Universität Münster

Bachelor Thesis

Identifying Sub-Solar Primordial Black Holes for Einstein Telescope Using Deep Learning Methods

Supervisor:
Prof. Dr. Alexander Kappes

Second examiner:
Prof. Dr. Anton Andronic

A thesis submitted in fulfilment of the requirements for the degree of

Bachelor of Science *at* Universität Münster

by

Caner Bahceci

AG Kappes
Institut für Kernphysik

Münster, March 2026

Contents

1 Introduction **1**

2 Theory **3**
2.1 Primordial Black Holes 3
2.2 Gravitational Waves 4
2.2.1 Development of Gravitational Waves 4
2.3 Einstein Telescope 6
2.3.1 General Detection Principle 6
2.3.2 Improvements with Einstein Telescope 7

3 Machine Learning **11**
3.1 Deep Neural Networks 11
3.2 Residual Network 12
3.3 Autoencoder 14

4 Methodology **17**
4.1 Data Simulation 17
4.2 ResNet and Autoencoder 21
4.2.1 Residual Network 21
4.2.2 Autoencoder 24

5 Results **25**
5.1 ResNet 25
5.2 Autoencoder 28

6 Conclusion **37**

Bibliography **39**

Declaration of Academic Integrity **43**

1 Introduction

Gravitational-wave (GW) analysis provides a deeper insight into high-energy astrophysical events, including the coalescence of black hole binaries. The detection of GWs has already been realised by experiments such as advanced LIGO [1] and Virgo [2]. These detectors measure distortions of spacetime caused by passing gravitational waves using Michelson interferometers with long detector arms in the kilometre range. While these observatories have successfully identified GW signals from different astrophysical sources, there is still room for improvement in detector sensitivity. The development of third-generation GW detectors marks a significant advance in the search for sub-solar mass **primordial black holes** (PBHs). In particular, the **Einstein Telescope** (ET) [3], with its excellent low-frequency sensitivity (down to 3 Hz), will allow for the early detection of GW signals from compact binaries, including sub-solar mass systems. Typically, detector noise overwhelms the signal, making it impossible to visually identify a signal from the given detector strain. Because of that, established techniques such as Bayesian inference and matched filtering have been developed, relying on statistical methods to extract weak signals from noisy data. However, these methods are not very efficient, as they can take days or even weeks to fully characterise a detected event. In contrast, although training **deep-learning models** on large datasets is itself computationally demanding, once trained, such models can identify and classify signals within a couple seconds or less. Sub-solar mass PBH binary coalescences pose a challenge for detection, as the high frequency (up to a couple of kHz) in the late inspiral and merger phase can exceed the most sensitive frequency range of current and future detectors (around 5 kHz). Nevertheless, PBHs are of considerable interest, as they represent a viable candidate for dark matter [4]. This thesis presents a methodology for simulating large-scale datasets of compact binary coalescences, including signals from **astrophysical black holes** (ABHs), PBHs, and noise-only signals. Based on these datasets, deep-learning models are developed to identify and classify GW signals of these compact binaries directly from detector strain. The models in question are a **residual network** (ResNet) with the sole goal of classifying the detector strain and an **autoencoder**, which extends the architecture of the ResNet and allows for a deeper analysis of the waveform. On top of the classification, the goal of the autoencoder is to perform parameter estimation and reconstruct the pure waveform of these binaries, directly from the detector strain.

2 Theory

In the following chapter, the theoretical background of the **Einstein Telescope** (ET) [3], **gravitational waves** (GWs) and **primordial black holes** (PBHs) is introduced. In addition, the general principle of GW detection is discussed, along with the planned improvements of ET and their expected impact on the search for PBHs.

2.1 Primordial Black Holes

PBHs are black holes created in the early universe from non-stellar sources. Their formation goes all the way back to the early expansion of the universe during the radiation dominated era with high density fluctuations. These so called "primordial fluctuations" are nearly spherical and can gravitationally collapse to black holes [4]. Such gravitationally collapsed black holes could have a mass of 10^{-5} g upwards. Furthermore, one with a mass of 10^{17} g could be at the centre of the Sun [5]. This means that PBHs can have masses well below a solar mass $M_\odot$, which distinguishes them from **astrophysical black holes** (ABHs), as a star with a mass of $\lesssim 1.5\,\mathrm{M}_\odot$ cannot support itself against gravity once it runs out of nuclear fuel [5]. The Schwarzschild radius, which determines the radius of the event horizon of a black hole, can be calculated using [5]:

$$r_S = \frac{2GM}{c^2}. \tag{2.1}$$

Here G is the gravitational constant, M the mass of the object and c the speed of light in a vacuum. Given eq. 2.1, taking the Planck length as the radius $r_S = 10^{-33}$ cm, results in a PBH of mass $M = 10^{-5}$ g. This concludes that a PBH cannot have a lower mass than that, as the quantum fluctuations would be of order unity, meaning the gravitational collapse is, in first approximation, a classical process [5]. Additionally, these black holes do not substantially increase in mass due to accretion, meaning that there could be PBHs around with masses as low as $M = 10^{-5}$ g and upwards today. This is due to the fact that shortly after the formation of PBH, the rapid expansion of the cosmic background and the radiation pressure suppress accretion, making it negligible compared to the PBH mass. For low mass PBH, Hawking evaporation dominates over the growth in mass. Additionally, as a consequence of the suppressed accretion, the PBH spins are thought to be very small compared to ABHs [6]. If two such small, compact objects begin to orbit each other, they start to inspiral until they finally merge. However, due to their small Schwarzschild radius, they can inspiral for a very long time and get very close before eventually merging, leading to high frequencies of GWs, which will be discussed in the following section.

2.2 Gravitational Waves

This section will provide a closer look at GWs as a consequence of Einstein's theory of **general relativity** (GR). The first thing one needs to understand about GR is that gravity is not a force of nature in the classical sense, as Newton proposed. The idea is that mass and energy curve spacetime itself, which forces other objects into an orbit around them, once they fall into the curved plane, creating an effect that seems like a force attracting masses. Additionally, the effect of gravity is not instantaneous, as suggested by Newton, it is actually limited by the speed of light. The descriptions and equations of GWs will mostly follow source [7] (Ch. 16, 21 and 23) and source [8].

2.2.1 Development of Gravitational Waves

A GW is generated whenever massive objects undergo accelerated, asymmetrical motion. These waves curve spacetime while propagating through spacetime themselves. They couple very weakly to matter, which makes them challenging to detect. Therefore, GWs are derived under the assumption that they are small perturbations of the Einstein equation [7]:

$$R_{\alpha\beta} - \frac{1}{2} g_{\alpha\beta} R = 8\pi T_{\alpha\beta}. \tag{2.2}$$

$R_{\alpha\beta}$ is formally known as the Ricci tensor and broadly describes the local geometry of space compared to Euclidean space. R represents the Ricci scalar, given by the relation $R = g^{\alpha\beta} R_{\alpha\beta}$. On the right side of the equation $T_{\alpha\beta}$ is the stress-energy tensor describing the flux of the αth component of the four-momentum vector across a surface in spacetime x^β [9]. To describe a small perturbation in this metric, examine $g_{\alpha\beta}$, which represents the structure of spacetime itself. If $n_{\alpha\beta}$ represents flat spacetime, then $h_{\alpha\beta}$ is a perturbation in the form of a weak ripple in spacetime:

$$g_{\alpha\beta} = n_{\alpha\beta} + h_{\alpha\beta}. \tag{2.3}$$

In rectangular coordinates (t, x, y, z), flat spacetime becomes $n_{\alpha\beta} = (-1, 1, 1, 1)$ and assuming a small perturbation $|h_{\alpha\beta}| \ll 1$ for all α and β. The left-hand side of eq. 2.2, also known as the Einstein tensor, is commonly written as $G_{\alpha\beta}$ and can be expressed in linear orders of $h_{\alpha\beta}$:

$$G_{\alpha\beta} = \frac{1}{2} \left(\partial_\alpha \partial^\mu h_{\mu\beta} + \partial_\beta \partial^\mu h_{\mu\alpha} - \partial_\alpha \partial_\beta h - \Box h_{\alpha\beta} + \eta_{\alpha\beta} \Box h - \eta_{\alpha\beta} \partial^\mu \partial^\nu h_{\mu\nu} \right), \tag{2.4}$$

where $\partial_\alpha \coloneqq \partial / \partial x^\alpha$, $\partial^\alpha \coloneqq \eta^{\alpha\beta} \partial_\beta$, $h = \eta^{\alpha\beta} h_{\alpha\beta}$ is the trace of $h_{\alpha\beta}$ and $\Box = \eta^{\alpha\beta} \partial_\alpha \partial_\beta$ is the flat space d'Alembertian operator [8]. This equation needs to be solved for the perturbation. To help clean up the equation, define $\overline{h}_{\alpha\beta} = h_{\alpha\beta} - n_{\alpha\beta} h/2$, which gives

$$G_{\alpha\beta} = \frac{1}{2} \left(\partial_\alpha \partial^\mu \overline{h}_{\mu\beta} + \partial_\beta \partial^\mu \overline{h}_{\mu\alpha} - \Box \overline{h}_{\alpha\beta} - \eta_{\alpha\beta} \partial^\mu \partial^\nu \overline{h}_{\mu\nu} \right). \tag{2.5}$$

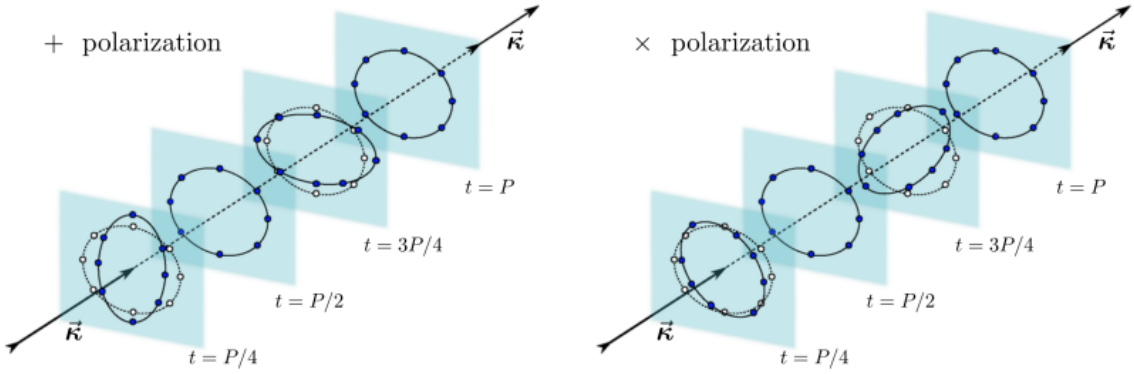


Figure 2.1: GW causing periodic deformations illustrated for two distinct polarizations + plus and × cross [11].

The Lorenz gauge, $\partial^\alpha \overline{h}_{\alpha\beta} = 0$, eliminates unphysical degrees of freedom from the equation 2.5 [8]

$$G_{\alpha\beta} = -\frac{1}{2}\Box \overline{h}_{\alpha\beta}. \tag{2.6}$$

The Lorenz gauge, therefore, reduces the degrees of freedom of the 4×4 $h_{\alpha\beta}$ matrix from 10 to 6, which finally gives:

$$\Box \overline{h}_{\alpha\beta} = -16\pi T_{\alpha\beta}. \tag{2.7}$$

This equation can be solved using the radiative Green's function [8, p. 19]:

$$\overline{h}_{\alpha\beta}(\vec{x}, t) = 4 \int d^4x' \, \frac{T_{\alpha\beta}(\vec{x}', t)}{|\vec{x} - \vec{x}'|}. \tag{2.8}$$

Here $\vec{x}$ is the coordinate in spacetime at which the point is evaluated and $\vec{x}'$ is the location of the source. Far away from the sources, one finds that $\Box \overline{h}_{\alpha\beta} = 0$. The solutions are wave-like with a propagation speed of c. To simplify the problem, the transverse-traceless gauge is applied, which eliminates the unphysical degrees of freedom and leaves two modes of polarisation for the GW [10]. Furthermore, assume that the GW propagates in the z-direction in a vacuum:

$$h^{\mathrm{TT}}_{\alpha\beta}(t, z) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & h_+ & h_\times & 0 \\ 0 & h_\times & -h_+ & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \cos\big(\omega(t - z)\big). \tag{2.9}$$

The final solution 2.9 provides the amplitudes for the two different polarizations h_+ and $h_\times$. In conclusion, the plane perpendicular to the dimension of propagation periodically deforms by stretching in one and squeezing in the opposite direction, oscillating between these deformations. This effect is illustrated in Fig. 2.1.

For compact binary coalescences[1], such as black holes orbiting each other, one must consider the quadrupole radiation formula for the metric perturbation [11]:

$$h_{ij} = \frac{2}{r}\Lambda_{ij,kl}\frac{d^2Q_{kl}}{dt^2}, \tag{2.10}$$

where r is the distance to the source, $\Lambda_{ij,kl}$ is the Lambda tensor that projects the metric perturbation from the Lorenz gauge into the transverse-traceless gauge and Q_{kl} is the reduced quadrupole moment of the sources mass density. Although the proof is non-trivial, for the sake of this thesis, proceed directly to the result describing the loss in the binary's orbital energy [8]:

$$L_{\mathrm{GW}} = \frac{128}{5}M^2R^4\omega^6 = \frac{128}{5}4^{1/3}\left(\frac{\pi M}{P}\right)^{10/3}, \tag{2.11}$$

where the loss depends on the mass of the binary M and the period P. The closer the two objects inspiral, the smaller the period; in other words, the higher the frequency of GWs, the greater the loss of energy in the binary, resulting in a larger strain on spacetime. This conclusion leads one to the detection principle for the strain of GWs. The objective is to determine an experiment to measure the resulting stretching and squeezing effects caused by the propagating GWs.

2.3 Einstein Telescope

The Einstein Telescope (ET) [3] is a planned next-generation GW detector designed to improve sensitivity through an extended frequency range and reduced noise compared to current GW detectors. GWs couple weakly to matter, meaning that the measurable effect is extremely small. This section outlines the general detection principle for GWs and the specifics of how ET will improve compared to current generation detectors.

2.3.1 General Detection Principle

GW detectors are modified Michelson interferometers with very long detector arms. A current generation observatory, advanced LIGO [12], is a network of two detectors (interferometers) separated by about 3000 km, which have arm lengths of 4 km. Each of these arms is bounded by two mirrors such that they form an interferometer (cf. Fig. 2.2).
In principle, an Interferometer consists of several essential components. First, a monochromatic light source is used to prevent any dispersion effects when reflecting the light. This light passes through a beam splitter, where about 50% of the light is transmitted and 50% is reflected at a 90° angle. Afterwards, the light travels as two separate beams through both detector arms simultaneously until they are reflected back by the mirrors at the end of each arm. While returning, the light passes through the beam splitter once more and the beams are directed towards the detector that measures the resulting interference pattern. If no GW passes through the detector plane, both beams' travel paths are of the same length; therefore, there

[1] coalescence describes the process of merging

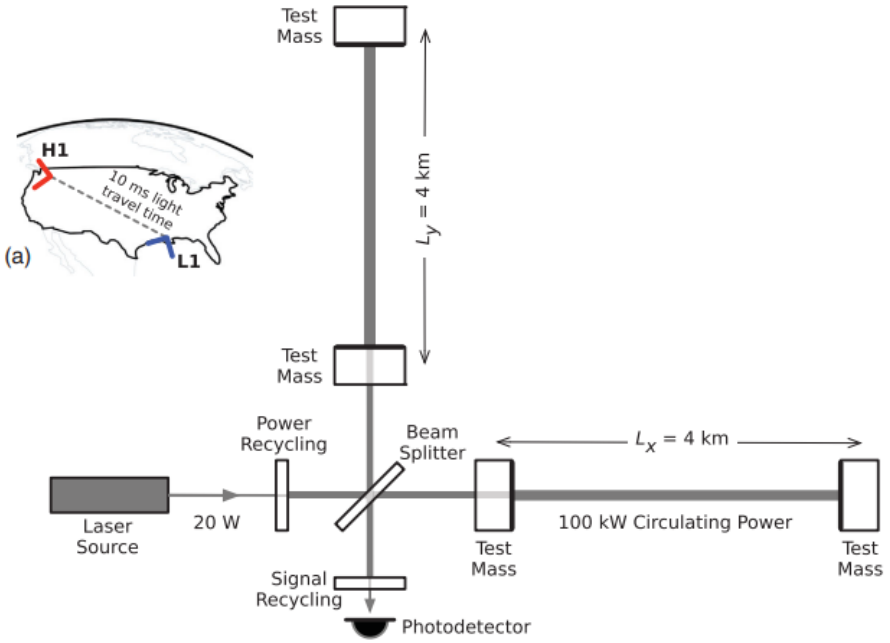


Figure 2.2: Simplified diagram of the advanced LIGO detector adapted from [12].

would be no phase difference between the two beams. In the other case, one of the detector arms will be stretched while the other will be compressed, leading to a difference in path length. This induces a phase shift between the two beams, which is measured through the resulting interference pattern and is directly proportional to the strain amplitude $h(t)$ of the GWs [12]:

$$\delta L(t) = \Delta L_x - \Delta L_y = h(t)L. \tag{2.12}$$

The following section focuses on the different proposed configurations of ET that aim to improve sensitivity, especially in the low frequency range, which is dominated by seismic noise. If the low-frequency sensitivity (a couple of Hz) can be successfully increased, it will allow for early detection and, therefore, longer lasting GW signals.

2.3.2 Improvements with Einstein Telescope

The ET is currently in the development phase and therefore does not yet have a defined geometry. There are currently six proposals, which consider either constructing two separate L-shaped detectors (2L) or a single triangular configuration [13][3]:

- a triangle with 10 km arms
- a triangle with 15 km arms
- a 2L with 15 km parallel arms
- a 2L with 15 km arms 45° relative orientation
- a 2L with 20 km parallel arms
- a 2L with 20 km arms 45° relative orientation [13]

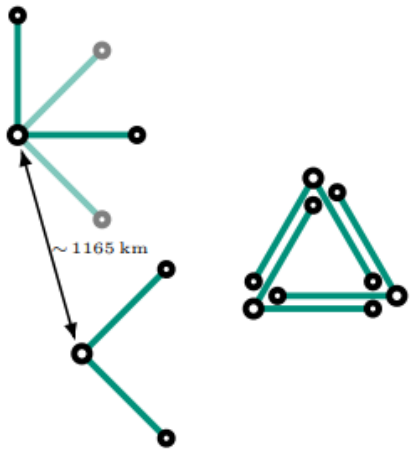


Figure 2.3: Left: Two separated L-shaped detectors either parallel or 45° degrees rotated. Right: Triangular configuration with three detectors [13].

The two configurations are represented schematically in Fig. 2.3. There is a study that investigates the optimal detector configuration for searching for PBHs with the ET, as well as in combination with other next-generation gravitational-wave detectors such as **Cosmic Explorer** (CE) [14]. This study considers all the configurations mentioned above. The "xylophone" setup proposes that each detector in the triangular configuration consists of two interferometers, such that one is optimised for the low frequency range 3–30 Hz and operates at cryogenic temperatures, while the other is optimised for the high frequency range 30 Hz up to a couple of kHz [13][15]. The study finds that there is only a minuscule difference between the setups in the search for PBHs. The most significant improvements in detection come from extending the length of the detector arms (cf. Fig. 2.4). In summary, the xylophone configuration outperforms the setup optimised for higher frequencies in both determining the masses of the objects in the binary and the redshift, which provides information about the distance of the binary. Combining ET with CE enables more confident parameter estimation and improved source localisation [14]. As discussed in previous sections, GWs from sub-solar mass PBH binaries produce weak detector strain during the early inspiral and their signals become comparable to those from ABHs once they reach the merger phase, due to their high frequency in close orbit, where the emitted GWs reach frequencies of several kHz. In conclusion, high frequency sensitivity is crucial in the search for PBHs using GW detectors. Finally, ET is expected to enable the following scientific advances: With its excellent low-frequency sensitivity, ET allows for a closer look at the early inspiral part. This enables the search for GW polarizations, which are inaccessible with current detectors. Additionally, it allows constraints to be placed on the mass of the graviton, the hypothetical mediator of the gravitational interaction. ET will be able to determine whether the gravitational interactions driving the coalescence of compact objects are described by GR or if the data points to another theory. Furthermore, ET will enable accurate measurements of the ringdown phase after the merger with an accuracy of $\mathcal{O}(1)\%$ and allow for tests of tidal deformability effects and absorption, with almost an order of magnitude improvement over current generation GW detectors. Finally, ET has the potential to probe the existence of new particles in nature, some

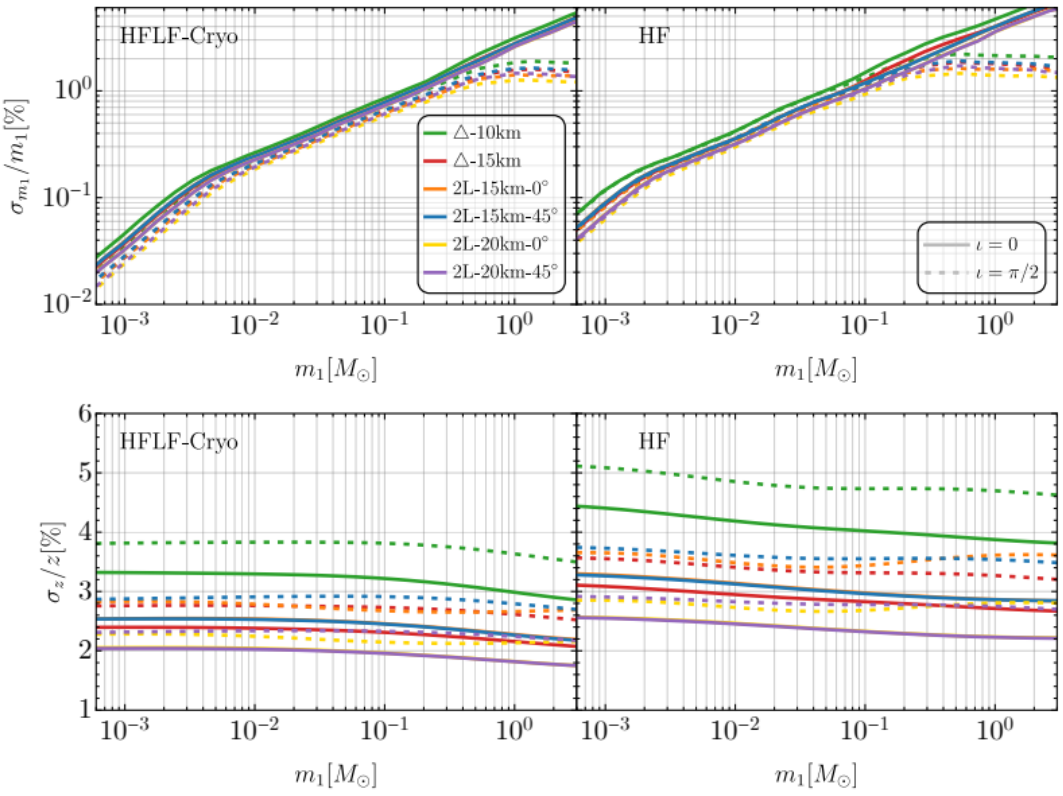


Figure 2.4: Relative error in primary source frame mass (top) and source redshift (bottom). HFLF is the xylophone configuration and HF a configuration only optimised in higher frequency ranges [14].

of which may serve as viable dark matter candidates, similarly to PBHs [3].

3 Machine Learning

Machine learning includes algorithms that use statistical methods to learn from data. Deep learning is a subset of this category in which multilayered neural networks learn different features from vast amounts of datasets. This chapter introduces the basic concepts required to understand the methodology and results of this thesis, including: a general introduction to deep neural networks, residual networks and autoencoders.

3.1 Deep Neural Networks

Deep neural networks (DNNs) are models designed to learn features from data by using multi-layered neural networks, which is referred to as "deep learning" [16]. For example, a GW signal is barely visible by eye (cf. Fig. 3.2). By inputting many labeled examples into a DNN, it can learn whether there is a signal within the strain after a so-called "training process". DNNs use many "hidden layers" to accomplish this goal. The output of the first layer becomes the input of the next layer and so on. Each layer consists of a number of neurons [17] representing the inputs and outputs. In Fig. 3.1, an example for a DNN with one hidden layer is illustrated. Typically, a deep neural network consists of multiple hidden layers where the early layers extract local low-level features, while deeper layers combine these into more global and abstract representations. What exactly happens during these layers differs depending on the architecture of the model. The inputs and outputs are vectors containing different values that characterise the input data. These models are then trained on datasets containing the input and the output. One needs to identify the task of the model and use a type of DNN that can accomplish the task. The model is essentially a mathematical equation to compute a prediction y based on an input x [16]:

$$y = f[x]. \tag{3.1}$$

Here, both x and y are vectors of a fixed size and the elements of the input data are ideally independent and identically distributed random variables. Typically, the input data is normalised. Normalisation puts all the features of a dataset on the same scale to speed up convergence. This is to mitigate issues related to vanishing or exploding gradients [18], which essentially means that the model does not put too much weight on large features, because the noise amplitude would overwhelm any nuance in the datasets. The model additionally contains parameters ϕ:

$$y = f[x, \phi]. \tag{3.2}$$

These parameters are the network's trainable variables, including weights and biases, that determine the relation between the input and the output. The weights determine how strong these relations are. The biases set a baseline for the parameters [16]. During DNN training, the model parameters ϕ are iteratively updated

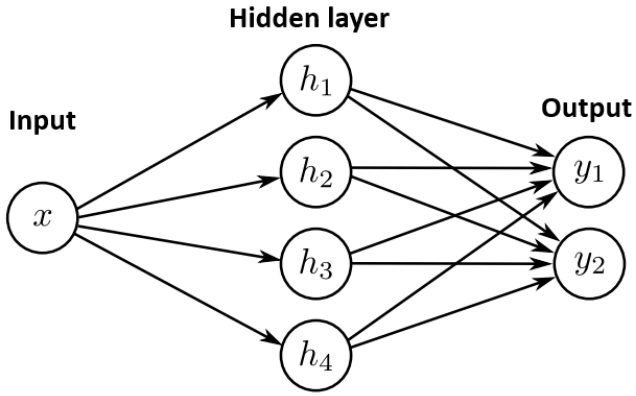


Figure 3.1: Example of a neural network with 1 input neuron, a hidden layer with 4 neurons and 2 output neurons that are all fully connected. Adapted from [16].

in the direction that minimises the values of a predefined loss function over the training data. These updates are computed through the so-called "gradient descent" algorithm [16]. The "loss" L is defined as a scalar value that determines how well the model is performing by comparing the model's outputs to the actual outputs of the corresponding inputs during training (the ground-truth information); therefore, the choice of the loss function strongly depends on the task. To optimise the model, the goal is to find ϕ so that $L[\phi]$, the loss function, is minimised:

$$\hat{\phi} = \text{argmin}\left[L[\phi]\right]. \tag{3.3}$$

By minimising the loss function, the optimal parameters $\hat{\phi}$ are determined to accurately predict the training outputs from the training inputs. In the next step, the model needs to be tested on data separate from the training set to evaluate its performance on inputs of the same type it has not been directly trained on. This approach is called "supervised learning" [16].

3.2 Residual Network

Residual Networks (ResNets) [20] are often used for image classification. ResNets are especially effective for creating **convolutional neural networks** (CNNs) [21] with many hidden layers. Originally, CNNs were used for image classification. CNNs use convolutional layers that take an input and move a filter or a "kernel" around the input image in an attempt to identify a pattern. An example can be seen in Fig. 3.3; if the input is an image of a dog, the different features could be the ears or the paws of the dog. These features can be found with a kernel that scans the image in predefined steps to identify different local features, such as curves, lines and edges. The features are then mapped out and collected in the output. Each kernel has the same weight, hence why it does not matter where the feature of an image was found. This same principle can be applied to datasets with underlying "chirp" patterns, such as the strain from GW detectors. These are called 1D CNNs, which slide a kernel along a single dimension [22], making the original CNN for image classification a 2D CNN. One would assume that adding many hidden layers

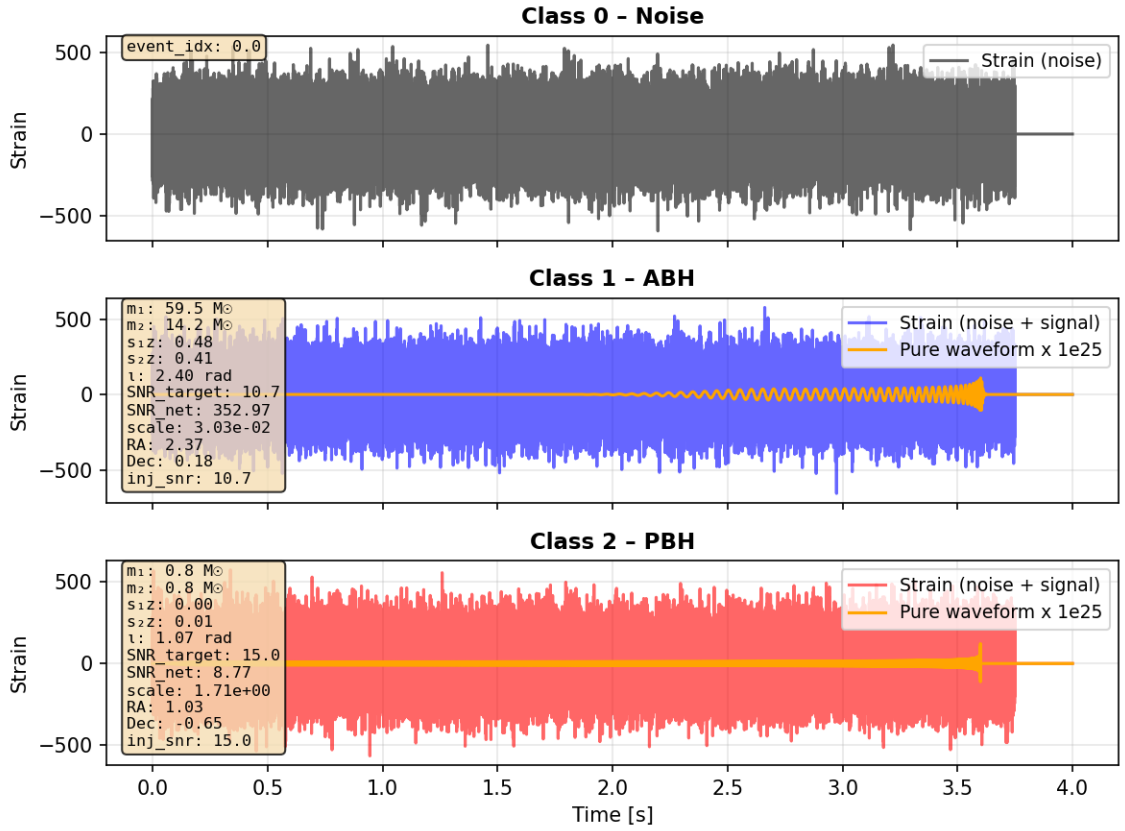


Figure 3.2: GW detector strain for all three classes: noise-only, ABH and PBH. Pure waveform amplitude increased arbitrarily for better visibility (pure waveform on units of the detector strain before whitening).

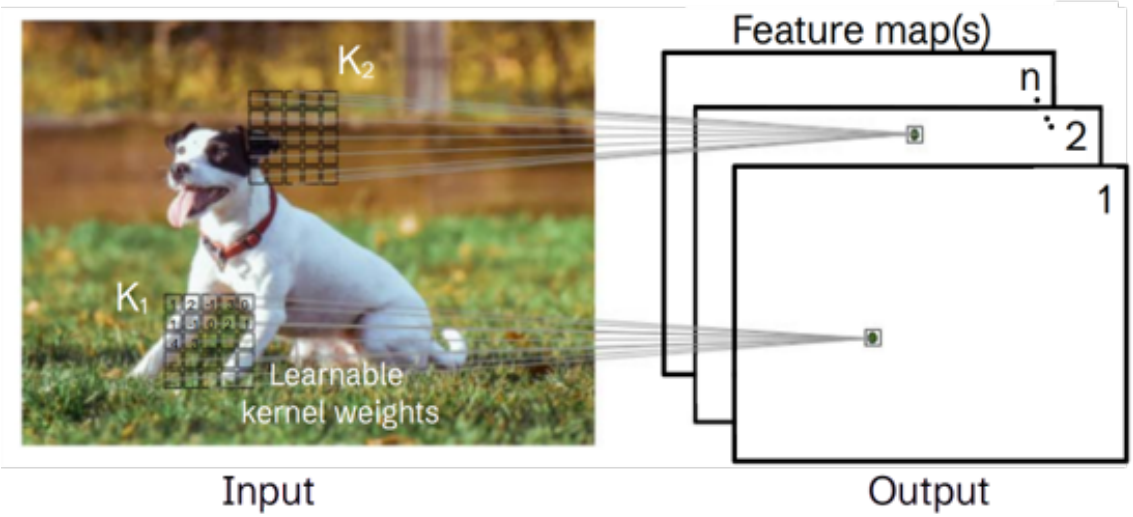


Figure 3.3: Convolutional operation on an input image of a dog. The different features are mapped out and collected in the output. K_1 and K_2 are kernels scanning for different features. Adapted from [19].

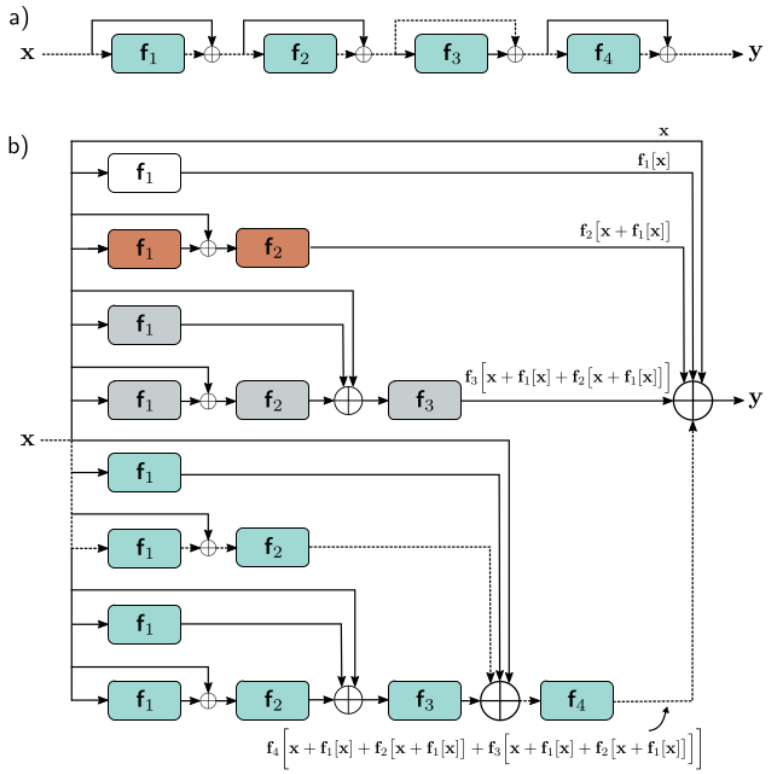


Figure 3.4: Residual connections up to four layers $f_k[x, \phi_k]$. The output of each function is added back to its input. This means that the final output is the sum of the input plus four smaller networks [16].

to a CNN would further increase its performance, as it looks for more underlying connections and further optimises the parameters to fit the model. However, as it turns out, beyond a certain number of layers, the performance of a CNN decreases [16]. ResNets use residual connections between the layers to solve this problem. If, for example, a network has a first layer $f_1[x, \phi]$ and the intermediate quantity $h_1 = x + f_1[x, \phi]$, where x is the input, then the quantities h_k are defined such that $h_k = h_{k-1} + f_k[h_{k-1}, \phi_k]$ [16]. This essentially implies that every output is added to the original input x of the model. This allows the ResNet to learn which layers to put a higher weight on and which layers to skip during training, if a layer is becoming too deep. An example can be seen in Fig. 3.4, in which there are multiple paths leading to the output of the model. If each layer is used, the final output is the addition of each output of each layer with their residual connections. The advantage here is that if the output of the model is more accurate after three layers, than it is after four, then there is the option to skip the fourth layer and instead use the output after three layers as the overall output of the model.

3.3 Autoencoder

Autoencoders are used to reconstruct the input from a learnt compressed latent (hidden) space [23]. As can be seen in Fig. 3.5, if the input is an image, the autoencoders output would be an image that is as similar as possible to the input. The autoencoder consists of two main components: An encoder that transforms the

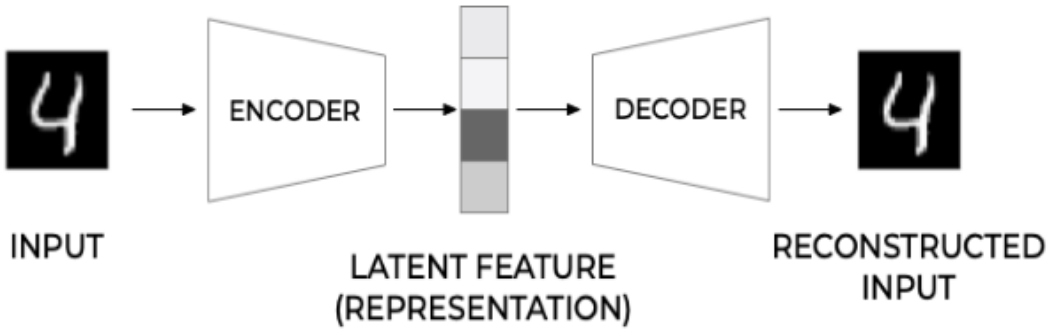


Figure 3.5: General structure of an autoencoder [23].

input into a latent feature representation and a decoder, which is typically a mirror image of the encoder. The latent feature representation is a tensor of real numbers and holds information for the model in the form of parameters. Essentially, this makes the latent representation a dimensionally compressed summary of the input tensor. The parameters are encoded such that they are not easily interpretable by humans and need to be decoded by the decoder. Typically, the encoder and decoder are neural networks [23]. Depending on the task, the neural networks and architectures can be adjusted accordingly. Training an autoencoder can be especially challenging, considering that the encoder and decoder need to work together. The summary in the latent space has to be meaningful enough to reproduce the input.

4 Methodology

This chapter outlines the methodology for creating the DNNs that can read the noisy detector strain and find signals of GWs. The first step is to simulate the detector strain data expected to be seen with ET. Afterwards, the deep learning approach and the steps to evaluate the DNNs will be described.

4.1 Data Simulation

In order to generate the datasets, the PyCBC [25] library was used to implement different parts of the process. The goal is to create datasets containing multiple events of each class "noise-only", "ABH" and "PBH". Each class needs a label assigned for training. Noise-only events are class 0, ABHs class 1 and PBHs class 2. The noise-only events include only the detector strain and time information. Furthermore, these events are Gaussian noise events and are uncorrelated for simplicity, coloured by the ET **Power Spectral Density** (PSD), with the goal of simulating background noise of real measurements. The PSD describes the power of the noise per unit frequency. In Fig. 4.1 the **Amplitude Spectral Density** (ASD) of ET can be seen. The ASD is the square root of the PSD. The ABH and PBH events each include the pure waveform and the pure waveform injected into the noise events, creating the detector strain. These two classes are differentiated by their masses and spins. The ABHs have a possible mass range of (5–80) $\mathrm{M}_\odot$ and a spin range from 0–0.998, where 0 means no spin and 1 is the maximum spin in theory; however, 0.998 is the maximum in practice due to radiative effects. The PBHs have a mass range of (0.2–1) $\mathrm{M}_\odot$ and near-zero spins 0–0.01. The masses are constrained so that the primary mass is greater than or equal to the secondary mass in the binary ($m_1 \geq m_2$) by convention. Going further down the mass range for PBH would lead to a rapid increase in the frequency content. The maximum frequency for GWs emitted by a binary of black holes can be calculated according to the formula [26, p. 175]:

$$f_{\max} \approx 4400\,\mathrm{Hz}\,\frac{M_\odot}{M}, \tag{4.1}$$

where M is the total mass of the binary. Simulating a binary of PBHs with $0.2\,\mathrm{M}_\odot$ would result in a maximum frequency during the merger of roughly 11 kHz, while a binary of $0.1\,\mathrm{M}_\odot$ PBHs has frequencies of up to 22 kHz. The increase in the maximum frequency against the chirp mass of the binary [26, p. 168]

$$\mathcal{M} = \frac{(m_1 m_2)^{3/5}}{(m_1 + m_2)^{1/5}} \tag{4.2}$$

can be seen in Fig. 4.2.
The events are simulated at a distance of 500 kpc. The **signal-to-noise ratio** (SNR)

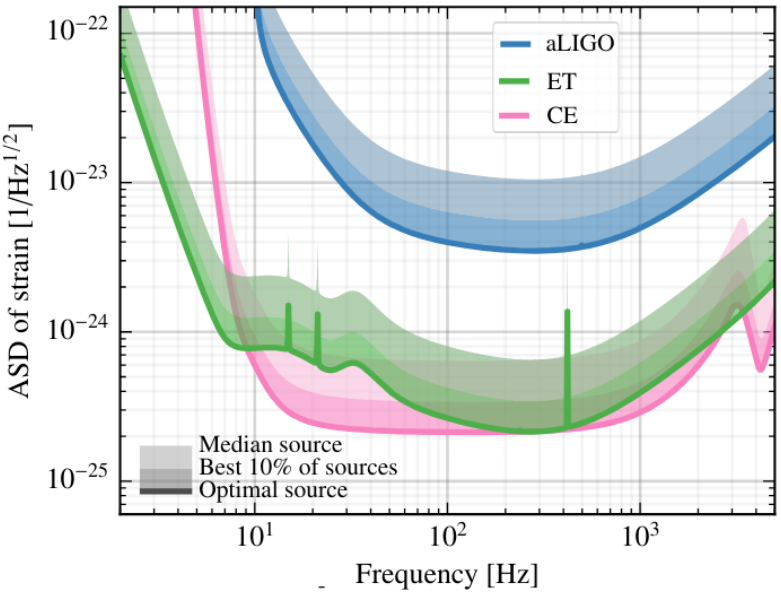


Figure 4.1: The ET Amplitude Spectral Density in comparison to advanced LIGO and Cosmic Explorer [24].

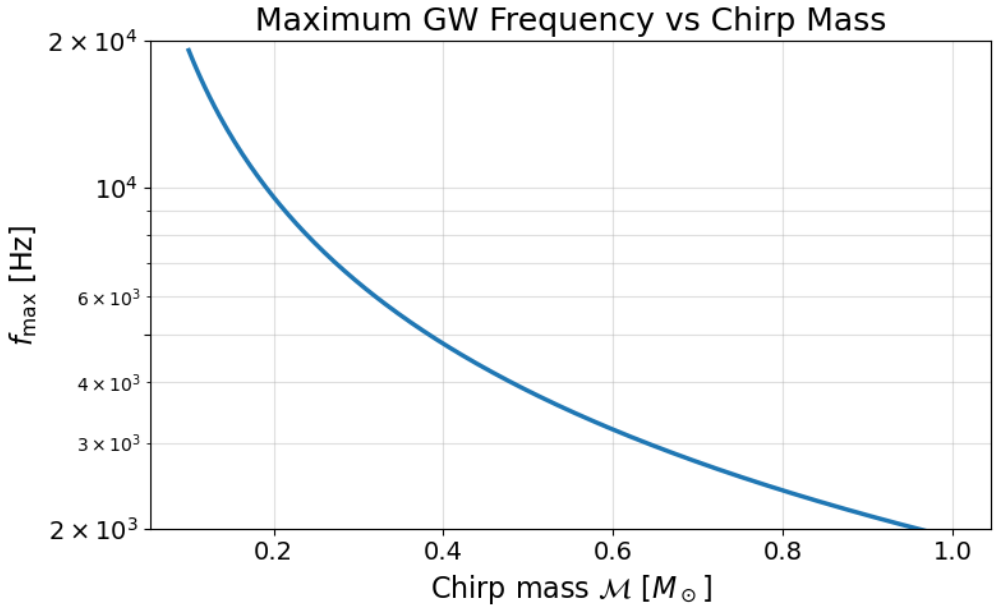


Figure 4.2: The maximum frequency f_{max} reached in the merger part plotted against the chirp mass of the binary $\mathcal{M}$.

values range from 5–20 and across all the datasets, the parameters are uniformly distributed (cf. Fig. 4.5). The SNR is a property that quantifies how detectable a signal event is when applying methods such as matched filtering [27], which distinguishes signals from noise by correlating the detector data with an expected waveform, weighted by the detector's noise characteristics (the PSD) over the entire frequency range [28]:

$$\rho^2 = 4 \int_0^\infty \frac{|\tilde{h}(f)|^2}{S_n(f)} df. \tag{4.3}$$

Here, ρ is the SNR, $\tilde{h}(f)$ the Fourier-transformed signal and $S_n(f)$ the detector PSD. Although it is called a "signal-to-noise ratio", it is not directly the ratio of the signal amplitude to the noise amplitude. The influence of the SNR on the signal can be seen in Fig. 4.3. With the increase of the SNR, the amplitude of the pure waveform becomes larger within the strain, but even at an SNR of 1000, the signal does not overwhelm the noise amplitude. Additionally, by varying the SNR, it effectively varies the distance of the event.
For a good resolution, the sampling rate is set at 32768 Hz. Setting it higher is technically possible, but each dataset would take up much more storage for the same number of events. The data is further simplified to a window of 4 s around the merger for the same reasons. The window is generated such that the merger starts 0.4 s before the end of the window. More specifically, the noise and ABH events were fully generated, and then the window around the merger was cut from there. It cannot work the same way for PBH, because simulating the entire waveform can be very difficult due to the high frequency content and possibly long inspiral phase. This is why the PBH parameters were generated first; then a post-Newtonian approximant (approximating with Newtonian gravity) was used to generate the waveform. Then, the lowest frequency of the inspiral part in a 4 s window around the merger was calculated. Finally, the waveform was generated from that frequency onwards with a more accurate approximant. Additionally, the data is simulated for the triangular configuration of ET, which means there are three separate detector responses for each event. For each dataset, there are 4096 events per class, meaning 12288 in total, which requires 30 GB storage per dataset. Optimally, as many datasets as possible, if not millions of datasets, are required to train a model. Making the waveform any longer would take up too much storage to create large-scale datasets. Increasing the length of the samples results in a longer inspiral part, which, although more accurate for detectors optimised for lower frequencies, contains less valuable information for the model and takes up more storage. In the next steps, the detector strain is "whitened" [28]:

$$\tilde{x}_w(f) = \frac{\tilde{x}(f)}{\sqrt{S_n(f)}}. \tag{4.4}$$

Here, $\tilde{x}_w(f)$ is the whitened detector strain, $\tilde{x}(f)$ the detector strain before whitening and $S_n(f)$ the estimated PSD from the given strain. Because detector noise is coloured, certain frequency bands are much noisier than others. Whitening emphasises less noisy frequency ranges, where the detector is most sensitive, by suppressing the highly noisy frequency ranges. This levels out the background noise across the

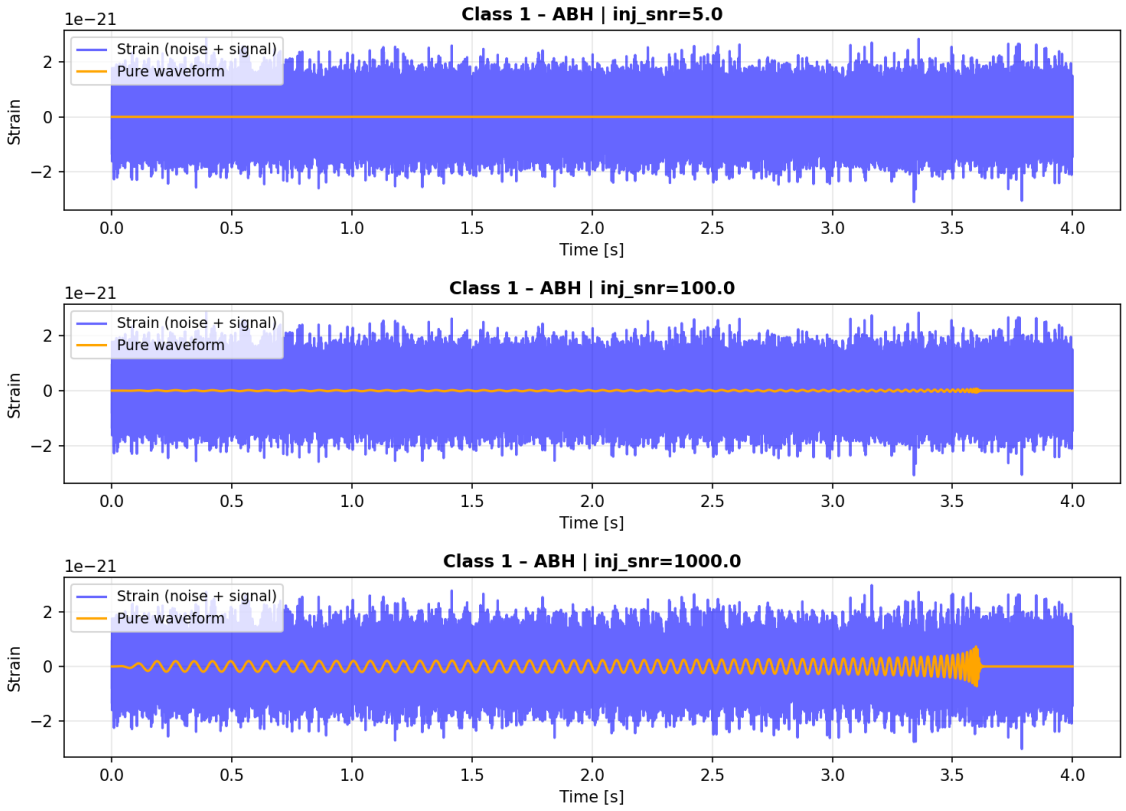


Figure 4.3: Pure waveform injected in the noise for three cases of the same signal. Top: SNR = 5; Middle: SNR = 100; Bottom: SNR = 1000

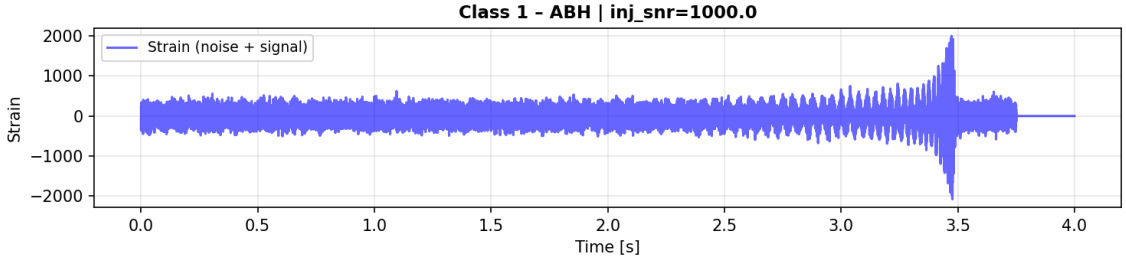


Figure 4.4: Whitened detector strain of the signal with SNR = 1000.

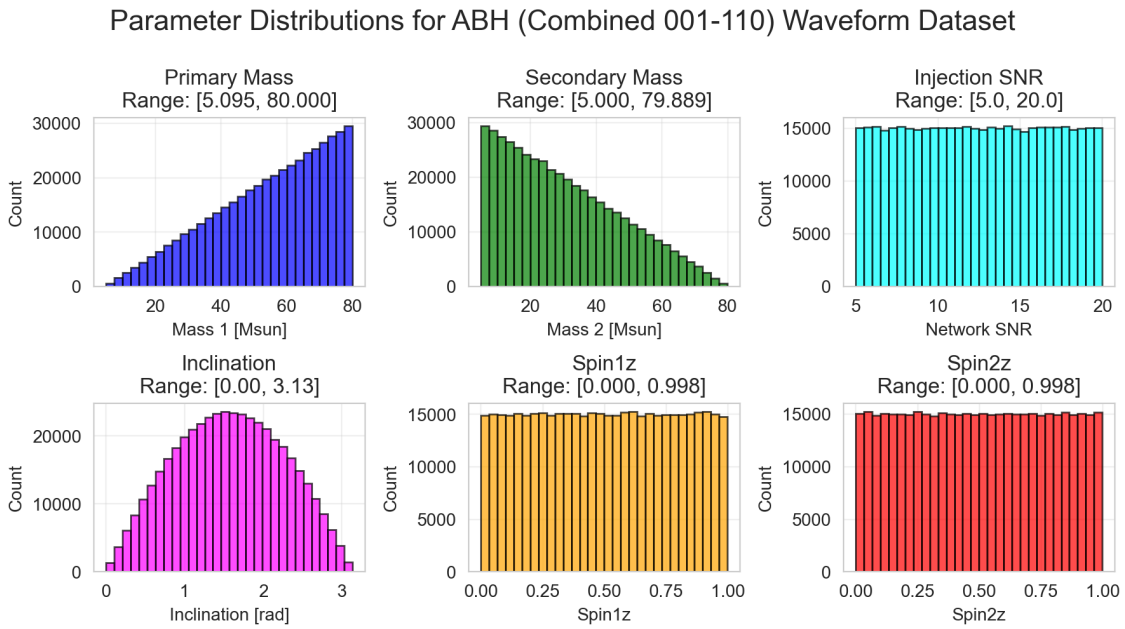


Figure 4.5: Parameter distribution for the ABH events in all datasets resulting in a uniform distribution. Parameter distribution for PBH is the same, just for the different parameter ranges.

entire frequency range of the data. An example of the whitened strain in comparison to non-whitened strain can be seen in Fig. 4.4. A bandpass filter is applied during the training to match the frequency range of the GW signals, in this case 20 Hz-5 kHz. The low sensitivity is not optimal for recreating a realistic simulated dataset for ET because the sensitivity is expected to go as low as 3 Hz or lower, but it saves a lot of computational power needed for other tasks during training. In summary, each dataset contains the detector strain of each event for all three detectors, the pure waveforms of each ABH and PBH event, a uniform distribution of all parameters such as masses, SNR values, spins and some additional parameters such as inclination, declination, right ascension, polarisation, event time and most notably the class labels of each event. Finally, the detector strain is normalised for training.

4.2 ResNet and Autoencoder

This section introduces the architectures of the models used to classify the events based on the detector strain and to reconstruct the pure waveform based on the parameters estimated by the model. The models were implemented using PyTorch [29] and PyTorch lightning [30] libraries.

4.2.1 Residual Network

The ResNets architecture can be seen in Fig. 4.6 (figures illustrating model architecture were created using AI-assisted diagram tools and manually verified.[1]). It

[1] https://gemini.google.com/

receives the detector strain as the input, and the expected output is the corresponding class label: 0, 1 or 2, making this a multiclass classification problem. The inputs are three channel time-series inputs with sample lengths of $4\,\mathrm{s} \times 32\,\mathrm{kHz} = 131072$ samples. Initial downsampling is handled by a stem of two strided 1D convolutional layers with kernel sizes of 31 and 15. The architecture then continues through three additional strided convolutional stages with kernel size 9 and a residual block. The stride refers to the movement of the kernel over the input, so a stride of 1 means the kernel moves one element at a time. A bigger stride means that the kernel skips a number of elements. In the end, the input is sampled down by a factor of 64. The residual blocks consist of 1D convolutional layers, batch normalisation layers, LeakyReLU activation [29] and skip connections. Unlike ReLU layers, the LeakyReLU layers return a small, but non-zero, gradient for negative inputs. This is to prevent neurons from becoming permanently inactive. In the final stage, an adaptive average pool [29] is used to accumulate the features of the time series before the final linear output layer, which returns the logits representing the probability that one event belongs to a class in the form of a three dimensional vector, where the first element corresponds to class 0, the second to class 1 and the third to class 2. To get the class label, the label with the largest number is assigned as the final output during evaluation. Optionally, when applying softmax [29], it returns a normalised output that can be interpreted as a percentage probability for each class label. The loss function of the model used is "CrossEntropyLoss" (CEL) [29]:

$$l_n = -\sum_{c=1}^{C} w_c \log \frac{\exp x_{n,c}}{\sum_{i=1}^{C} \exp x_{n,i}} y_{n,c}. \tag{4.5}$$

Here, x is the target, y the prediction, w the weight, which is 1 here, C the number of classes and N spans the minibatch dimension. CEL compares the true probability distribution to the predicted probability distribution. In the example of this thesis, each class appears the exact same number of times in the datasets, so beyond some minor statistical fluctuations, the model should predict every class roughly 33.33% of the time. Class imbalance is avoided to prevent the model from favouring over-represented classes. For smoother convergence, the input data are randomly shuffled so that every batch contains representative examples of each class. Otherwise, this would allow the model to be biased towards guessing just one class, especially if this happens early on during training. The specific configurations and parameters for the training can be seen in table 4.1.

Table 4.1: ResNet Configurations and Hyperparameters

Setting	**Value/Description**
Trainable Parameters	2.9 million
Batch Size	32
Learning Rate	0.001
Data Split (Train / Val / Test)	80% / 10% / 10%
Training Duration	2 days
GPU Used	NVIDIA A100 [31]

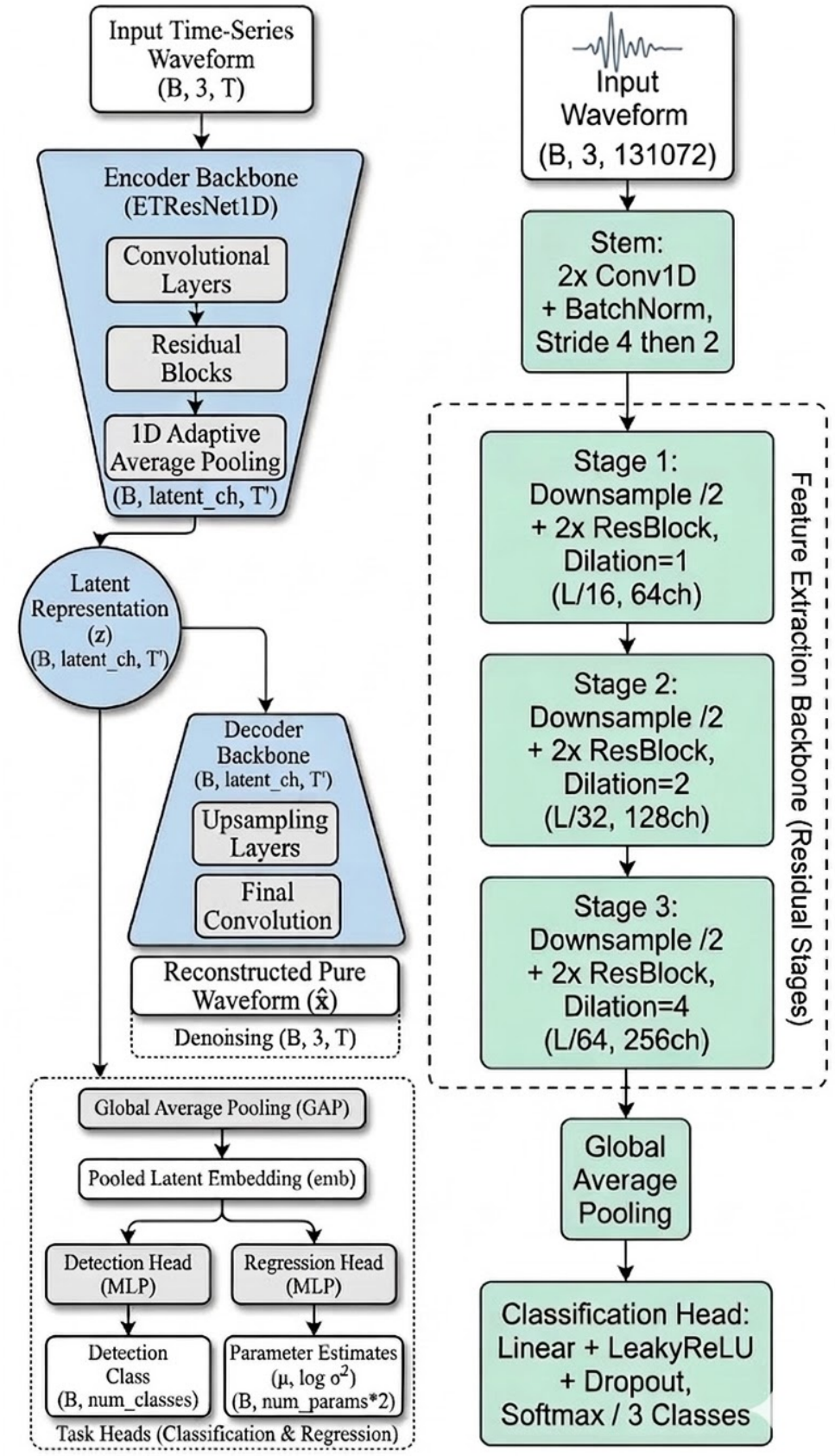


Figure 4.6: Architectures of the autoencoder (left) and ResNet (right).

4.2.2 Autoencoder

The autoencoder architecture extends the ResNet (cf. Fig. 4.6; figures illustrating model architecture were created using AI-assisted diagram tools and manually verified.[2]). The input is the same and the encoder has the same structure as the ResNet. The output of the encoder is projected into the latent space. From the latent space, the parameters are pooled and lead to the detection (det) and regression (reg) heads. The detection head classifies the events and the regression head estimates the parameters of the waveform. The decoder, which has the exact mirrored structure as the encoder, receives the parameters of the latent space as input. From there, the denoising (den) head follows, where the pure waveform is reconstructed with the same sample length for all three detector channels of the input strain. For the three tasks, three loss functions are needed, resulting in the total loss l_{total} being the result of all three loss functions with their weights:

$$l_{\text{total}} = w_{\text{det}} l_{\text{det}} + w_{\text{reg}} l_{\text{reg}} + w_{\text{den}} l_{\text{den}}. \tag{4.6}$$

The detection part uses the same CEL as the ResNet, the denoising part uses mean squared error (MSE) or L1Loss (depending on configuration) [29]:

$$\begin{aligned} l_{\text{MSE}} &= (x_n - y_n)^2 \\ l_{L1} &= |x_n - y_n| \end{aligned} \tag{4.7}$$

and the regression part uses Gaussian negative log likelihood loss [29]:

$$l_{\text{gau}} = \frac{1}{2}\left(\log(\text{var, eps}) + \frac{(x_n - y_n)^2}{\max(\text{var, eps})}\right) + \text{const.}, \tag{4.8}$$

where var is the variance σ^2 and eps is a small constant, to keep the variance above 0. If the difference between the target and prediction becomes large, the loss takes positive values. However, if the difference between the target and the prediction is small, the variance becomes small, causing the log and the overall loss to become negative. Other than that, the same training method from before is applied. The specific configurations and parameters for the training can be seen in table 4.2.

Table 4.2: Autoencoder Configurations and Hyperparameters

Setting	**Value/Description**
Trainable Parameters	2 million
Batch Size	32
Learning Rate	0.0003
Data Split (Train / Val / Test)	80% / 10% / 10%
Training Duration	2 days
GPU Used	NVIDIA A100
Loss Weights	$w_{\text{det}} = 1.0$, $w_{\text{den}} = 1.0$, $w_{\text{reg}} = 0.1$

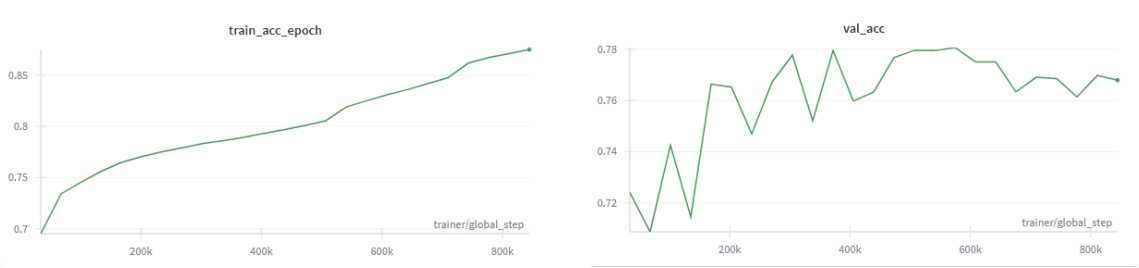
[2] https://gemini.google.com/

5 Results

The goal of this chapter is to compare the accuracy of the two models trained for classification on the same datasets. Furthermore, the possible reasons for the results and how to improve them will be discussed. The regression and denoising parts of the autoencoder will not be presented or discussed, as these are beyond the scope of this thesis.

5.1 ResNet

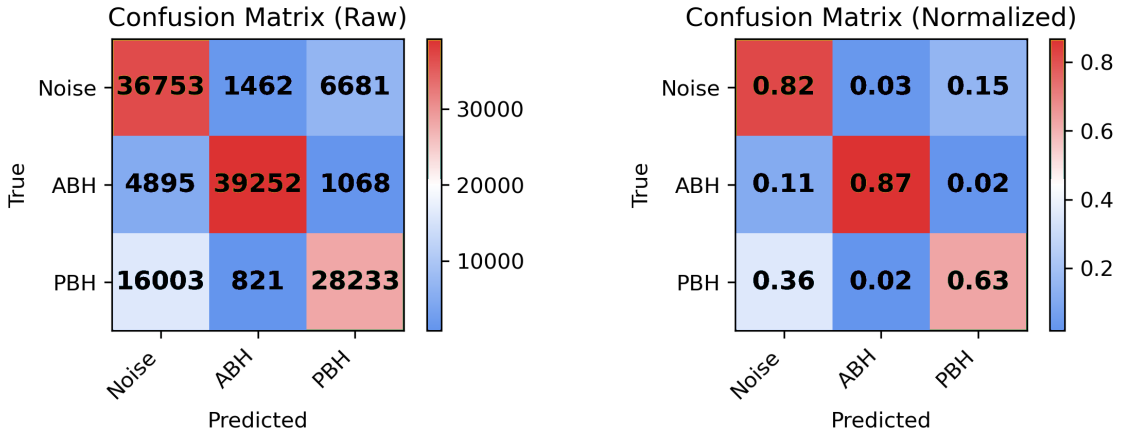
In subsection 2.3.2 of the theory, it has been shown that the xylophone setup leads to an overall improvement in the detection of the parameters, such as the masses and the redshift, of the binaries. This leads one to believe that the overall distinction between a noise event and a signal event will also improve. Being limited by the high computational effort required to simulate extremely low-frequency sensitive detector data, two settings of the same model will be compared. First with the bandpass set from 20 Hz-5 kHz and then from 50 Hz-5 kHz instead. The model starting at 50 Hz was slightly overfitting, as can be seen in Fig. 5.1. The training accuracy was significantly better than the validation accuracy. It performs worse than the 20 Hz model in terms of overall accuracy and the accuracy of predicting each class. This is illustrated in Fig. 5.2, using confusion matrices. The confusion matrix compares the true class label to the model's prediction, so the numbers on the main diagonal are accurate predictions for each class. Additionally, it can be seen in Fig. 5.3 that the higher the SNR, the more likely it is that the model will accurately predict the event, as the accuracy consistently increases with the SNR.
Note that the 50 Hz model has an overall accuracy of 77% and the 20 Hz model even reaches an accuracy above 99%. The PBH accuracy increases from 63% to 100% with this simple change. Paying even closer attention, the model's biggest flaw was differentiating a signal from noise, but it performed very well in classifying the correct signal, which is why there is only a small overlap when predicting one signal class with the other signal class. This leads one to believe that the lower sensitivity improved the model's ability to identify when an event is only detector noise. This



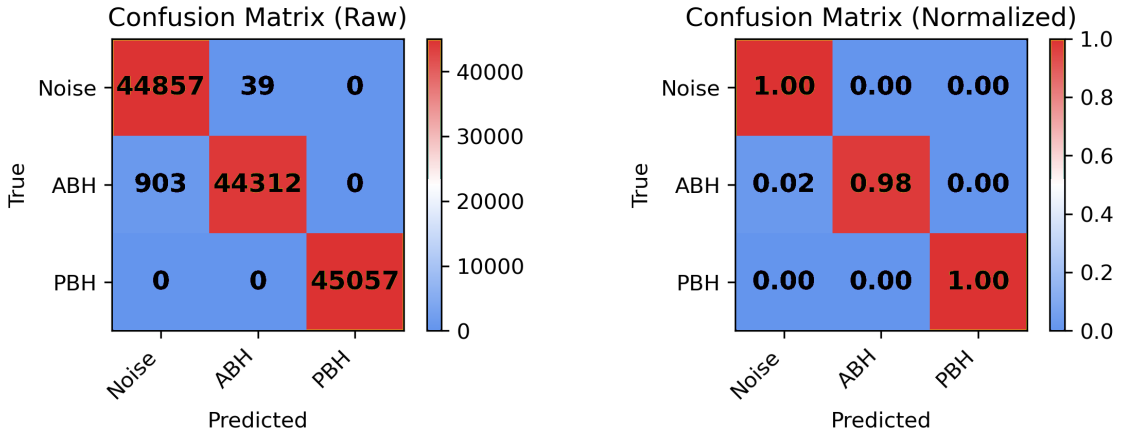
(a) Training accuracy

(b) Validation accuracy

Figure 5.1: Training (a) and validation (b) accuracy as a function of training steps of the model with 50 Hz lower sensitivity.

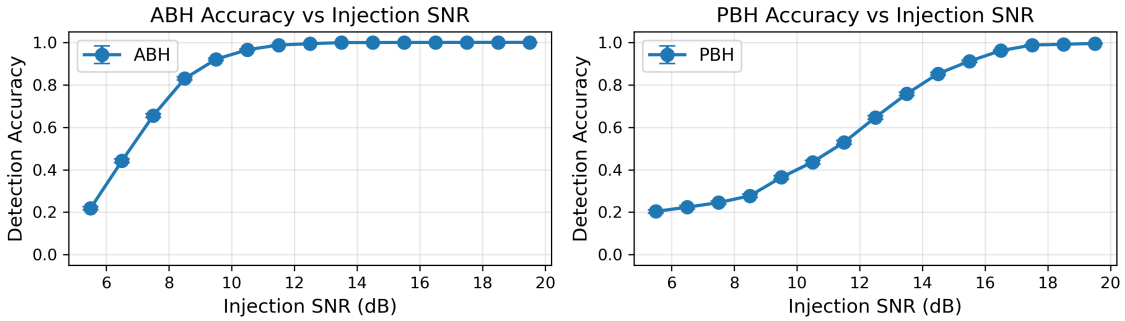


(a) Confusion matrices of the 50 Hz model.

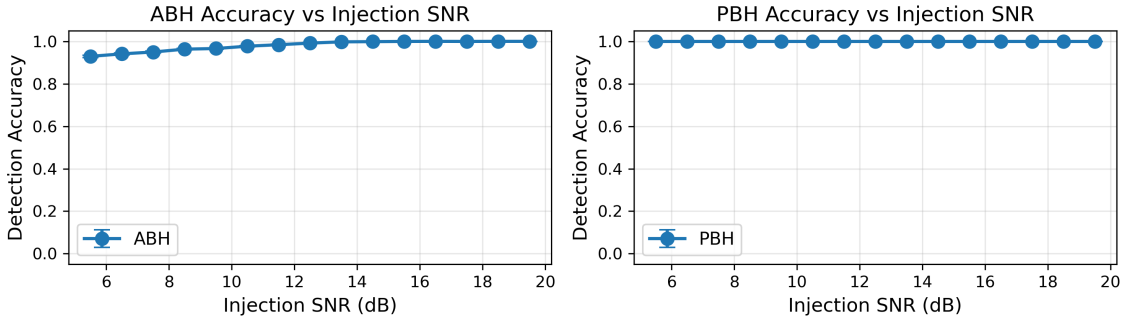


(b) Confusion matrices of the 20 Hz model.

Figure 5.2: Confusion matrices of the two models. Top: 50 Hz model. Bottom: 20 Hz model (left: raw numbers, right: normalised as a number representing a percentage). Testing data used to compare the true class label against the predicted class label.



(a) Accuracy vs SNR of the 50 Hz model.



(b) Accuracy vs SNR of the 20 Hz model.

Figure 5.3: Accuracy vs SNR plot of the two models. Top: 50 Hz model. Bottom: 20 Hz model. Separated accuracy for ABH (left) and PBH (right).

observation would not explain why the model is significantly more accurate for PBHs than for ABHs. The answer could be the high frequency content of the PBH in the merger phase. The maximum frequency of the detector is set to 5 kHz due to the bandpass filter, but the lowest mass binary possible for ABHs would lead to a maximum frequency of 440 Hz calculated using equation 4.1. Since the binary mass for the PBHs is significantly lower, the maximum frequency is overall higher, even exceeding the 5 kHz in some rare cases. The amplitude of the strain also reaches its maximum at the merger. In conclusion, the model receives two major clues during the merger phase: the amplitude and the frequency of the GWs. This could allow for a confident classification of the events. The frequency of the PBH events before the merger is also a lot higher than the maximum frequency of the ABH events. In the average case of a PBH binary, with binary mass $M = 1.2\,\mathrm{M}_\odot$, the maximum frequency becomes ~ 3666 Hz, which means they usually do not exceed the bandpass upper limit. This means that this high accuracy in the detection of PBH is likely not correlated to the bandpass clipping the event's frequency before the merger. Further discussions on possible reasons for the high accuracy of the PBH classification will follow in the next section. Several simplifications were made for these datasets and it is not simple to say how this model would perform for a waveform with a long inspiral phase, before eventually measuring the merger, once the Einstein Telescope is built and real data can be measured. Still, this provides a proof of concept for this model and shows that the low-frequency sensitivity does indeed improve the detection and classification of events, even with noisy detector strain.

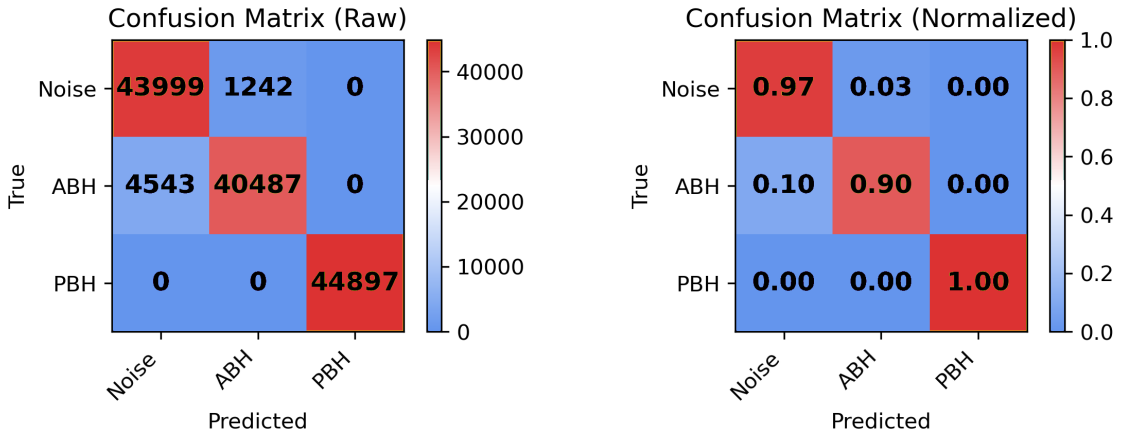


Figure 5.4: Confusion matrices of the autoencoder. (left: raw numbers, right: normalised as a number representing a percentage). Testing data used to compare the true class label against the predicted class label.

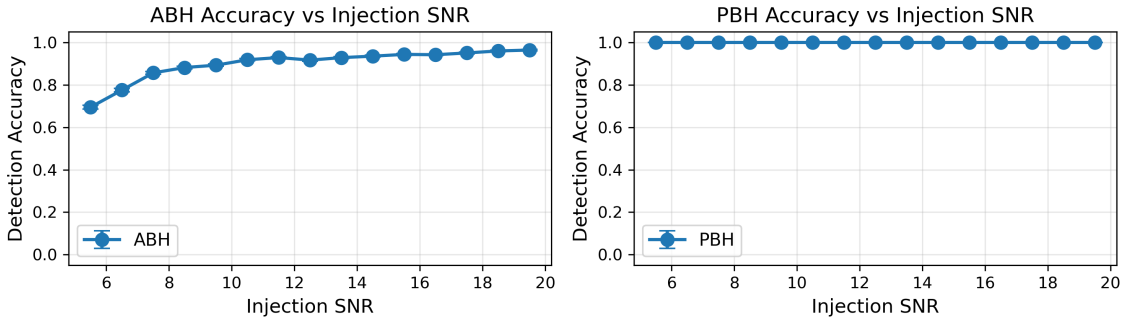


Figure 5.5: Accuracy vs SNR plot of the autoencoder. Separated accuracy for ABH (left) and PBH (right).

5.2 Autoencoder

In this section, the results of the autoencoder will be evaluated and more context on the suspiciously high accuracy for the PBH events will be provided. The bandpass filter is now set at a lower frequency of 20 Hz. The overall accuracy of the model is 95.8%. In Fig. 5.4, it can be seen that the accuracy for the detection of PBH is again at 100%. The accuracy for the noise and ABH detection slightly decreased in comparison to the ResNet. In the following, different possible reasons for the high accuracy of the PBH classifications will be examined.
In Fig. 5.5, it can be seen that the accuracy remains at 100% throughout every SNR value in the given range. The first idea is to lower the SNR, until the accuracy goes down or the signal becomes undetectable entirely. Unfortunately, even an SNR range as low as 0.001-0.01 could not decrease the model's accuracy for PBH, even though it is expected that the events were not detectable. On the other hand, the ABH classification accuracy dropped to nearly 0% as expected (cf. Fig. 5.6).
This raises a subsequent question regarding the possibility of data leakage during the training phase, specifically whether the model was exposed to the test data. If that were the case, it would not just apply to PBH events, but to every class, since the data is loaded in the same way for each class, by inputting the strain from

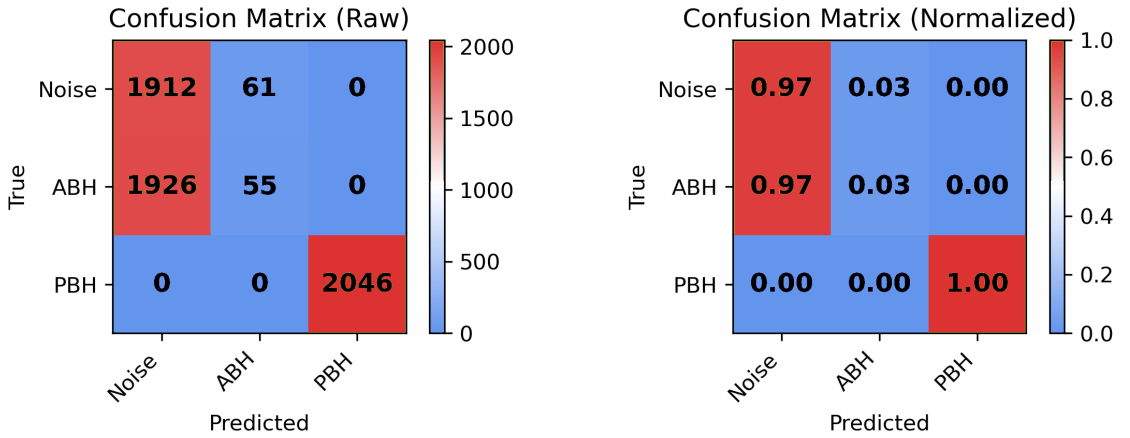


Figure 5.6: Confusion matrices of the autoencoder for a test set with SNR values 0.001-0.01.

the datasets. On top of that, test sets separate from the training sets were used to verify this and the accuracy remains 100%. To determine whether there is any minuscule uncertainty, a histogram was created to show the true detection value as a probability for each class. Essentially, the histogram shows how certain the model is in the classification of an event, by counting the probabilities with which the model classified each event. As one can see in Fig. 5.7, there are only 2 large peaks. At 0, the ABH and noise peaks overlap and at 1, there is the PBH peak. This means that the classification is truly very close to 100% accurate, as there is barely a fraction of a percentage that the model is not certain whenever it classifies a PBH event, which is indicated by the smaller peak beside the large PBH and ABH peaks. For the sake of comparison, an example can be seen in Fig. 5.8 for the ABH classification, which exhibits a more expected behaviour. Even in this histogram, there is only a minor distribution of the predictions of PBH events beyond 0%. This observation is consistent with the confusion matrix, as there is not a single ABH event that has been misclassified as a PBH event. These fluctuations in accuracy could be explained by some of the extreme cases that are highly unlikely to appear in the datasets, such as a PBH binary with masses $m_1 = m_2 = 0.2\,\mathrm{M}_\odot$, which were possibly generated because of the sheer number of events generated overall, but are statistically unlikely. It is concluded that there is most likely not a bug where the class label is being passed to the model during testing. Since decreasing the SNR all the way did not change the accuracy, the next idea is to generate a few datasets where the PBH and ABH mass ranges are the same (5-80) $\mathrm{M}_\odot$. Without any further illustrations, it was found that the PBH accuracy remains at 100%, which means that the model is learning something that is unique to the PBH events.
At this point, the data generation needs to be examined. The PBHs being the only 100% accurate class could be correlated to how the 4 s window around the merger was created. To verify this possibility, another 110 datasets were created, where the windows for all the events were generated in the same way as the PBH events. In Fig. 5.9, there is a visual difference from the previous method (cf. Fig. 3.2 p. 13), as the pure waveform from the ABH event has a seemingly longer inspiral part. Before this change, the lower cut-off frequency for ABH events was set at 20 Hz, which means the first part of the inspiral of that pure waveform was clipped early. This

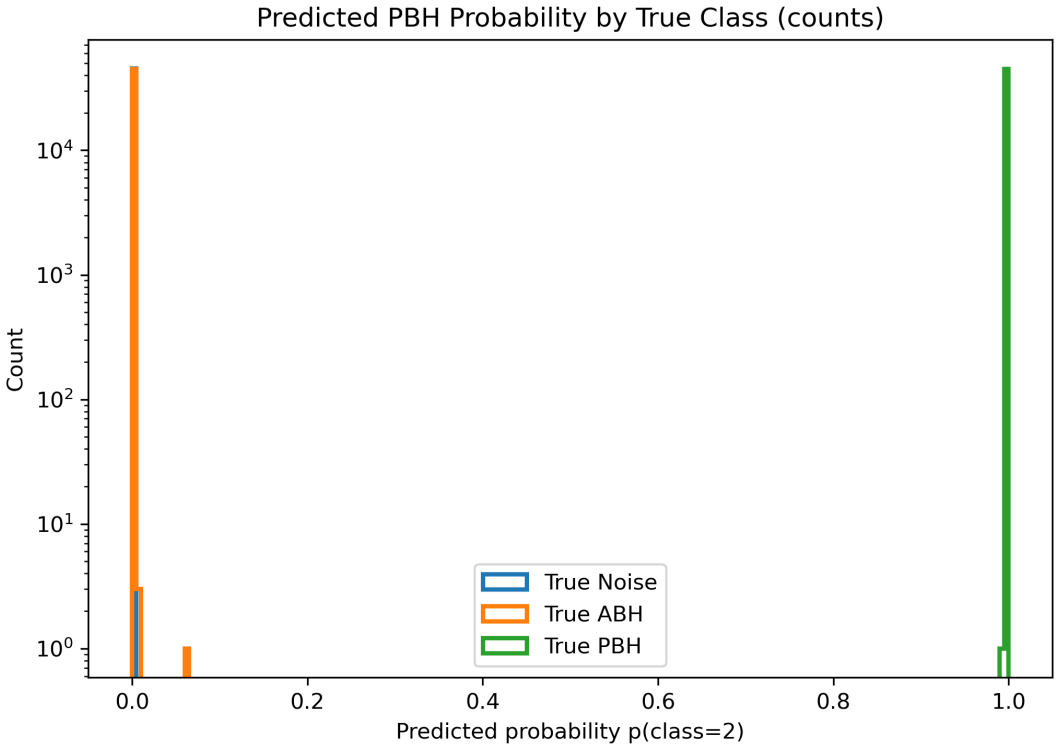


Figure 5.7: Probability distribution of the PBH classification events. Counts in log scale vs predicted probability for PBH events (class = 2). SNR 5-20

.

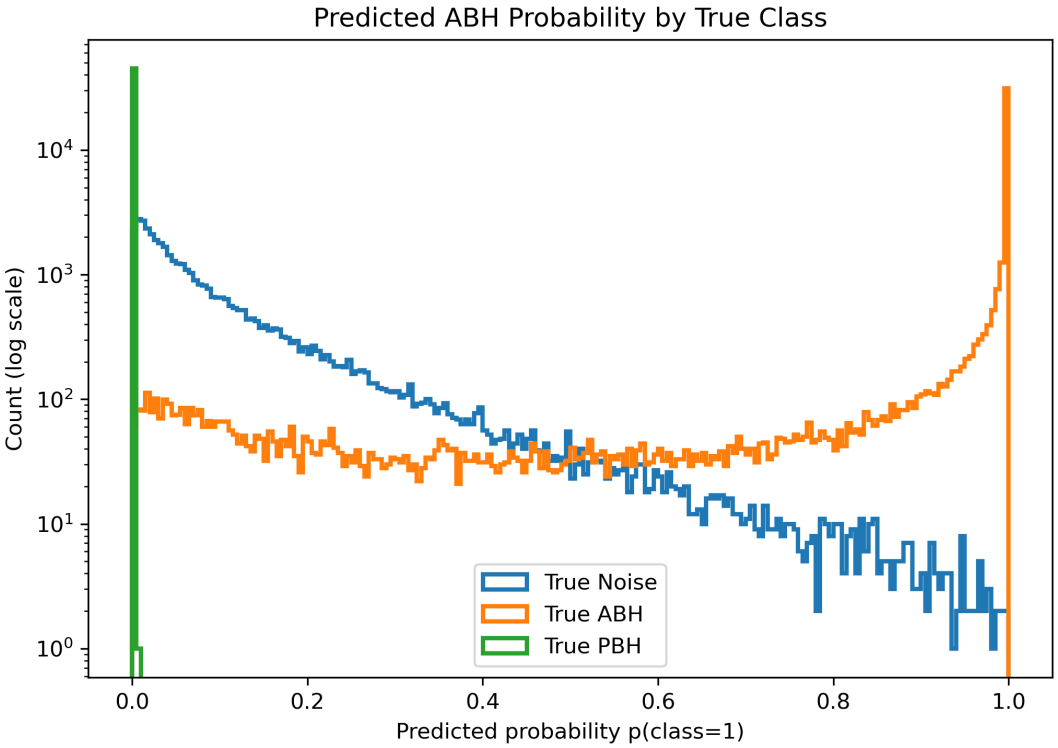


Figure 5.8: Probability distribution of the ABH classification events. Counts in log scale vs predicted probability for the ABH events (class = 1). SNR 5-20

.

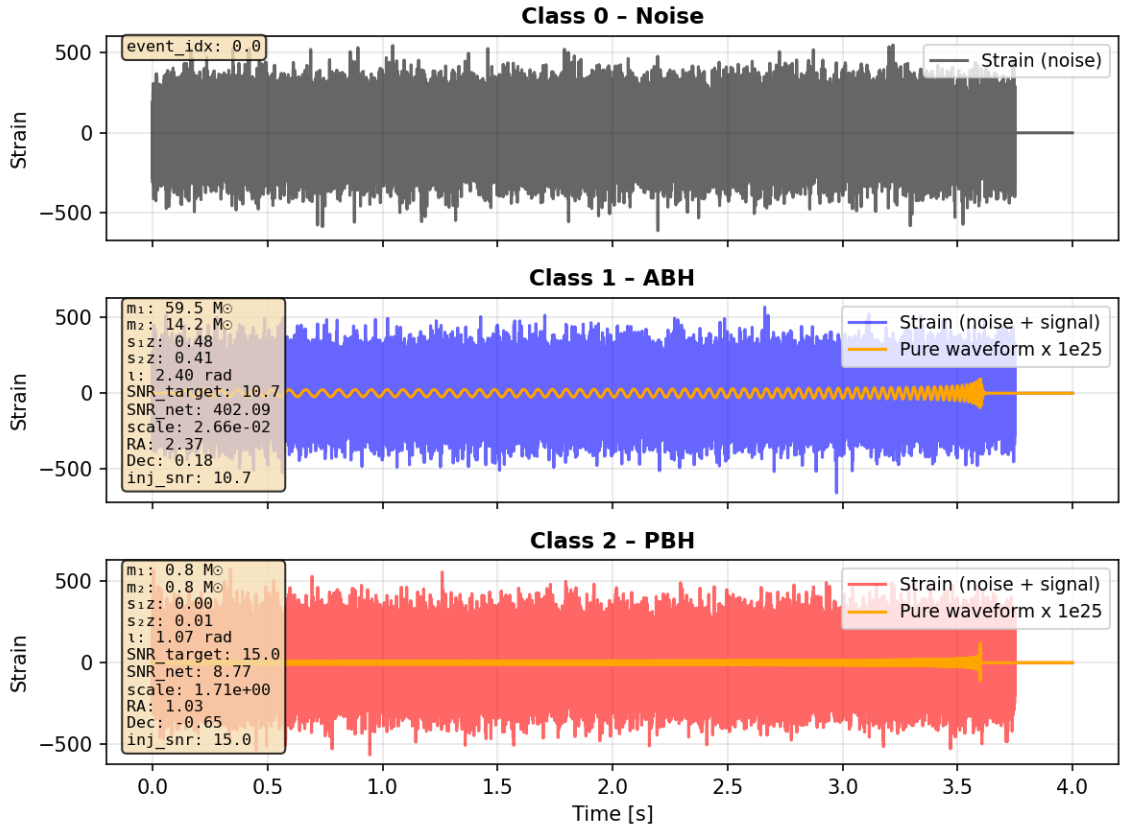


Figure 5.9: GW detector strain for all three classes. Pure waveform amplitude increased arbitrarily for better visibility (pure waveform on units of the detector strain before whitening).

also explains why the models were always slightly inaccurate at classifying ABH and noise signals. To keep in mind, the illustrated events are exactly the same as the ones before. The idea is to get a conceptual understanding of what the difference in the data generation could change, but there is no way to compare every single pure waveform of every event. In the next step, the influence of this change on the models' performances is examined. The first thing checked with the new datasets is what the model, trained on the previous datasets, will predict (cf. Fig. 5.10). The model now seems to predict that every event is a PBH binary. This confirms the suspicion that the model learnt something about the way the window was generated around the merger and connected this to the PBH class.

The accuracy of the ResNet, after being trained on the new datasets, can be seen in Fig. 5.11. Its accuracy is very high for every class, but it now misclassifies some of the events and is not exactly at 100% accuracy for PBH. One could now assume that the high frequency content of the PBH events leads to a more accurate detection than the ABH events. The next step is to train and evaluate the autoencoder with these datasets.

The autoencoder's loss weights have been adapted before training. The detection loss weight was set to 0.1, while the regression loss weight was set to 2.0. This is due to the fact that the autoencoder classified every event as a PBH event during training with the new datasets. The results can be improved by testing different configurations, but for now, these are the settings that will be used to train the

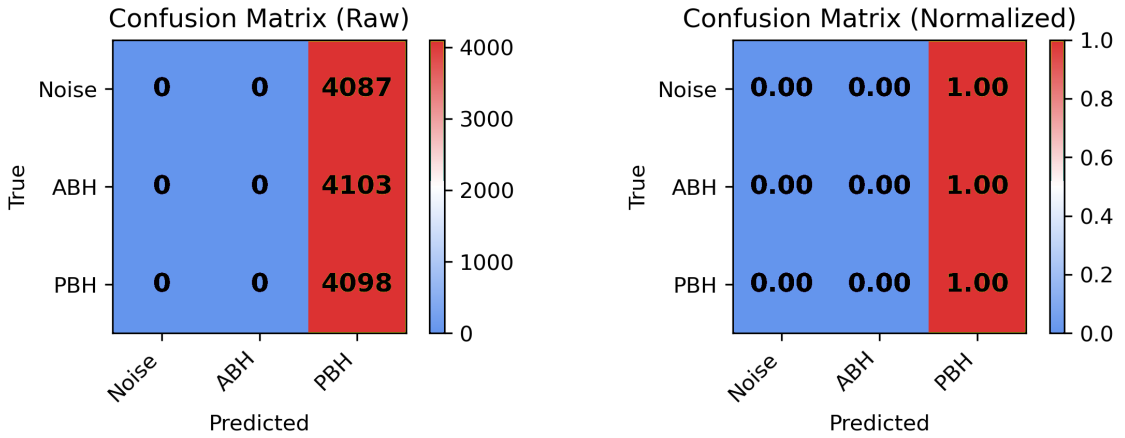


Figure 5.10: Confusion matrices of the autoencoder trained on the previous datasets, tested with the new datasets.

autoencoder. In Fig. 5.12, it is clear that the autoencoder still finds a pattern to reach 100% accuracy for PBH events, but it has overall become more inaccurate for the other classes. Possible reasons for the autoencoder performing worse than the ResNet include the lower number of trainable parameters and the challenge of balancing multiple tasks, such as regression, denoising and detection.
To further investigate the learnt latent representations, a t-distributed stochastic neighbour embedding (t-SNE) [32] visualisation is presented in Fig. 5.13. t-SNE is a nonlinear dimensionality-reduction technique that projects high-dimensional feature representations into a low-dimensional space while approximately preserving local neighbourhood relationships between data points [33]. The resulting embedding reveals a substantial overlap between ABH and noise-only events, alongside a clear separation from PBH events. This structure provides insight into the model's behaviour: the strong separability of PBH events suggests that distinctive signal characteristics, such as enhanced high-frequency content, are readily captured by the latent representation. In contrast, the similarity between ABH and noise events likely contributes to the reduced classification accuracy for these classes.
Moreover, because the detection objective carries only a small loss weight, the model

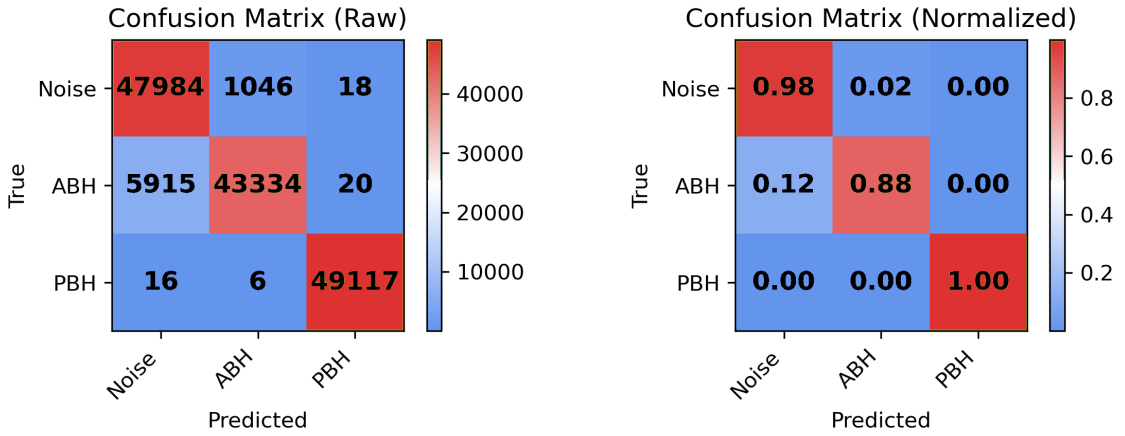


Figure 5.11: Confusion matrices of the ResNet trained on the new datasets.

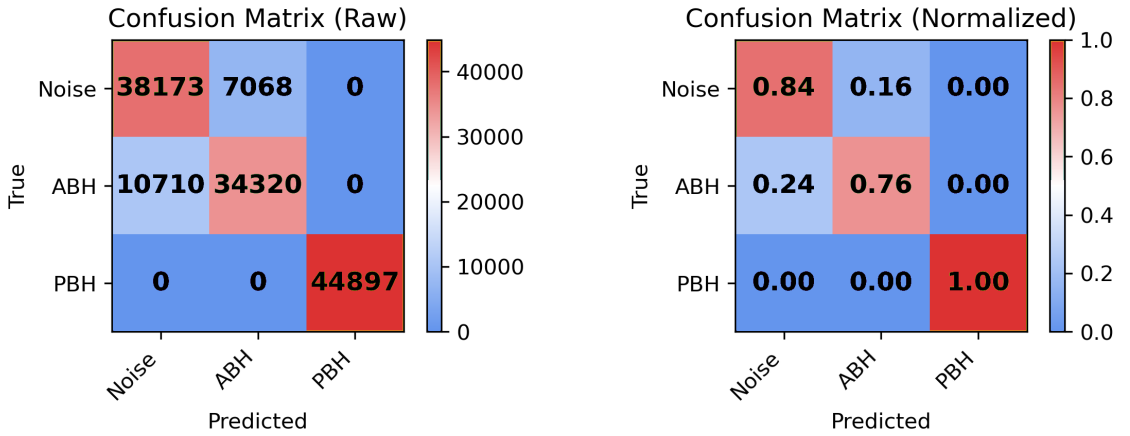


Figure 5.12: Confusion matrices of the autoencoder trained on the new datasets.

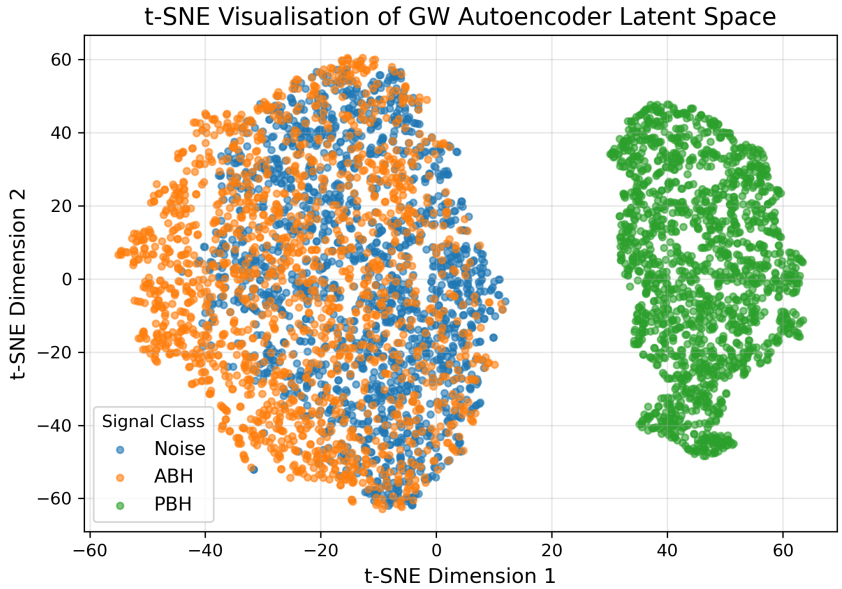
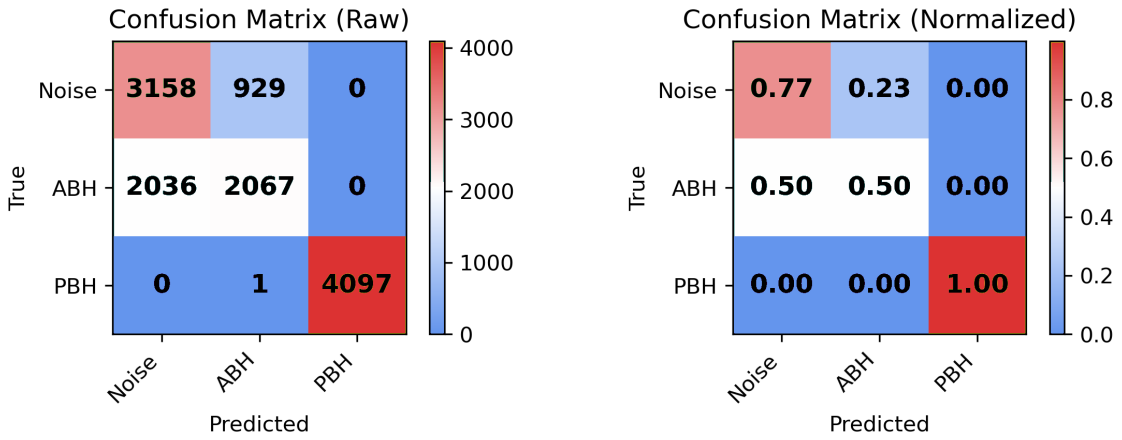


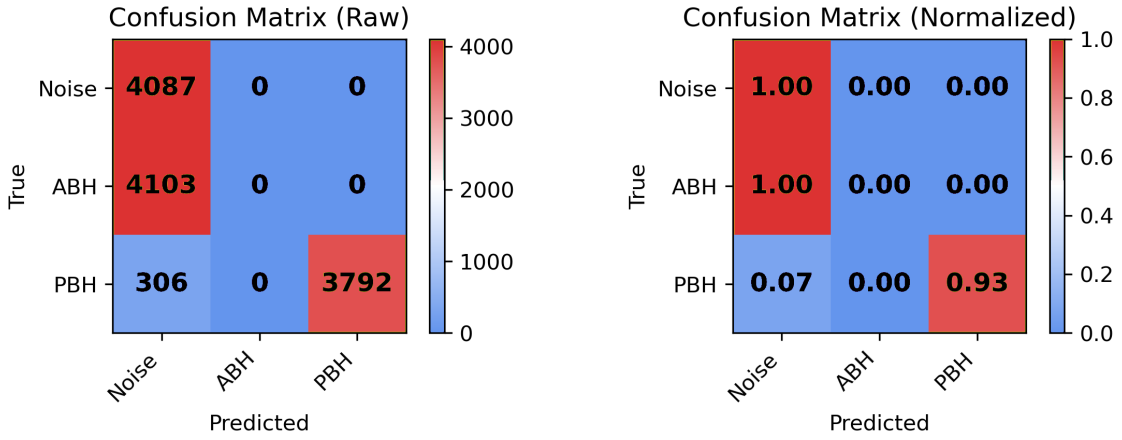
Figure 5.13: Visualisation of the parameters of the latent space in a t-SNE plot for the different classes.

is not strongly constrained to learn a fully discriminative representation of the signals. This may further explain why detection performance remains relatively robust, while regression accuracy and denoising quality are comparatively degraded.
To make the task harder, the autoencoder will be trained on a test set of 10 datasets, in which the mass range of the ABH is set to (1-80) $M_\odot$. Although this is not accurate for the physics of this thesis anymore, it means that there is now a direct transition of the mass range from PBHs to ABHs, instead of the $4\,M_\odot$ gap between the highest PBH and the lowest ABH mass. Additionally, the model will be trained once with and once without bandpass filtering, as it is not clear whether the filter plays a role in the model's accuracy. In Fig. 5.14, a clear difference can be seen between the two models. These are not easy to explain because, even though the mass range of the ABH events is closer to the PBH events, there is only a single PBH event misclassified as an ABH event in the case of using the bandpass filter. For the prediction of ABH events, the split between ABH and noise only events has become 50%. The overall

worse accuracy is expected, as there were fewer datasets for training, resulting in an overall shorter training time and a less accurate model. Especially surprising is the model without bandpass filtering. It cannot distinguish the ABH events from noise, but it is still very accurate in classifying PBH events, although now 7% were misclassified as noise. Clearly, the bandpass filtering influences the 100% accuracy for PBH classification. Looking at the t-SNE plot of the model with the bandpass filter 5.15, there are two populations again, as seen in the previous t-SNE plot. The ABH and noise events overlap, while the PBH events can be seen distinctly. One would expect that the PBH distribution would at least be closer to the ABH distribution, due to the direct transition of the mass priors. Since that is not the case, this raises the question if there is a systematic error. Finally, the autoencoder will be trained on a test set of 10 datasets with SNR values 0.001-0.01 at the normal mass range again. As before, this should make the signal undetectable. Last time, the suspicion was that the way the window is generated around the merger is causing the high accuracy. In Fig. 5.16, it can be seen that the PBH events remain at 100% accuracy, despite the changes. Additionally, it can be seen that no matter which dataset the autoencoder is trained and evaluated on, it will always have 100% accuracy for the PBH classification. These findings strongly suggest the presence of anomalies within the PBH dataset. There could be many things such as artifacts on the edges of the signals from the data simulation or an error in the overall methodology of the data simulation and training process. In retrospect, it would be a good idea to test different frequencies with the bandpass setting in the higher frequency range. The next steps for these models are to further investigate the data simulation and examine what the model is paying attention to when classifying the signals in an attempt to clarify these results. These objectives are beyond the scope of this thesis due to a lack of time. With everything that has been found in this section, it is highly likely that there is a systematic error that leads to the high PBH accuracy.



(a) Bandpass filtering applied



(b) No bandpass filtering applied

Figure 5.14: Confusion matrices of the autoencoder trained on 10 datasets with the ABH mass range set from (1-80) $M_{\odot}$. Top: with bandpass; Bottom: without bandpass

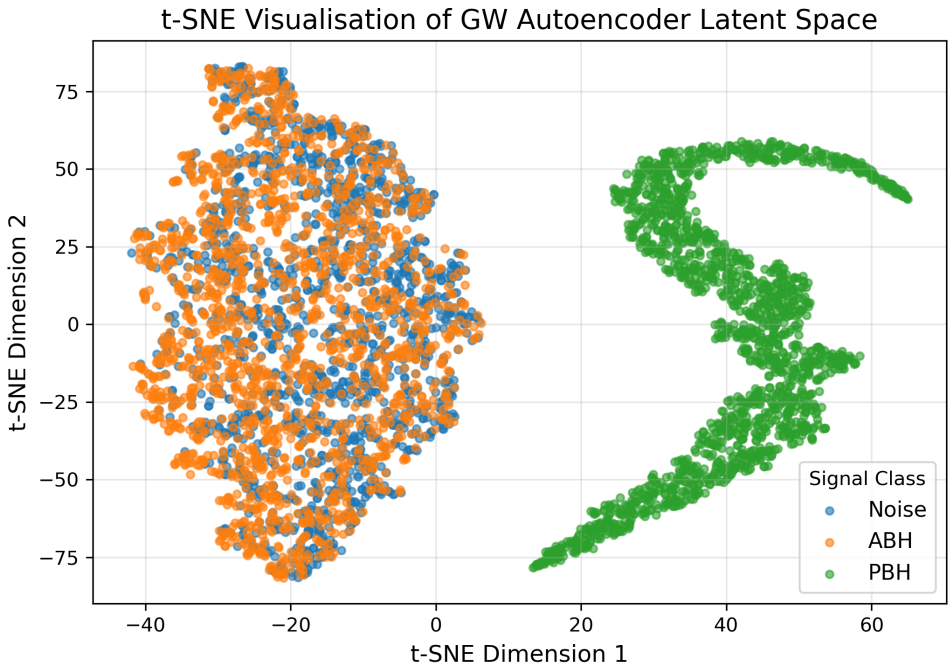


Figure 5.15: Visualisation of the parameters of the latent space in a t-SNE plot for the different classes. ABH mass prior set to (1-80) $M_\odot$, model trained on 10 datasets and bandpass filter applied.

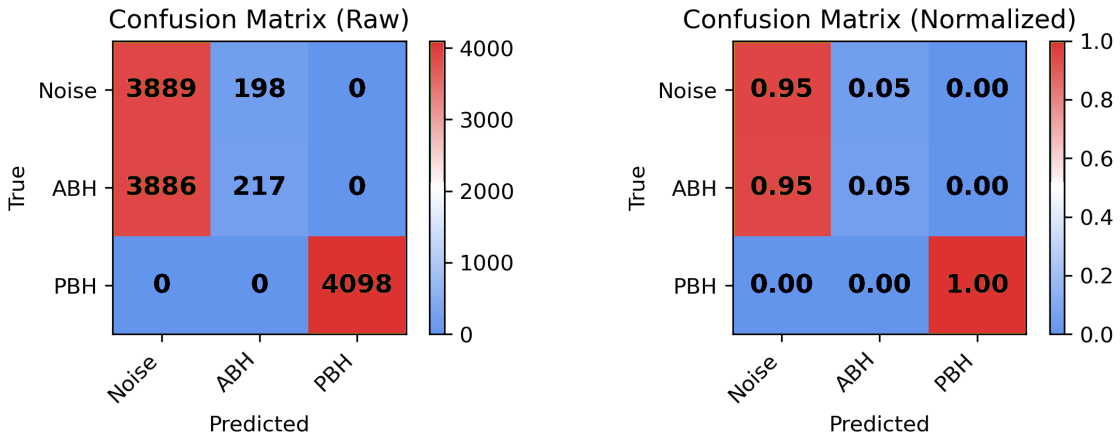


Figure 5.16: Confusion matrices of the autoencoder trained on 10 datasets with SNR values 0.001-0.01.

6 Conclusion

In this thesis, large-scale datasets were simulated for the planned next-generation gravitational-wave detector **Einstein Telescope** in order to train deep-learning models for the classification of events. The Einstein Telescope is designed to have excellent low-frequency sensitivity (a few Hz), which allows for the early detection of gravitational-wave events. The datasets were split into three classes: noise events, **astrophysical black hole** (ABH) binaries and **primordial black hole** (PBH) binaries. The main difference between the ABHs and PBHs is their mass ranges. While the mass range of the ABHs ranges from (5-80) $M_\odot$, the mass range of the PBHs only ranges from (0.2-1) $M_\odot$. These low masses lead to extremely high frequency content in the late-inspiral and merger phase (up to several kHz). As a result, a high sampling rate of 32768 Hz was required to accurately simulate these events. These waveforms were then injected into noise events to simulate the strain of gravitational-wave detectors. The models used were a **residual network** (ResNet) and an **autoencoder**. The primary task of these models was to classify the signals based on the detector strain as the input. The output should then be a class label associated with either a noise event (0), an ABH binary (1) or a PBH binary (2). At first, the classification worked seemingly well for both the ResNet and the autoencoder. Surprisingly, the ResNet reached an overall accuracy of 99%, whereas the autoencoder reached 95.8% for the classification. Both the ResNet and the autoencoder had a 100% accuracy in common for the PBH class, which is suspiciously high. Being convinced that there was no data leakage during training, the next step was to test the results for different scenarios, such as using test datasets separate from the training sets and lowering the signal-to-noise ratio (SNR) until the signal should be undetectable. The accuracy of the PBH events was still 100% in all cases, even at SNR values of 0.001, where the accuracy of the ABH events was then at 0%. The models were trained on new datasets with updated data generation, and they both still had 100% accuracy for the PBH events. The overall accuracy decreased, and the ResNet even misclassified some PBH events (between 6-20 events out of roughly 50000 events per class). The biggest change could be seen in the accuracy of the autoencoder, as it now misclassified around 20% of the noise events as ABH and vice versa, but the PBH accuracy remained at 100%. The model was then trained once more for low SNR datasets, where again the PBH accuracy was still 100%. There could be many different possibilities for this, such as something that happens during data simulation or a simpler explanation, such as the high frequency content and the bandpass clipping the events in the high frequency range. There is still a lot of work that can be done to investigate and optimise the models. Future steps include clarifying the best and most accurate way to simulate the data, defining the optimal frequency range for the bandpass filter, and fine-tuning the parameters of the autoencoder. The models trained for this thesis serve as a proof of concept, and the final verdict must be that there is no clear explanation for the suspiciously high accuracy in the PBH detection. Further work can be done to investigate the data simulation and examine what the models are paying attention to during training.

Bibliography

[1] The LIGO Scientific Collaboration and B. Abbott. "LIGO: The Laser Interferometer Gravitational-Wave Observatory" (2007). DOI: 10.48550/ARXIV.0711.3041. URL: https://arxiv.org/abs/0711.3041.

[2] F. Acernese et al. "Advanced Virgo: a 2nd generation interferometric gravitational wave detector" (2014). DOI: 10.48550/ARXIV.1408.3978. URL: https://arxiv.org/abs/1408.3978.

[3] Adrian Abac et al. *The Science of the Einstein Telescope*. 2025. DOI: 10.48550/ARXIV.2503.12263. URL: https://arxiv.org/abs/2503.12263.

[4] Albert Escrivà, Florian Kuhnel, and Yuichiro Tada. "Primordial Black Holes" (2022). DOI: 10.48550/ARXIV.2211.05767. URL: https://arxiv.org/abs/2211.05767.

[5] S. Hawking. "Gravitationally Collapsed Objects of Very Low Mass". *Monthly Notices of the Royal Astronomical Society* 152.1 (Apr. 1971), pp. 75–78. ISSN: 1365-2966. DOI: 10.1093/mnras/152.1.75. URL: http://dx.doi.org/10.1093/mnras/152.1.75.

[6] B. J. Carr and S. W. Hawking. "Black Holes in the Early Universe". *Monthly Notices of the Royal Astronomical Society* 168.2 (Aug. 1974), pp. 399–415. ISSN: 1365-2966. DOI: 10.1093/mnras/168.2.399. URL: http://dx.doi.org/10.1093/mnras/168.2.399.

[7] James B. Hartle. *Gravity: An Introduction to Einstein's General Relativity*. Cambridge University Press, June 2021. ISBN: 9781316517543. DOI: 10.1017/9781009042604. URL: http://dx.doi.org/10.1017/9781009042604.

[8] Stephen R. Taylor. *The Nanohertz Gravitational Wave Astronomer*. 2021. DOI: 10.48550/ARXIV.2105.13270. URL: https://arxiv.org/abs/2105.13270.

[9] *The Einstein Equation*. URL: https://www.damtp.cam.ac.uk/user/tong/gr/four.pdf (visited on 03/11/2026).

[10] Yadong Xue and Zhoujian Cao. *The transverse-traceless gauge and the gauge problem of second order gravitational waves*. 2025. DOI: 10.48550/ARXIV.2509.26159. URL: https://arxiv.org/abs/2509.26159.

[11] Nigel T. Bishop and Luciano Rezzolla. "Extraction of gravitational waves in numerical relativity". *Living Reviews in Relativity* 19.1 (Oct. 2016). ISSN: 1433-8351. DOI: 10.1007/s41114-016-0001-9. URL: http://dx.doi.org/10.1007/s41114-016-0001-9.

[12] B. P. Abbott et al. "Observation of Gravitational Waves from a Binary Black Hole Merger". *Physical Review Letters* 116.6 (Feb. 2016). ISSN: 1079-7114. DOI: 10.1103/physrevlett.116.061102. URL: http://dx.doi.org/10.1103/PhysRevLett.116.061102.

[13] Marica Branchesi et al. "Science with the Einstein Telescope: a comparison of different designs". *Journal of Cosmology and Astroparticle Physics* 2023.07 (July 2023), p. 068. ISSN: 1475-7516. DOI: 10.1088/1475-7516/2023/07/068. URL: http://dx.doi.org/10.1088/1475-7516/2023/07/068.

[14] G. Franciolini et al. "Searching for Primordial Black Holes with the Einstein Telescope: impact of design and systematics" (2023). DOI: 10.48550/ARXIV.2304.03160. URL: https://arxiv.org/abs/2304.03160.

[15] Xhesika Koroveshi et al. "Cryogenic payloads for the Einstein Telescope: Baseline design with heat extraction, suspension thermal noise modeling, and sensitivity analyses". *Physical Review D* 108.12 (Dec. 2023). ISSN: 2470-0029. DOI: 10.1103/physrevd.108.123009. URL: http://dx.doi.org/10.1103/PhysRevD.108.123009.

[16] Simon J.D. Prince. *Understanding Deep Learning*. The MIT Press, 2023. URL: http://udlbook.com.

[17] F. Rosenblatt. "The perceptron: A probabilistic model for information storage and organization in the brain." *Psychological Review* 65.6 (1958), pp. 386–408. ISSN: 0033-295X. DOI: 10.1037/h0042519. URL: http://dx.doi.org/10.1037/h0042519.

[18] *What is Normalization in ML*. URL: https://www.datacamp.com/tutorial/normalization-in-machine-learning (visited on 03/11/2026).

[19] Sebastian Raschka, Yuxi (Hayden) Liu, and Vahid Mirjalili. *Machine Learning with PyTorch and Scikit-Learn*. Birmingham, UK: Packt Publishing, 2022. ISBN: 978-1801819312.

[20] Kaiming He et al. *Deep Residual Learning for Image Recognition*. 2015. DOI: 10.48550/ARXIV.1512.03385. URL: https://arxiv.org/abs/1512.03385.

[21] Keiron O'Shea and Ryan Nash. *An Introduction to Convolutional Neural Networks*. 2015. DOI: 10.48550/ARXIV.1511.08458. URL: https://arxiv.org/abs/1511.08458.

[22] Serkan Kiranyaz et al. "1D convolutional neural networks and applications: A survey". *Mechanical Systems and Signal Processing* 151 (Apr. 2021), p. 107398. ISSN: 0888-3270. DOI: 10.1016/j.ymssp.2020.107398. URL: http://dx.doi.org/10.1016/j.ymssp.2020.107398.

[23] Umberto Michelucci. *An Introduction to Autoencoders*. 2022. DOI: 10.48550/ARXIV.2201.03898. URL: https://arxiv.org/abs/2201.03898.

[24] *ET steering committee, design update 2020*. URL: https://apps.et-gw.eu/tds/ql/?c=15418 (visited on 03/11/2026).

[25] *PyCBC Documentation*. URL: https://pycbc.org/pycbc/latest/html/ (visited on 03/11/2026).

[26] Michele Maggiore. *Gravitational Waves: Volume 1: Theory and Experiments*. Oxford University PressOxford, Oct. 2007. ISBN: 9780191717666. DOI: 10.1093/acprof:oso/9780198570745.001.0001. URL: http://dx.doi.org/10.1093/acprof:oso/9780198570745.001.0001.

[27] Sanjeev Dhurandhar and Sanjit Mitra. *General Relativity and Gravitational Waves: Essentials of Theory and Practice*. Springer International Publishing, 2022. ISBN: 9783030923358. DOI: 10.1007/978-3-030-92335-8. URL: http://dx.doi.org/10.1007/978-3-030-92335-8.

[28] B P Abbott et al. "A guide to LIGO–Virgo detector noise and extraction of transient gravitational-wave signals". *Classical and Quantum Gravity* 37.5 (Feb. 2020), p. 055002. ISSN: 1361-6382. DOI: 10.1088/1361-6382/ab685e. URL: http://dx.doi.org/10.1088/1361-6382/ab685e.

[29] *PyTorch documentation*. URL: https://docs.pytorch.org/docs/stable/index.html (visited on 03/11/2026).

[30] *PyTorch lightning documentation*. URL: https://lightning.ai/docs/pytorch/stable/ (visited on 03/11/2026).

[31] *NVIDIA A100*. URL: https://www.nvidia.com/en-us/data-center/a100/ (visited on 03/11/2026).

[32] Laurens Van der Maaten and Geoffrey Hinton. "Visualizing data using t-SNE". *Journal of Machine Learning Research* 9.11 (2008), pp. 2579–2605.

[33] *Scikit-learn documentation TSNE*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html (visited on 03/11/2026).

Declaration of Academic Integrity

I hereby confirm that this thesis, entitled "Identifying Sub-Solar Primordial Black Holes for Einstein Telescope Using Deep Learning Methods", is solely my own work and that I have used no sources or aids other than the ones stated. All passages in my thesis for which other sources, including electronic media and AI Tools, have been used, be it direct quotes or content references, have been acknowledged as such and the sources cited. I am aware that plagiarism is considered an act of deception which can result in sanction in accordance with the examination regulations.

Münster, March 11, 2026 

I consent to having my thesis cross-checked with other texts to identify possible similarities and to having it stored in a database for this purpose.

I confirm that I have not submitted the following thesis in part or whole as an examination paper before.

Münster, March 11, 2026 