dataset with many realizations of observed data $x(t)$ associated with the total mass M_T value of the BBH system. To obtain $x(t)$, we first compute GW signal $h(t)$ for a BBH system with masses M_1 and M_2. Each GW was computed using the offline PyCBC search pipeline (Usman et al., 2016). We then obtain a noise realization of the LIGO noise $w(t)$. This is carried out using the real observed noise data available in (https://gwosc.org). The observation is obtained by simply adding the noise $w(t)$ and the GW signals $h(t)$. This procedure is repeated to cover all the parameter space of M_1 and M_2. We consider individual BBH masses $M_1 \in [5; 100]\ M_\odot$ and $M_2 \in [5; 100]\ M_\odot$ with $M_1 \geq M_2\ M_\odot$, which yields to a parameter space for the total mass of $M_{total} \in [10; 200]\ M_\odot$. Note that $M_\odot$ indicates units of solar masses. We sampled M_1 and M_2 with a resolution of 1 $M_\odot$ which results in a total of 4656 pairs of M_1 and M_2 values.

For each observation $x(t)$ we compute the time-frequency representation $X(t, f)$ based on the spectrogram technique. This provides a 2D data matrix of dimensions $N_r \times N_c$, where N_r and N_c represent the number of rows and columns, respectively. As a result, we obtain the dataset $\mathcal{D} = \{X^i(t, f), M_T^i\}_{i=1}^N$, where $X^i(t, f) \in \mathbb{R}^{Nr \times Nc}$, $M_T^i \in [10, 200]$, and N is the total number of images. In total we created a dataset with $N = 4656$ images and each image has a dimension of $N_r = N_c = 28$.

2.3. Convolutional Neural Networks for regression (CNNR)

As a novel approach for estimating the value of parameters for BBH systems we propose to use of DL (LeCun et al., 2015), in particular, a CNNR model that calculates the total mass. The choice of a CNNR model is motivated by the fact that they can learn linear and non-linear relationships between the input and output, they are more appropriate for handling large-dimensional input data, and they offer high performance at a low computational cost to make predictions. The input to the proposed CNNR model is a 2D image $X(t, f)$ representing the time-frequency evolution of the observed data while the output is a 1D parameter representing the total mass M_T. The architecture of a CNNR model is illustrated in figure 1 and consist of the several processing stages.

The fist stage is a convolutional layer with N_F kernels or filters of dimension $m \times n$ which are smaller than the 2D input image. Each kernel is used to perform the convolution operation with the input image followed by a rectified linear unit (ReLU) operation to introduce nonlinearity to the model. This produces N_F 2D feature maps.

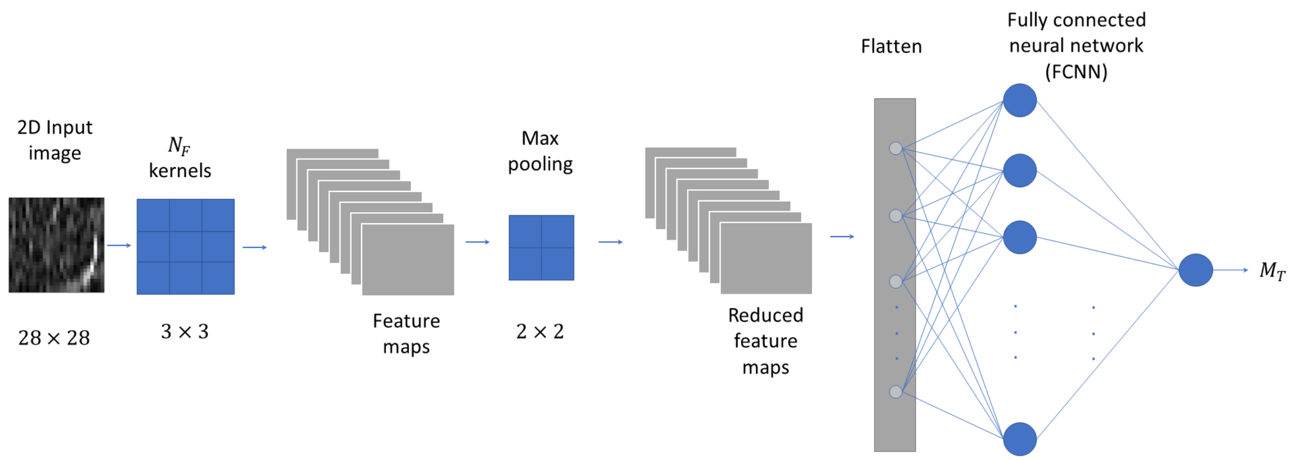

Figure 1. Illustration of the architecture of the CNNR models which consist of one convolutional stage, and max pooling stage, flatten, and FCNN of one hidden layer and one-nueron output.

The next stage is a a pooling operation to decrease the dimension of the feature maps by sliding a filter of dimension $p \times q$ while calculating the maximum or the average. This is followed by a flatten stage which converts the reduced feature maps into a single data vector to be used in the next stage. The final stage is a fully connected neural network (FCNN) with hidden layers and neurons on them to transform the information in the feature maps into a response. Note that more convolutional and pooling stages can be added to form a stack of computational layers prior feeding the FCNN. The number of convolution layers, the number and size of the kernels, the pooling size and type, and the number of hidden layers and the number of neurons on them for the FCNN are tunable parameters (hyperparameters), while the weights of the kernels and of the FCNN have to be learned from a training data set.

We tested several models with different hyperparameters; in especific we considered CNNR models with one, two, and three convolutional layers, kernels of different sizes, different number of kernels in each convolutional layer, and FCNN's with 32, 64, 128, 256 neurons in the hidden layers. And we found the higher estimation performance with only one convolutional layer. Here we present the results achieved with five CNNR models consisting of one convolutional layer consisting of 8, 16, 32, 64, and 128 kernels of size 3×3, pooling of size 2×2 using the maximum operation, and a FCNN with consisting of one hidden layer of 128 neurons and a single-neuron output layer with linear activation function. The used models were named CNN1A, CNN1B, CNN1C, CNN1D, and CNN1E. The architecture of these five CNNR models is illustrated in figure 1, note that they consits if only one convolutional and pooling layers and only differ in the number of kernels N_F. The Keras package was used to implement the CNNR, while for model training the mean squared error was used as a loss function and the adam optimizer, with a batch of 512 samples and 150 epocs

Table 1. Performance metrics (r^2, $rmse$ and mae) obtained with all regression models.

Metric	r^2	$rmse$	mae
Linear	0.94	10.98	2.85
CNN1A	0.96	7.45	2.29
CNN1B	0.96	7.41	2.29
CNN1C	0.96	6.92	2.19
CNN1D	0.96	8.00	2.36
CNN1E	0.95	8.96	2.53

As baseline we also used the linear regression model to predict the total mass M_T as function of the input information $X(t, f) \in \mathbb{R}^{Nr \times Nc}$, where $N_r = N_c = 28$. In this case, the linear regression model is given by:

$$M_T = b_0 + b_1 x_1 + b_2 x_2 + \cdots + b_{784} x_{784} \quad (2)$$

where, b_0, b_1, b_2, ... , b_{784} are the parameters that need to be learned from a training dataset.

2.4. Evaluation procedure and performance metrics

To asses the CNNR performance in the estimation of the total mass M_T from $x(t)$, we employed a hold-out cross-validation (HOCV) procedure where the entire data set $\mathcal{D}$ was randomly splitted into two mutually exclusive sets, one for training (70%) and other for testing (30%). The train set

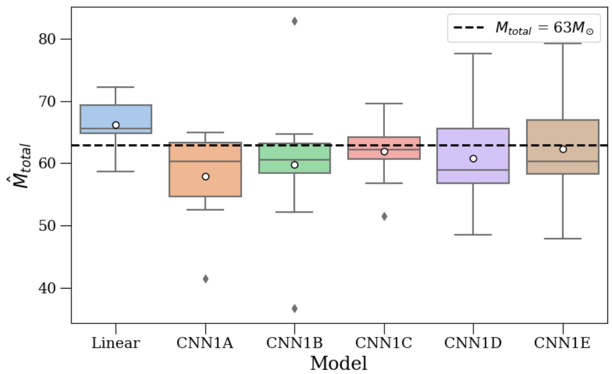

Figure 2. Distribution of estimated values of the total mass $\hat{M}_\odot$ obatined with all models for a true total mass M_T value of 63M$\odot$.

is used to tune the CNNR model while the test set is used to compute the performance in the estimation of M_T for unknown GW signals. To asses performance we computed the r-squared coefficient (r^2) to measure the linear relationship between the distribution of estimated values $\hat{M}_T$ and the distribution of true values M_T, and the root mean-squared error ($rmse$) and the mean absolute error (mae) to measure the average of squared errors and the average of absolute errors across true values M_T and the estimated values $\hat{M}_T$.

3. Results

Table 1 shows the values of the performance metrics obtained with the linear and the five CNNR models. The r^2 metric is greater than 0.94 for the models with the linear model providing the lower linear correlation between real and estimated values. Regarding the error metrics, $rmse$ and mae, all models provided low estimation errors with the higher errors provided by the linear model. These global results indicate that all models provided good estimation of the unknown parameter total mass, with slightly better performance provided by models CNN1B and CNN1C.

To examine the estimation performance for specific values of the total mass, we analyzed the distribution of estimated values for specific true values. For instance, figure 2 shows the distribution of $\hat{M}_T$ for the case of true total mass M_T value of 63M$\odot$. All models but CNN1C presented a $\hat{M}_T$ distribution with mean and median values separated from the true value. MLR and CNN1A provided the worse performance with bias of around 10 $M\odot$, respectively. The CNN1C model on the other hand presented the best estimation performance with a distribution of $\hat{M}_T$ closely centered at 63 $M\odot$, indeed the bias for this model was lower than 1 $M\odot$. Similar results were obtained for other true values of the true total mass M_T. Hence, the CNN1C model provided the best estimation performance and was selected in the subsequent analyses.

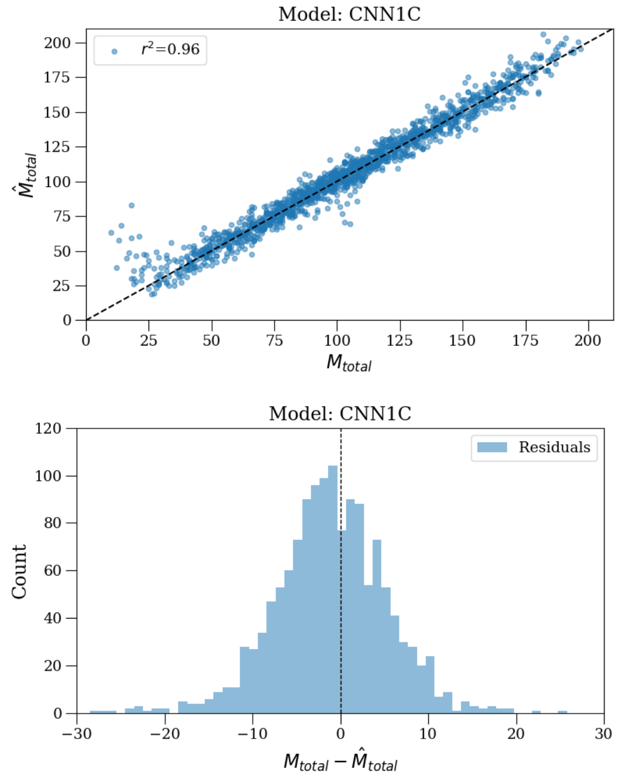

Figure 3. Scatter plot (upper panel) of real versus estimated values of the total mass and the distribution of residuals (lower panel) obtained in the CNN1C model.

To further analyse the performance of the CNN1C model, in figure 3 we present the scatter plot of M_T versus $\hat{M}_T$ and its distribution of residuals (i.e., $M_T - \hat{M}_T$). The scatter plot shows a high linear relationship between M_T and $\hat{M}_T$ with a correlation coefficient of 0.96, while the distrubution of residuals shows a bell-shaped distribution with mean value close to zero. The only observation to consider is the high dispersion in $\hat{M}_T$ for small values of M_T, which can be due by the fact that the number of samples in the dataset is little in those values of M_T. Irrespective of this, these results reveal a high estimation accuracy with the CNN1C model.

4. Conclusions

In this work we propose the novel use of CNNR models to estimate the total mass of BBH systems from GW signals detected with the LVK network of detectors. Traditionally, this kind of estimation problem is addressed with sampling methods which presents several difficulties as the extremely high computational burden and the high experience required to use the methods. On the contrary, our proposed CNNR model offer several advantages. Once a model is trained we only need to provide input data to obtain a point-wise estimation. It is possible reconstruct of physical parameters within a few milliseconds using a single computer. The noise and signal models are used only to obtain data to train the models, and not every time we want to perform estimations. It is possible to consider multiple noise a signals models in the construction of the training data set, so that, the learned model can automatically reconstruct parameters in different modelling situations. Also, it is possible to re-train the model as new data is available.

The proposed CNNR models consists of a convolutional layer with processing kernels, a fully connected neural network, and an output neuron that provide the point-wise estimation. The input to this model is a 2D data matrix representing the spectrogram of the observed data while the output is the value of the estimated parameter. As part of this study, we constructed a data set of GW signals embedded in observation noise with known values of the parameter, the total mass, and we use this data set to asses the estimation performance of several CNNR architectures and of basic multiple linear regression model. The results showed that deep learning methods can readable be used to estimate information from noisy observations, with superior performance than that of the linear model. This is possibly because CNNR models can learn linear and non-linear relationships between the input and output without any assumptions. The potential limitations of our CNNR models is not considering other wave forms in the dataset, and as such the impossibility of not being able to estimate the M_T of that type of GW.

As future work we plan to test our CNNR models on real BBH GW signals detected by LVK network of detectors which could further demonstrate the effectiveness and robustness in real scenarios. In addition we plan to extend the estimation to multiple parameters of BBH systems as the masses and the spins of the two black holes. Finally we will consider Bayesian neural networks to incorporate uncertainty in the predictions because in astronomy it is important to provide confidence interval containing model error instead of a point wise prediction.

References

Aasi, J. et al. Advanced LIGO. *Class. Quant. Grav.*, 32: 074001, 2015. doi: 10.1088/0264-9381/32/7/074001.

Abbott, B. P. et al. Observation of gravitational waves from a binary black hole merger. *Phys. Rev. Lett.*, 116:061102, Feb 2016. doi: 10.1103/PhysRevLett.116.061102.

Abbott, R. et al. GWTC-3: Compact Binary Coalescences Observed by LIGO and Virgo During the Second Part of the Third Observing Run. 11 2021.

Acernese, F. et al. Advanced Virgo: a second-generation interferometric gravitational wave detector. *Class. Quant. Grav.*, 32(2):024001, 2015.

LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *Nature*, 521(7553):436, 2015.

Usman, S. A. et al. The pycbc search for gravitational waves from compact binary coalescence. *Classical and Quantum Gravity*, 33(21):215004, oct 2016.