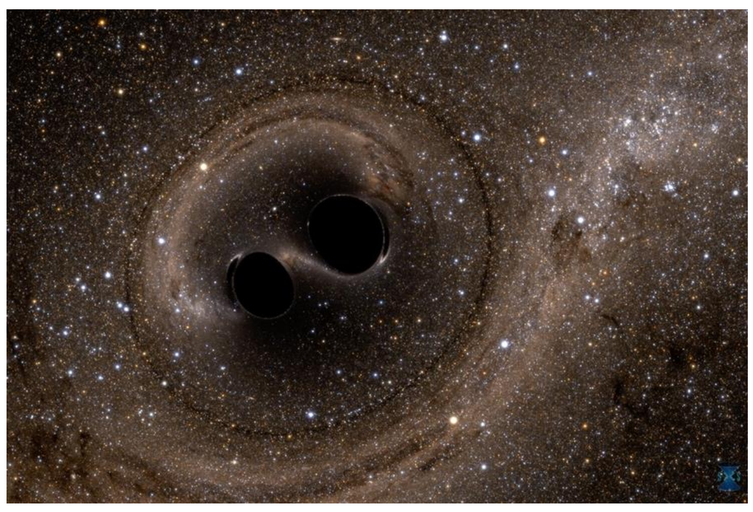
UNIVERSITY of WASHINGTON

Daniel Fredin[1], Cole Welch[1]
[1]*Department of Physics, University of Washington*

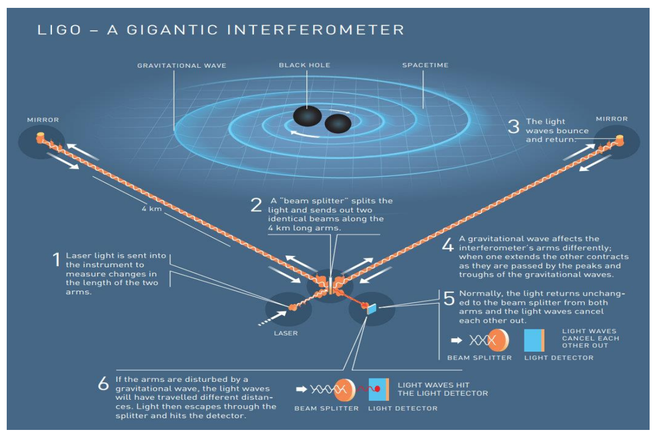
UNRAVELING GRAVITATIONAL RIPPLES: NEURAL NETWORK CLASSIFICATION

OBJECTIVES

The Glitch dataset originates from the Laser Interferometer Gravitational-Wave Observatory (LIGO), an observatory designed to harness the characteristics of light and space in order to identify and comprehend the sources of gravitational waves. We analyze a dataset consisting of four distinct data categories: Glitch, Background, Binary Black Hole (BBH), and Sine-Gaussian (SG). Our objective is to develop a binary classifier capable of identifying signals as either Glitch/Background or Binary Black Hole/Sine-Gaussian.



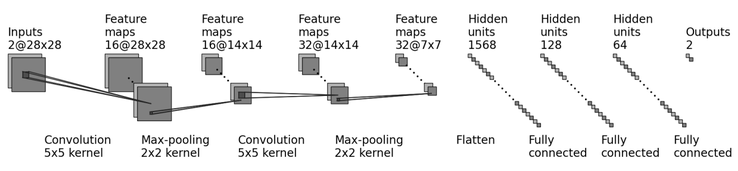
Snapshot from artist's rendition of binary black hole merger.



Basic layout of the LIGO interferometer.

METHODOLOGY

To solve this problem, we utilize PyTorch to implement a binary classifier that takes a time series as input and outputs either signal (for glitch/background data) or background (for binary black hole/sine-Gaussian data). Before passing data to our classifier, we need to convert from time series to a 2D image in order for our CNN to parse it. We accomplish this process using Gramian Angular Fields (GAFs) as implemented in the Python library, *pyts*. All GAF images are 28x28 pixels, which was found to be a good compromise between performance and accuracy. Next, we split our data into training (70%, 102227 images), testing (15%, 21910 images), and validation (15%, 21908 images) lists. These are then passed into our CNN architecture, which begins with a convolution layer of 2 input channels and 16 output channels followed by a convolution layer with 16 input channels and 32 output channels; each has kernel size 5, stride length 1, and padding 2. Following each of these convolution layers is a max pooling layer of kernel size 2, stride 2, and padding 0. Finally, the network contains 3 subsequent linear, fully connected layers of decreasing size followed by the output; their exact input and output dimensions are 1568x128, 128x64, and 64x2. We trained the model for 16 epochs at a learning rate of $5 \cdot 10^{-6}$ in batches of 677 with an L2 regularization term of coefficient 0.001 to decay weights and reduce overfitting. The network was subsequently trained using a cross-entropy loss function and Adam optimizer.



Basic structure of our CNN.

Quick Facts About Our CNN

> **~97%** testing accuracy

> Data split of **70%** training, **15%** validation, and **15%** testing

> Dataset contains a total of **146045** gravitational wave time series

> Applied **Gramian Angular Field** (GAF) algorithms to encode gravitational wave data as 2D images for classification

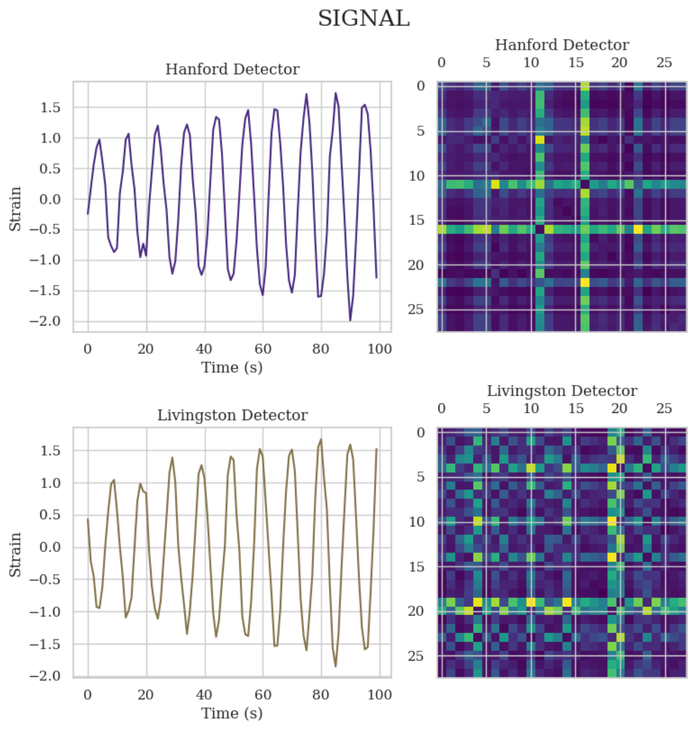
> Neural network consisted of **2** convolutional / pooling layers followed by **3** fully connected layers and an output

> Utilized a **cross-entropy** loss function with an **Adam** optimizer

> Utilized **batch normalization** and **ReLU** activation functions along with **L2 Regularization**

DATA ANALYSIS

The data that we used was obtained from the LIGO observatories at both Hanford, WA and Livingston, LA. To analyze this dataset properly, we confirmed that it was scaled properly by utilizing the StandardScaler class which ensured that our data had a mean of 0 and a standard deviation of 1. From this we converted our normalized strain data into GAF images for insertion into a deep CNN for binary classification. The output of the model gave us a probability prediction of the images for either background or signal which we then compared to our ground truth targets. The trained model was then tested against our testing data set and achieved an overall accuracy of 97%. From the plots of our correctly and incorrectly identified signals, it appears that our model incorrectly classifies the images when the two detectors have disagreeing data. For instance, in our last incorrectly classified signal, the Hanford detector appears to be noise while the Livingston detector is observing a signal. However, our model classifies both as noise, seeming to become confused when the two detectors receive contradictory signals. This could also be a flaw arising from the observatories, where one may be capturing signals while the other is solely identifying noise.

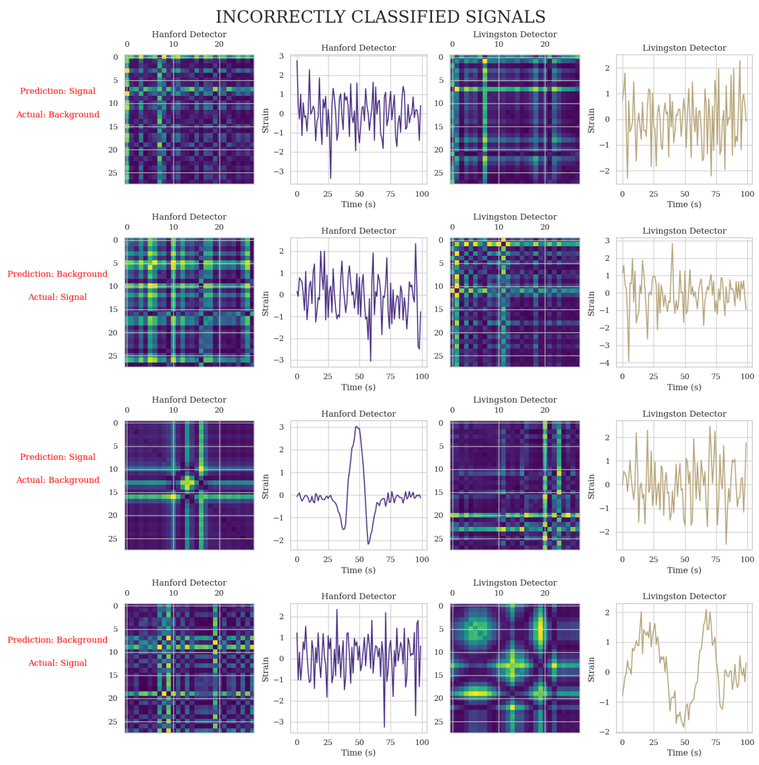


Sample signal from time series training data (left) and the corresponding GAF (right) for both the Hanford and Livingston detectors.

RESULTS

Observing our training loss and validation accuracy plot, we can infer that our model is learning the dataset well. After a total of 16 epochs and more than 2400 iterations, our training loss is about 0.1 while our validation accuracy is near 96%. Overall, our CNN model's testing accuracy obtains a respectable target of 97%, nearly achieving the same accuracy seen in similar CNN architectures in the literature. Some examples of these are (Fernandes et al., 2018) and (George D. et al., 2023), who achieve 99% and 98.8% accuracy, respectively. However, it should be noted that our dataset comprised entirely separate time series for gravitational wave signal data and noise data, while the authors of these works had to employ more complex methods to filter the noise out from their data. For instance, while Gramian Angular Fields were quite effective at image conversion, others found Fourier transforms better suited to the task of handling complex noise and background by filtering in the frequency domain.

Sample Incorrect IDs



***Top**: Some examples of incorrectly identified signal or background data.*
***Bottom**: Our CNN's training loss and validation accuracy as a function of epochs elapsed.*

CONCLUSION

In this work we analyze the potential of convolutional neural networks (CNNs) to help classify gravitational waves strain data as background noise/glitches or transient sine-gaussian/binary mergers. Employing a simplified version of the most common approaches in the literature, we were able to surpass our expectations by achieving an accuracy above 95% and only slightly below the more complex CNNs used for comparison. A question to be further explored could be to see how well the Gramian Angular Field approach to converting time series to images and passing them through a simple CNN would translate to more realistic data.

[1] Fernandes T. S., Vieira S. J., et al. (2023). Convolutional Neural Networks for the classification of glitches in gravitational-wave data streams. arXiv:2303.13917 [gr-qc].
[2] George D., Shen H., Heurta E.A. (2018). Glitch Classification and Clustering for LIGO with Deep Transfer Learning. Phys Rev. D 97, 101501. arXiv:1711.07468v2 [astro-ph.IM].