Cosmological Parameter Inference using Gravitational Waves and Machine Learning

Federico Stachurski

SUBMITTED IN FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

SCHOOL OF PHYSICS AND ASTRONOMY

COLLEGE OF SCIENCE AND ENGINEERING

09 2024

To my family,
Angela Lina Carnemolla, Dale Stachurski,
Riccardo Stachurski, Marinella Carnemolla,
Richard Stachurski, Mary Lou Stachurski, Traci Klabo,
Billy Ray Klabo, Giovanni Carnemolla, and
Giuseppina Pluchino.

Abstract

In 1929, Edwin Hubble's discovery of the relationship between galaxy distances and their recession velocities unveiled the universe's expansion, laying the foundation for modern cosmology and the task to measure the Hubble constant, H_0. In 1986, Bernard F. Schutz proposed using gravitational waves from compact binary mergers, such as neutron stars and black holes, as a novel method to estimate H_0. This innovative approach marked the beginning of a new era in cosmological research, significantly advanced by the advent of gravitational wave detection through the Laser Interferometer Gravitational Wave Observatory (LIGO). The field has since evolved, employing advanced Bayesian techniques and extensive galaxy catalogues to improve the precision of H_0 measurements. However, as the sensitivity of detectors increases and the rate of gravitational wave observations grows, computational challenges—particularly in hierarchical Bayesian analysis—pose significant hurdles due to the intensive and time-consuming nature of traditional methods. In response to these challenges, this thesis, under the supervision of Dr. Christopher Messenger and Prof. Martin Hendry, explores the integration of machine learning into cosmological research, specifically focusing on a novel approach called `CosmoFlow` to extract cosmological information from gravitational waves. `CosmoFlow` uses Normalising Flows, machine learning models capable of efficiently estimating probability distribution functions of complex datasets, providing a faster and and potentially advantageous approach to hierarchical Bayesian inference of the Hubble constant. Our work demonstrates how `CosmoFlow` can significantly accelerate the process compared to existing methodologies. Throughout this thesis, we rigorously compare the results of `CosmoFlow` with those obtained using `gwcosmo`, a well-established tool in gravitational wave cosmology. By contrasting `CosmoFlow` with `gwcosmo` results, we highlight the strengths and limitations of each method, emphasising the potential of machine learning to address existing computational bottlenecks in cosmological analyses. This comparative study aims to contribute to the ongoing efforts to resolve the current 4.4σ tension between different H_0 measurement techniques, paving the way for more efficient and accurate future analyses in this rapidly evolving field.

Contents

List of Figures

List of Tables

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisors, Christopher Messenger and Martin Hendry, for their guidance and support throughout my PhD. I truly could not have completed this work without their insight and encouragement. A special thanks also goes to the whiteboard in Chris' office, which played a pivotal role in helping me map out complex mathematical formulas and analytical concepts.

I extend my heartfelt thanks to my family, who supported me through this entire journey, my parents, grandparents, and brother, for their constant encouragement and belief in me. I am also grateful to the Cosmology subgroup within the LVK collaboration for allowing me to contribute to their successful projects and for generously sharing ideas that shaped my analysis. Their collaboration was invaluable to my work. A sincere thank you to the University of Glasgow and the College of Science and Engineering for their support and for funding my PhD. Last but not least, I want to give a big shout-out to my 253 office mates: Ross Johnston, Jessica Irwin, Victoria Graham, Storm Colloms, Thejas Seetharamu, Leigh Smith, Eungwang Harry Seo and Naren Nagarajan. Your camaraderie, laughter, and support through tough times made this journey much easier and more enjoyable.

I would like to thank my close childhood friends: Federico Tumino, Gabriele Mauro, Giovanni Leonardi, Simone Licitra, Alessandro Carnemolla, and many others. Your friendship, support, and appreciation for who I am have always meant a lot to me. I am grateful for the good times, the laughter, and the encouragement you've given me over the years. Thank you for always being there.

Thank you to everyone who contributed to this incredible journey. Your help and support have been deeply appreciated.

Declaration

I declare that, except where explicit reference is made to the contribution of others, that this dissertation is the result of my own work and has not been submitted for any other degree at the University of Glasgow or any other institution

Federico Stachurski

Abbreviations

- `EM` - electromagnetic
- `GW` - gravitational waves
- `LIGO` - Laser Interferometer Gravitational Wave Observatory
- `SNR` - signal to noise ratio
- `FAR` - false alarm ratio
- `ML` - machine learning
- `MLP` - multi layer perceptron
- `NN` - neural network
- `PDF` - probability distribution function
- `CDF` - cumulative distribution function
- `KL` - Kullback-Leibler
- `SGD` - stochastic gradient descent
- `Adam` - adaptive moment estimation
- `VAE` - variational autoencoders
- `GWTC` - gravitational wave transient catalog
- `EOS` - equation of state
- `LVK` - LIGO-Virgo-KAGRA
- `CBC` - compact binary coalescence
- `NF` - normalising flow
- `BBH` - binary black hole
- `BNS` - binary neutron star
- `NSBH` - neutron star black hole
- `PE` - parameter estimation
- `PSD` - power spectral density
- `GPU` - graphics processing unit
- `PLP` - power law plus peak
- `KDE` - kernel density estimation
- `JS` - Jensen–Shannon
- `CMB` - Cosmic Microwave Background
- `ASD` - amplitude spectral density

- `MCMC` - Monte Carlo Markov Chain
- `LOS` - line-of-sight
- `MSE` - mean squared error
- `GAN` - Generative Adversarial Networks
- `ABC` - Approximate Bayesian Computation
- `CUDA` - Compute Unified Device Architecture
- `ET` - Einstein Telescope
- `MDC` - Mock Data Challenge

Chapter 1

Introduction

1.1 The Universe is Expanding

As astronomers began to examine the light spectra of galaxies, they observed that the absorption lines of specific elements were displaced from expected positions. This displacement results from the Doppler effect, a phenomenon where the frequency of light or sound emitted by a source changes in relation to an observer's relative velocity. When the source approaches the observer, the wavefronts of the emission compress, leading to a *blueshift* in electromagnetic (EM) observations. Conversely, as the source recedes, the wavefronts extend, causing a *redshift*. The magnitude of this shift for an absorption line is quantified as:

$$z = \frac{\lambda_{\text{Obs}} - \lambda_{\text{True}}}{\lambda_{\text{True}}}. \tag{1.1}$$

Here, λ_{True} is the true rest frame wavelength and λ_{Obs} is the observed wavelength. A negative value of z indicates a blueshift (source approaching), while a positive z signifies a redshift (source receding). Given that the emission is EM, the velocity of the source is related to the redshift by the relativistic Doppler equation:

$$z = \sqrt{\frac{1+\frac{v}{c}}{1-\frac{v}{c}}} - 1, \tag{1.2}$$

where v represents the velocity of the emitter, and c is the speed of light, approximately $3 \times 10^8\,\mathrm{m\,s^{-1}}$. This formula accounts for relativistic effects [1].

1.1. The Universe is Expanding

In the early 20th century, significant advancements were made in understanding the scale of the universe. Between 1908 and 1912, Henrietta Leavitt, an American astronomer, made a groundbreaking discovery involving a class of stars known as Cepheid Variables. Leavitt observed that these stars exhibited a consistent relationship between their brightness and the period of their variation in brightness. Specifically, she found that the longer the period of a Cepheid star's pulsation, the brighter it was. This discovery provided astronomers with a powerful tool: by measuring the period of a Cepheid Variable, one could determine its intrinsic luminosity. Once the luminosity was known, the distance to the star could be calculated by comparing its intrinsic luminosity to its observed brightness, making Cepheid Variables "standard candles" for measuring astronomical distances [2].

This method of distance measurement laid the groundwork for significant astronomical discoveries. In 1929, American astronomer Edwin Hubble expanded on these ideas by studying the relationship between the distances to galaxies and their redshifts [2]. Hubble's research revealed a linear relationship between the distance to a galaxy and its recessional velocity, which is related to the redshift through the relativistic Doppler formula, described in Eq.(1.1). Essentially, the farther away a galaxy is, the faster it is moving away from us. This observation led to the formulation of what is now known as Hubble's Law, which not only demonstrated the expansion of the universe but also provided a means to quantify the expansion rate. At low redshifts, where $z \ll 1$, the redshift-distance relation can be approximated by the linear form of Hubble's Law:

$$H_0 \approx \frac{cz}{d}, \tag{1.3}$$

where d denotes the distance to the galaxy, z is the redshift, and c is the speed of light. This simple yet profound relationship has become a fundamental framework in cosmology, allowing astronomers to estimate the expansion rate of the universe, known as the Hubble constant (H_0). Although initially underestimated, Hubble's discovery provided the first quantitative understanding of the universe's expansion and paved the way for modern cosmological studies.

In the context of a Friedmann-Lemaître-Robertson-Walker (FLRW) universe, which is the foundation of modern cosmology, the expansion of the universe is more comprehensively described by the Hubble-Lemaître parameter as a function of redshift z [1]:

$$H(z) = H_0\sqrt{\Omega_m(1+z)^3 + \Omega_k(1+z)^2 + \Omega_\Lambda}, \tag{1.4}$$

Here, H_0 is the Hubble constant, representing the rate of expansion in the current epoch. The parameters Ω_m, Ω_Λ, and Ω_k correspond to the dimensionless densities of different components of the universe: Ω_m includes both baryonic matter and dark matter, Ω_Λ represents dark energy density, often attributed to a cosmological constant responsible for the universe's accelerated expansion, and Ω_k corresponds to the contribution from the spatial curvature of the universe. Observations suggest that the universe is flat, implying $\Omega_k = 0$ [3]. Consequently, the relationship between these components follows the constraint $\Omega_m + \Omega_\Lambda = 1$, indicating that the combined contributions of matter and dark energy equal the critical density, consistent with a flat universe geometry.

The expansion history of the universe translates into a "redshift-distance relation ", associating the redshift z of observable sources with their luminosity distance $D_{\rm L}(z)$. For a flat universe, this relation is given by:

$$D_{\rm L}(z) = \frac{c(1+z)}{H_0} \int_0^z \frac{dz'}{H(z')} \tag{1.5}$$

where $D_{\rm L}(z)$ is the luminosity distance, c is the speed of light, and $H(z)$ is the Hubble parameter at a given redshift. This relationship allows astronomers to measure cosmological parameters by comparing the observed redshift with the luminosity distance of sources such as supernovae, variable stars, or GW events. By combining observations of z and $D_{\rm L}(z)$, one can estimate the cosmological parameters governing the universe's expansion. If the parameters Ω_m, Ω_Λ, and Ω_k are known from independent observations (e.g., cosmic microwave background studies [4, 5]), this redshift-distance relation can be used to measure H_0 more precisely.

1.2 The Hubble Tension

Since Edwin Hubble's discovery of the universe's expansion, scientists have focused on precisely measuring the expansion rate, known as the Hubble constant (H_0), which is very important for understanding the fate of our universe. Determining H_0 helps to predict the ultimate destiny of the universe, whether it will continue expanding forever, eventually slow down, or collapse in a "big crunch" .Currently, there exists a significant tension in the measured values of H_0, approximately 4.4σ. The Planck experiment, which analyses the Cosmic Microwave Background (CMB) from the early universe, estimates the Hubble constant to be $H_0 = 67.4 \pm 0.5$ km s^{-1}Mpc^{-1} (1σ confidence interval) [4]. In contrast, the

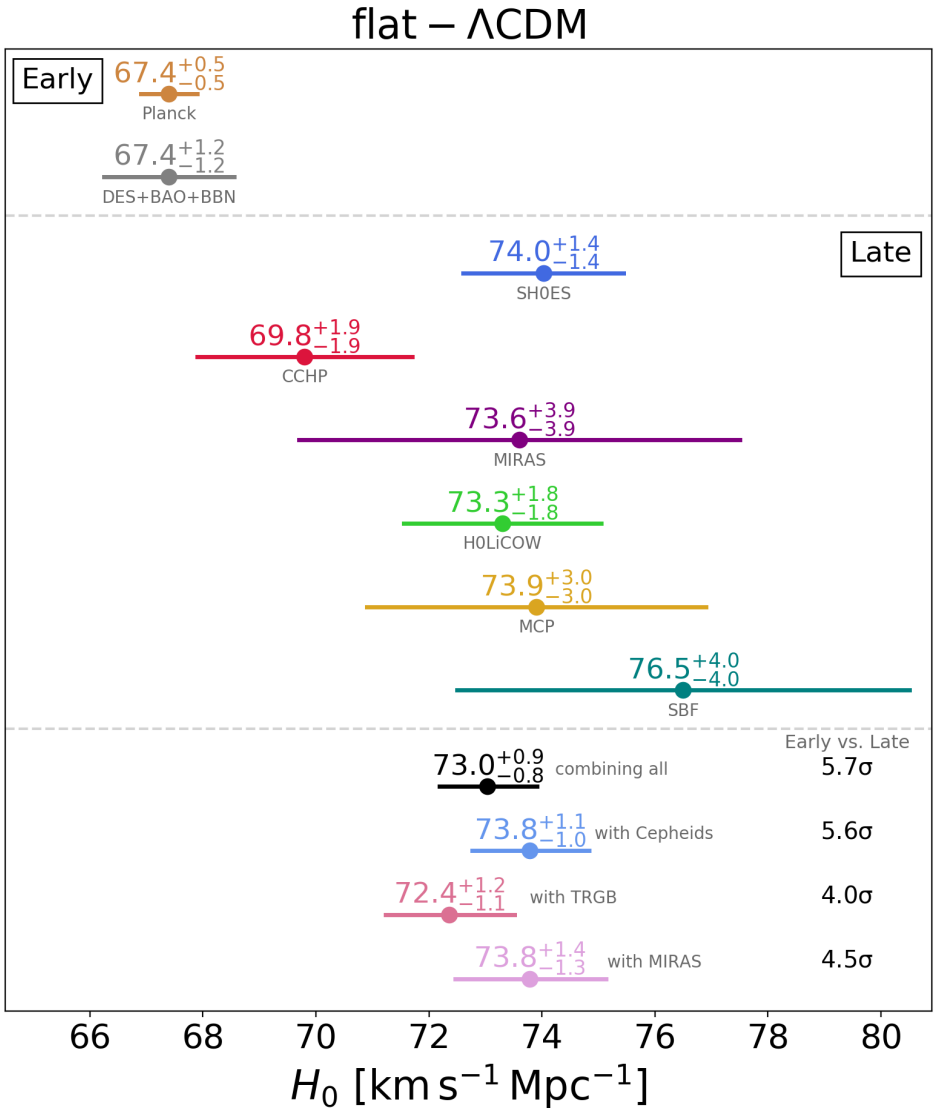

Figure 1.1: Measured values of the Hubble constant H_0 from early and late universe observations, illustrating the varying results from different experiments and analyses and highlighting the current Hubble tension. Image credit [8].

SH0ES collaboration, which measures distances to Type Ia supernovae in the late universe, reports $H_0 = 74.03 \pm 1.42$ km s^{-1}Mpc^{-1} (1σ confidence interval) [6]. This discrepancy, illustrated in Figure 1.1, suggests a potential inconsistency between early and late universe measurements or may point to unknown systematic errors or even new physics [7].

The difference between early and late universe measurements is a fundamental aspect of the Hubble tension. Early universe measurements, like those from the Planck satellite, rely on the CMB, which provides a snapshot of the universe when it was only 380,000 years old [2]. These measurements depend on our understanding of the physics of the early universe, including the composition of matter and energy. In contrast, late universe measurements, such as those from the SH0ES project, involve direct observations

of supernovae in relatively nearby galaxies. These measurements are influenced by the local universe's expansion rate and require precise calibration of the distance ladder, a method that connects different astronomical distance indicators. This tension between early and late universe measurements raises questions about our current understanding of cosmology. If the difference is not due to systematic errors, it could indicate new physics beyond the standard cosmological model, such as a variation in dark energy over time or the presence of an unknown component affecting the expansion rate [7]. Other methods have also been used to measure H_0. For instance, the Dark Energy Survey Year 1 (DES Y1) [9], which combines clustering and weak lensing data with Baryon Acoustic Oscillations (BAO) and Big Bang Nucleosynthesis (BBN), has constrained the Hubble constant to $H_0 = 67.2 \pm 1.1\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$ [10], aligning with early-universe measurements. Similarly, observations of Mira Variables in the Type Ia Supernova Host NGC 1559 yield a late-universe value of $H_0 = 73.6 \pm 4.0\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$ [11], consistent with the SH0ES result. Given the ongoing tension in the measurement of the Hubble constant, it is of key importance to explore independent methods for resolving this discrepancy.

Gravitational wave (GW) events provide a promising and independent approach to measuring H_0 by using them as *standard sirens* [12]. GWs are ripples in spacetime caused by the acceleration of massive objects, such as the mergers of black holes or neutron stars. The concept of *standard sirens* was first proposed by Schutz in 1986 [12]. Similar to how standard candles in astronomy use the known intrinsic luminosity of objects to determine distance, standard sirens rely on the amplitude of GW signals. This amplitude directly correlates with the energy released during events such as the mergers of neutron stars or black holes, allowing for an independent estimation of distances and, consequently, the Hubble constant. Standard sirens can be categorized into two types: bright sirens and dark sirens. Bright sirens are GW events with an observed electromagnetic counterpart, allowing a direct identification of the host galaxy. Dark sirens, on the other hand, are events without an identified electromagnetic counterpart, where galaxy catalogues are used statistically to infer the host galaxy [13].

By combining redshift data from EM galaxy catalogues with distance measurements derived from GW events, one can estimate the Hubble constant using the relation given in Eq. (1.3). Recently, the LIGO-Virgo-KAGRA (LVK) collaboration has made significant progress in inferring the Hubble constant through two distinct methodologies: `gwcosmo` [14] and `IcaroGW` [15]. Their primary publication [16] reports an estimated value of $H_0 = 68^{+12}_{-6}\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$ (68% credible interval) when combining both dark siren and bright siren events.

In the next section, we will look at the mathematical framework of GW events, from how they are generated in astrophysical events to how they are detected by observatories like LIGO and Virgo. We will also see how these mathematical models help us extract important cosmological parameters, including H_0, from the observed GW signals. This exploration will lay the groundwork for understanding how these events can play an important role in modern cosmology and help address the ongoing challenges in the field.

1.3 Gravitational Waves

1.3.1 General Relativity and Gravitational Waves

In 1916, Albert Einstein published his groundbreaking work titled *The Foundations of the General Theory of Relativity* [17], introducing a new framework for understanding gravity that went beyond the limitations of Newtonian physics. Einstein's theory showed that massive objects cause spacetime to curve, and that energetic events can produce ripples, known as gravitational waves (GWs), which propagate through the fabric of spacetime. To study GWs, we begin with a fundamental concept from relativity: Minkowski space [18]. This is a four-dimensional spacetime that combines three spatial dimensions and one time dimension. It is the simplest model of spacetime, representing a flat, empty universe where no gravitational effects are present. In General Relativity, however, spacetime is typically curved by the presence of mass and energy. When studying GWs, we often begin by approximating spacetime as nearly flat, similar to Minkowski space, and treat any deviations from flatness as small perturbations. This approach allows us to analyze GWs as minor disturbances on a flat background spacetime. In this approximation, the metric tensor $g_{\mu\nu}$ is expressed as a perturbation $h_{\mu\nu}$ added to the flat Minkowski metric $\eta_{\mu\nu}$:

$$g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu} \tag{1.6}$$

where $\eta_{\mu\nu} = \text{diag}(-1,1,1,1)$ and $|h_{\mu\nu}| \ll 1$ is a small perturbation, for all values of μ and ν [18]. Then in a new coordinate system, the metric tensor takes the form of

$$g'_{\mu\nu} = \frac{\partial x^\mu}{\partial x'_\mu}\frac{\partial x^\nu}{\partial x'_\nu}\eta_{\mu\nu} + \frac{\partial x^\mu}{\partial x'_\mu}\frac{\partial x^\nu}{\partial x'_\nu}h_{\mu\nu} \tag{1.7}$$

where the prime notation indicates the new coordinate system. In the assumption of a nearly Lorentzian transformation, Eq.(1.7) holds true for any observer [18]. We can then make a small change in the new coordinate system, as $x'^{\mu} = x^{\mu} + \xi^{\mu}(x^{\nu})$, where ξ^{μ} are functions whose components depend on x^{ν}, with the approximation of the rate of change being very small (i.e. $\xi^{\mu}_{,\nu} \ll 1$). Then, knowing that $\frac{\partial x'_{\mu}}{\partial x^{\nu}} = \delta^{\mu}_{\nu} + \xi^{\mu}_{,\nu}$,

$$\frac{\partial x^{\mu}}{\partial x'^{\gamma}} = \delta^{\mu}_{\gamma} - \frac{\partial x^{\nu}}{\partial x'^{\gamma}} \frac{\partial \xi^{\mu}}{\partial x^{\nu}} \simeq \delta^{\mu}_{\gamma} - \xi^{\mu}_{,\gamma}, \tag{1.8}$$

where higher-order terms were neglected. In Eq. (1.8), we have used the comma notation indicating partial differentiation. Specifically, $\xi^{\mu}_{,\gamma}$ represents the partial derivative of ξ^{μ} with respect to the coordinate x^{γ}, i.e., $\xi^{\mu}_{,\gamma} = \frac{\partial \xi^{\mu}}{\partial x^{\gamma}}$. This notation is commonly used in general relativity to simplify the representation of derivatives. The term δ^{μ}_{γ} is the Kronecker delta, which acts as an identity matrix in this context, ensuring that when $\mu = \gamma$, the value is 1, and otherwise it is 0.

Then by substituting Eq.(1.8) into Eq.(1.7), and simplifying,

$$g'_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu} - \xi_{\mu,\nu} - \xi_{\nu,\mu}, \tag{1.9}$$

where we have defined $\xi_{\gamma} = \eta_{\gamma\beta}\xi^{\beta}$, and the partial derivatives of $\eta_{\mu\nu}$ are zero. From this we see that in the new coordinate system, the perturbation term has the form $h'_{\mu\nu} = h_{\mu\nu} - \xi_{\mu,\nu} - \xi_{\nu,\mu}$ and still retains the assumption of a nearly flat spacetime. We can then choose the ξ functions to simplify the problem as much as possible. This is called choosing a *gauge* for the problem. Then, computing the fully covariant Riemann Christoffel tensor [18], and considering the new metric tensor defined in Eq.(1.9), follows:

$$R_{\mu\nu\gamma\delta} = \frac{1}{2}\big(h_{\mu\delta,\nu\gamma} + h_{\nu\gamma,\mu\delta} - h_{\mu\gamma,\nu\delta} - h_{\nu\delta,\mu\gamma}\big). \tag{1.10}$$

By further contracting Eq.(1.10), thus computing the Ricci tensor, and inserting this into the definition of the Einstein tensor, defined as $G_{\mu\nu} = R_{\mu\nu} - \frac{1}{2}g_{\mu\nu}R$, where R is the Ricci scalar (obtained by further contracting the Ricci tensor), follows:

$$G_{\mu\nu} = -\frac{1}{2}\big(\overline{h}^{,\delta}_{\mu\nu,\delta} + \eta_{\mu\nu}\overline{h}^{,\delta\gamma}_{\delta\gamma} - \overline{h}^{,\delta}_{\mu\delta,\nu} - \overline{h}^{,\delta}_{\nu\delta,\mu}\big), \tag{1.11}$$

where for simplicity we have defined the term $\overline{h}_{\mu\nu} \equiv h_{\mu\nu} - \frac{1}{2}\eta_{\mu\nu}h$. After computing the value $G_{\mu\nu}$ in Eq.(1.11), and knowing that $G_{\mu\nu} = 8\pi T_{\mu\nu}$ [18] it follows that

$$\overline{h}^{,\delta}_{\mu\nu,\delta} + \eta_{\mu\nu}\overline{h}^{,\delta\gamma}_{\delta\gamma} - \overline{h}^{,\delta}_{\mu\delta,\nu} - \overline{h}^{,\delta}_{\nu\delta,\mu} = -16\pi T_{\mu\nu}, \tag{1.12}$$

where $T_{\mu\nu}$ is the energy density tensor. Finally, by applying a Lorentz gauge (denoted by LG) transformation, which just helps to simplify the problem in Eq.(1.12), of the form $\overline{h}^{(\mathrm{LG})}_{\mu\nu} = \overline{h}^{\mathrm{old}}_{\mu\nu} - \xi_{\mu,\nu} - \xi_{\nu,\mu}$ with solution of the form $\overline{h}^{\mu\delta}_{,\delta} = 0$, in free space (i.e. $T_{\mu\nu} = 0$),

$$\overline{h}^{,\delta}_{\mu\nu,\delta} = \eta^{\delta\delta}\partial_\delta\partial_\delta\overline{h}_{\mu\nu} = \Box\overline{h}_{\mu\nu} = 0, \tag{1.13}$$

where $\Box = \eta^{\delta\delta}\partial_\delta\partial_\delta$ is the *d'Alembertian* operator. Given the metric properties of $\eta^{\mu\nu}$ in flat spacetime and using natural units ($c = 1$), this becomes:

$$\left(\frac{\partial^2}{\partial t^2} - \nabla^2\right)\overline{h}_{\mu\nu} = 0, \tag{1.14}$$

where $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ is the Laplacian operator. The function shown in Eq.(1.14) describes a wave, indicating that perturbations in spacetime propagate through space as waves, or more specifically, as GWs, traveling at the speed of light. This is evident from the velocity term in the wave equation, $\eta^{00} = -\frac{1}{c^2}$. From this point forward, we will refer to the term h as *strain*, which quantifies the relative change in length experienced by free particles as a GW passes through them [18].

The GWs, like EM radiation, are polarised. They exhibit two fundamental polarisation modes: the "plus "(h_+) and the "cross "($h_\times$) polarisations. These modes describe the distinct ways in which space is stretched and compressed as the wave propagates and can be expressed as a linear combination of the two as:

$$h = h_+\mathbf{e}_+ + h_\times\mathbf{e}_\times, \tag{1.15}$$

where $\mathbf{e}_+$ and $\mathbf{e}_\times$ are the polarization tensors. Mathematically, the "plus" polarization tensor, $\mathbf{e}_+$, and the "cross" polarization tensor, $\mathbf{e}_\times$, are defined as follows in the plane perpendicular to the direction of wave propagation (typically along the z-axis):

$$\mathbf{e}_+ = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad \text{and} \quad \mathbf{e}_\times = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

where $\mathbf{e}_+$ describes the stretching along the x-axis and compression along the y-axis, and $\mathbf{e}_\times$ describes the stretching and compression along axes that are rotated by 45 degrees relative to the x and y axes. To explicitly demonstrate that $\mathbf{e}_\times$ is rotated by 45° relative to $\mathbf{e}_+$, we apply a rotation matrix to $\mathbf{e}_+$. The 45° rotation matrix is given by:

$$R(45^\circ) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix},$$

and its transpose is:

$$R(45^\circ)^T = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}.$$

To rotate $\mathbf{e}_+$ by 45°, we compute:

$$e'_+ = R(45^\circ)\,\mathbf{e}_+ R(45^\circ)^T,$$

which results in:

$$e'_+ = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \mathbf{e}_\times.$$

The resulting matrix is $\mathbf{e}_\times$, which shows that applying a 45° rotation to the plus polarization matrix $\mathbf{e}_+$ results in the cross polarisation matrix $\mathbf{e}_\times$.

To simplify the description of the polarised GW, we apply the Transverse-Traceless (TT) gauge [13]. In this gauge, a GW propagating in the z direction has a particularly simple form, where only the physical degrees of freedom, those associated with the "plus" and "cross" polarisations, are retained. The metric perturbation in the TT gauge is given by

$$h_{\mu\nu}(t,x,y,z) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & h_+ & h_\times & 0 \\ 0 & h_\times & -h_+ & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \cos(\omega t - kz). \tag{1.16}$$

This representation shows how the h_+ and $h_\times$ polarisation modes manifest as distortions in the xy-plane, while the wave propagates along the z-axis. The term $\cos(\omega t - kz)$, where ω is the angular frequency and k is the wave number in the z-direction, characterizes the wave's oscillatory behavior in both time (t) and space (z). In the framework of the Transverse-Traceless (TT) gauge, the GW affects only the x and y directions, leaving the z-axis unaffected. The traceless nature of this gauge ensures that the perturbation induces a shearing motion without changing the overall volume of the affected region. This results in pure deformation within the xy-plane, as depicted in Fig. 1.2, which shows the oscillations of the h_+ and $h_\times$ polarization modes over a single cycle T.

The GW events are produced when massive objects undergo acceleration, but not just any kind of acceleration results in detectable GW. For GWs to be emitted, the system must have an asymmetry in its mass distribution. This is where the *quadrupole* moment comes into play. The *quadrupole* moment is a measure of how mass is distributed across an

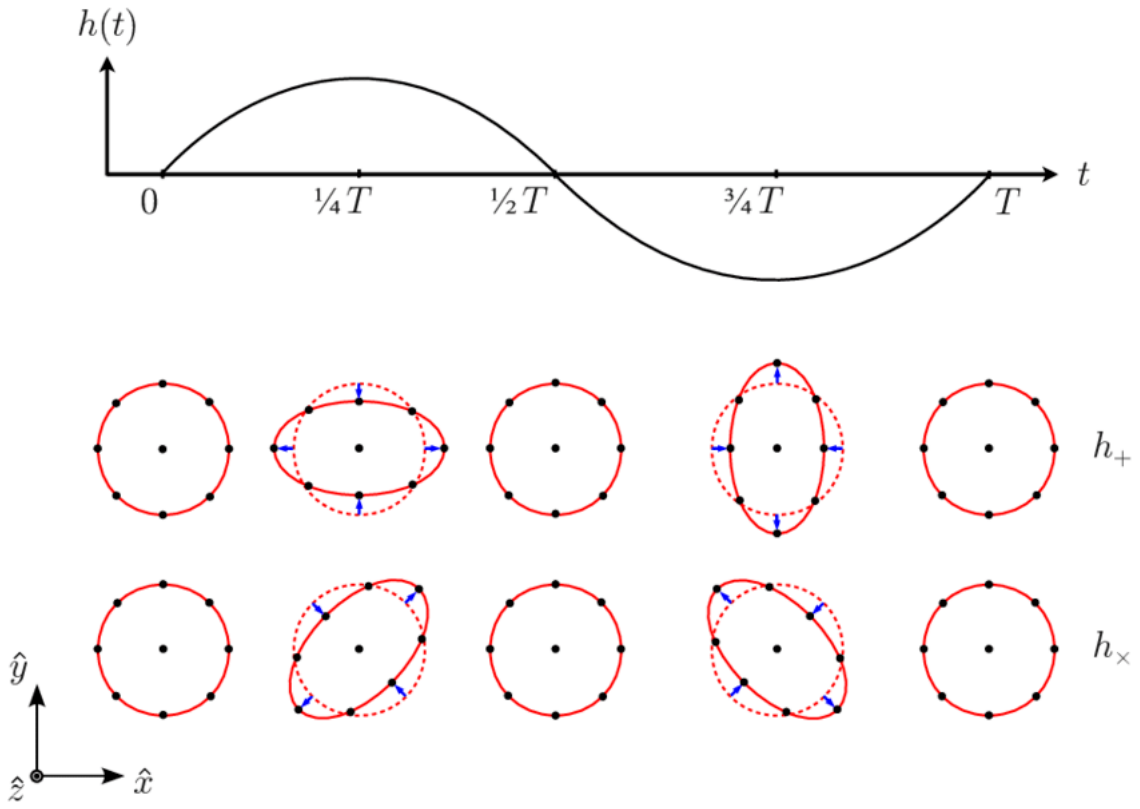

Figure 1.2: Illustration of a GW polarizations oscillating over the course of one period, T. Image credit [19].

object or a dynamical system in space. It is particularly important in systems like compact binary objects, such as BBHs, NSBHs, or binary neutron stars (BNSs), where two massive objects orbit each other. As these objects move, their mass distribution changes, which generates GWs [18].

The quadrupole moment, $Q_{\mu\nu}$, of a system is a tensor that represents deviations from spherical symmetry in the mass distribution. It is defined as:

$$Q_{\mu\nu}^{\mathrm{TT}} = \int d^3x \rho \left(x_\mu x_\nu - \frac{1}{3}\delta_{\mu\nu} r^2 \right), \tag{1.17}$$

where ρ is the mass density, x_μ and x_ν are spatial coordinates, and $\delta_{\mu\nu}$ is the Kronecker delta. In GW emission, the lowest-order contribution comes from changes in the quadrupole moment of the system. Monopole contributions, representing the overall mass, do not vary in time and therefore cannot generate GWs. Dipole contributions correspond to the center of mass, which follows a geodesic and does not radiate. Thus, the quadrupole moment provides the dominant and lowest-order contribution that leads to GW emission, making the quadrupole approximation an effective tool for describing these waves [18].

Using the quadrupole approximation [18], the GW amplitude $h_{\mu\nu}$ can be expressed as:

$$h_{\mu\nu} = \frac{2G}{rc^4}\ddot{Q}^{\mathrm{TT}}_{\mu\nu}\left(t - \frac{r}{c}\right), \tag{1.18}$$

where G is the gravitational constant, c is the speed of light, r is the distance to the observer, and $\ddot{Q}^{\mathrm{TT}}_{\mu\nu}$ is the second time derivative of the quadrupole moment. For simplicity, the amplitude of the GW, h, can be described as:

$$h \propto \frac{1}{r}\ddot{Q}, \tag{1.19}$$

where $\ddot{Q}$ represents the second time derivative of the quadrupole moment. This relationship indicates that the more rapidly the mass distribution changes, the stronger the emitted GW signal becomes. In compact binary coalescences (CBCs), such as when two black holes or neutron stars merge, the GW polarizations are directly related to the evolution of the system's quadrupole moment. During the inspiral and merger phases, rapid changes in the mass distribution amplify the gravitational radiation, which we detect as GWs.

In the case of CBCs, where two compact objects merge, the polarisation modes of the GW are tied to the changes in the system's quadrupole moment over time. In this approximation, the two linear polarisations of the GW can be quantified as

$$h_+(t) \equiv \frac{2\mathscr{M}_z}{D_{\mathrm{L}}}\left(1+\cos^2(\iota)\right)(\pi\mathscr{M}_z f)^{\frac{2}{3}}\cos(\Phi+\Psi), \tag{1.20}$$

$$h_\times(t) \equiv \frac{4\mathscr{M}_z}{D_{\mathrm{L}}}\cos(\iota)(\pi\mathscr{M}_z f)^{\frac{2}{3}}\sin(\Phi+\Psi), \tag{1.21}$$

where $h_\times(t)$ and $h_+(t)$ represent the "cross" and "plus" polarisations of the GW strain at time t. The term $\mathscr{M}_z$ represents the redshifted chirp mass of the source, defined as the reduced mass between the two masses, m_1 and m_2, as

$$\mathscr{M}_z = (1+z)\left(\frac{(m_1 m_2)^{3/5}}{(m_1+m_2)^{1/5}}\right), \tag{1.22}$$

where z is the redshift. The term D_{L} is the luminosity distance to the source, ι represents the inclination angle defined as the angle between the line of sight and the orbital angular momentum vector (for a non-spinning system), f is the frequency of the GW, Φ is the intrinsic phase of the GW related to the orbit of the system, and the Ψ is the polarisation

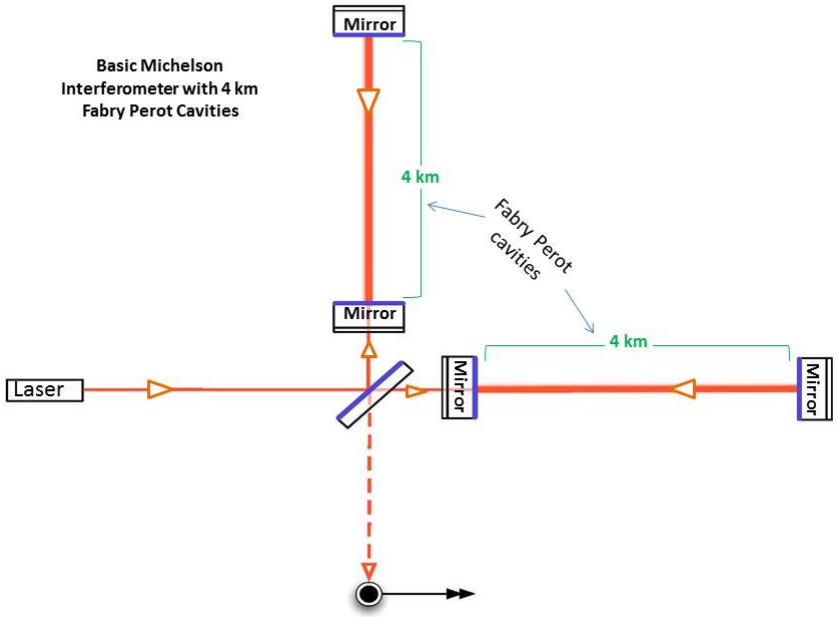

Figure 1.3: Diagram of a Michelson interferometer with 4-kilometer-long Fabry-Perot cavities in each arm. A laser beam is split into two perpendicular paths, traveling down orthogonal arms with mirrors at both ends. The Fabry-Perot cavities increase the effective path length, enhancing sensitivity to GWs. The reflected beams recombine at the beam splitter, creating an interference pattern, which allows the measurement of small changes in arm-length caused by passing GWs. Image credit [22]

angle [20, 21]. Eq.(1.20) and Eq.(1.21) are of fundamental importance, as they give us a more concrete quantification of the strain terms, and they tell us the strain of GWs is inversely proportional to the luminosity distance and linearly proportional to the chirp mass (the closer and more massive the source is, the higher the relative strain).

1.3.2 Detecting Gravitational Waves

Understanding the nature of GWs has led scientists to explore how these phenomena may be observed by detectors. As a GW passes through space, it causes tiny oscillations in the relative distances between objects. These relative length changes can be measured using a Michelson Interferometer. The instrument has two perpendicular arms through which light travels back and forth. When a GW passes, it stretches one arm and compresses the other, resulting in a small but measurable difference in light travel time. This difference creates an interference pattern, generated by constructive and destructive interference of the laser with itself, enabling scientists to detect and observe the impact of the GW by measuring the change in the relative lengths of the interferometer's arms.

A Michelson interferometer divides a light beam into two perpendicular beams, which travel along separate paths before recombining. A depiction of the setup at the Laser Interferometer Gravitational Wave Observatory (LIGO) can be seen in Fig 1.3. By comparing the combined light with the original light, one can detect differences in the path lengths of the two beams. These differences result in either constructive or destructive interference, depending on the phase shift between the beams. This interference pattern reveals small variations in the arm-lengths, allowing the detection of GWs. This is the main idea adopted by LIGO, where in this case the light paths after the beam splitter are 4 kilometers long enclosed in vacuum sealed tubes. The reason for the 4 kilometer long arms is due to the extremely small effects of a GW on the path length of the laser beams. The longer the arms, the more the signal caused by the GW is amplified, making it easier to detect the minute changes in length that these waves induce.

The extended arm-length increases the sensitivity of the interferometer, allowing it to measure changes in distance as small as a fraction of the width of a proton [23]. Consider a test particle located at a fixed position along the y-axis within the interferometer, specifically at the coordinates $(0,0,L_y,0)$ at some initial time. A GW is assumed to be incident on the interferometer, traveling along the z-axis (perpendicular to the plane of the interferometer arms). The spacetime interval ds^2, which accounts for both spatial and temporal separations, is described using the perturbed metric tensors due to the GW as

$$\begin{aligned} ds^2 &= g_{\mu\nu}dx^\mu dx^\nu \\ &= (\eta_{\mu\nu}+h_{\mu\nu})dx^\mu dx^\nu \\ &= -c^2dt^2+(1+h_+)dx^2+(1-h_+)dy^2+2h_\times dxdy+dz^2. \end{aligned} \tag{1.23}$$

The terms dt, dx, dy, and dz correspond to infinitesimal displacements in time and the spatial directions x, y, and z, respectively. The terms h_+ and $h_\times$ correspond to the two polarization states of the GW. This equation describes how spacetime is stretched and squeezed in the x and y directions in the presence of the GW, affecting the infinitesimal interval ds for the test particle. Also, since the light beam within the cavity tube of the interferometer travels at the speed of light, hence, $ds^2 = 0$ [13, 18]. Therefore, by considering the path of a photon traveling solely along the y-axis of the interferometer, we set $dx = dz = 0$ in Eq. (1.23), and by setting the left-hand side to zero and using the

approximation $\sqrt{1-h_+} \approx (1-\frac{1}{2}h_+)$, where $h_+ \ll 1$, follows

$$\begin{aligned} \int_0^{t_0} c dt &= \int_0^{L_y} \sqrt{1-h_+} dy \\ ct_0 &= L_y\left(1-\frac{1}{2}h_+\right) \\ L_y' &= L_y - \frac{1}{2}L_y h_+ \end{aligned} \tag{1.24}$$

where $ct_0 = L_y'$ represents the length that the photon would have traveled in the presence of the GW. The second term on the right-hand side of the equation represents the change in path length compared to the case where no GW is present. Therefore, we can express the change in the path length as

$$\Delta L_y = -\frac{1}{2}L_y h_+. \tag{1.25}$$

The path difference shown in Eq. (1.25) shows how the perturbation of the GW changes the optical path length of the laser. By performing the same analysis for the x-axis, we find the same flight path change in the opposite direction, as $\Delta L_x = \frac{1}{2}L_x h_+$. Therefore, to account for the total change in the interferometer's path length, we must consider both the changes in the x-axes and y-axes and the fact the light travels down each arm and back. Since the round-trip flight path is doubled, the total path length change can be expressed as

$$\Delta L_{\text{total}} = 2\Delta L_x - 2\Delta L_y = 2Lh_+, \tag{1.26}$$

where for the same arm-length detector, $L_x = L_y = L$. Equation (1.26) is regarded as the strain-to-length relation for GW detections via Michelson interferometers.

Although the 4 kilometer length of the interferometer arms may seem substantial, simple Michelson interferometers of this size would be insufficient for detecting GWs. According to Eq. (1.26), the change in path length caused by a GW is proportional to the arm-length and the GW strain, h_+. However, the strain produced by GWs is extremely small, often on the order of 10^{-21}, depending on factors like distance from the source and the masses involved. This means that even with 4 kilometer arms, the resulting displacement would only be a fraction of a proton's diameter, making direct detection incredibly difficult. To overcome this limitation, an additional mirror is placed in each arm near the beam splitter, with another mirror 4 kilometers away at the terminal end of the arm. This setup forms a Fabry-Perot cavity, where light is reflected multiple times within each arm, approximately 300 times, effectively increasing the optical path length and amplifying the sensitivity of the interferometer [24]. By increasing the number of round trips the light makes, the interferometer can detect the tiny perturbations caused by GWs more effectively. It is important to note, however, that the benefit of longer arms relies on the long-wavelength

approximation, where the wavelength of the GW is much larger than the arm-length. If the frequency of the GW was too high, the wavelength would be comparable to or shorter than the arm-length, leading to cancellation effects that reduce the detector's sensitivity. Therefore, detectors like LIGO are designed to optimise sensitivity for a specific frequency range of GWs, where the long wavelength approximation holds.

1.3.3 Noise sources and Matched Filtering

While the interferometer is designed to maximize sensitivity, detecting GWs is complicated by the presence of various noise sources. These noise sources are frequency-dependent and contribute to the overall noise profile of the detector, typically expressed using the amplitude spectral density (ASD). The ASD represents how the noise varies across different frequencies, setting the limit on the smallest GW strain that can be detected at each frequency [24]. In detectors like LIGO and Virgo, the noise is characterized by several sources, including quantum sensing noise, seismic noise, suspension thermal noise, mirror coating thermal noise, and gravity gradient noise [24, 25]. Additionally, transient noise events can occur due to human activities, weather conditions, or equipment malfunctions [26]. All of these noise sources directly affect the sensitivity of the detectors and are represented in the ASD. The PSD, denoted as $S(f)$, characterizes how the power of noise is distributed across different frequencies. It describes the noise power per unit frequency, providing insight into the frequency components of the noise present in the detector. The units of the PSD are typically strain^2/Hz, reflecting the squared amplitude of the noise at each frequency. The ASD is derived as the square root of the PSD, which makes it more intuitive to interpret in terms of amplitude. It provides a clear picture of the detector's sensitivity across various frequency bands, showing how much noise amplitude exists per unit frequency. A representation of the ASD for both LIGO detectors, Hanford and Livingston, is shown in Fig. 1.4, during the O1 observing run. The figure highlights the characteristic shape of the ASD, demonstrating that both detectors are most sensitive to GW events in the frequency range of 10 Hz to 1000-2000 Hz. The frequency range between 10 Hz and 1000-2000 Hz is of general interest for LIGO, since this is where GWs from a subset of astronomical events, like the merging of stellar mass black holes or neutron stars, are expected to be found [24].

Since the noise has now been summarized and characterized, the next step is to extract the GW signal from it using the method of matched filtering [28]. Matched filtering is a technique for detecting signals buried in noise when the form of the signal is known in advance. This method works by comparing the data with a set of precomputed theoretical

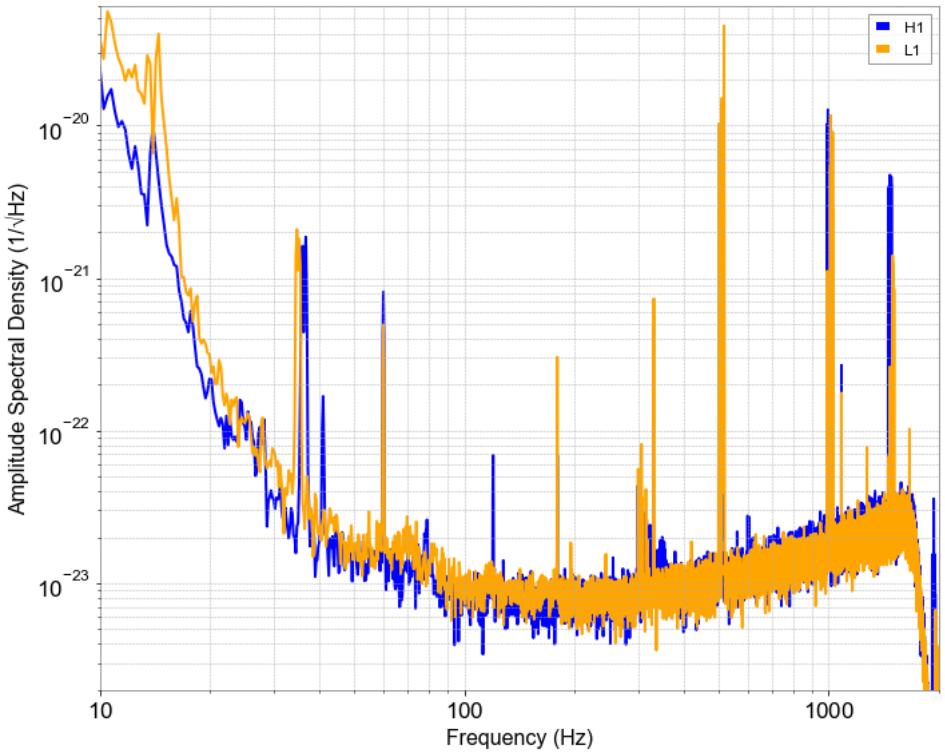

Figure 1.4: The Amplitude Spectral Density (ASD) of LIGO Hanford (H1) and Livingston (L1) on September 14, 2015, 09:50:45 UTC. The plot was generated using `gwpy` [27].

waveforms, known as a template bank, which represent the expected gravitational waveforms from various astrophysical sources [29]. The goal of matched filtering is to determine if the data contains any signals that closely resemble one of the templates in the bank. Since the template waveforms describe GWs from a range of CBCs events, any sufficiently strong signal should be detected by this method. Assuming the detector's output is given by $s(t) = h(t) + n(t)$, where $h(t)$ represents the GW signal and $n(t)$ is stationary Gaussian noise, and the equivalent frequency-domain representation is given as $\tilde{s}(f) = \tilde{h}(f) + \tilde{n}(f)$, the detector's output is then cross-correlated with the template waveform $h(t)$, with the filtered signal $z(t)$ computed as

$$z(t) = \int s(t')h(t'-t)dt', \tag{1.27}$$

which measures the similarity between the observed signal $s(t)$ and the template $h(t)$. The match is highest when the GW signal aligns with the template waveform, and when the template accurately represents the shape of the signal. For CBCs, this means having the correct mass parameters and potentially the correct spins, ensuring the template best reflects the true GW signal [29]. To estimate the strength of the signal in the detector's data stream, the matched filtering SNR, ρ, is computed as:

$$\rho^2 = 4\Re\left\{\int_0^\infty \frac{|\tilde{h}(f)\tilde{s}^*(f)|^2}{S_n(f)}df\right\}, \tag{1.28}$$

where $\tilde{h}(f)$ is the Fourier transform of the template waveform $h(t)$, $\tilde{s}(f)$ is the Fourier transform of the detector's signal $s(t)$, and $S_n(f)$ is the noise PSD of the detector, representing noise as a function of frequency. The integral is computed over the entire frequency range of the detector, though in practice, it is typically calculated over the frequency band for which the template was constructed, usually starting at a low cutoff frequency. The SNR defined in Eq. (1.28) effectively compares the signal $\tilde{s}(f)$ to the noise $S_n(f)$ in the frequency domain. Using the definition given in Eq. (1.28) for the matched filter SNR, we can define the power of each template waveform in terms of the optimal SNR (ρ_{opt}) as

$$\rho_{\text{opt}} = \sqrt{4 \int_{f_{\text{min}}}^{f_{\text{high}}} \frac{|\tilde{h}(f)|^2}{S_n(f)} df} \tag{1.29}$$

where the integral spans the frequency range over which the template was evaluated [28].

Therefore, by calculating the optimal SNR for each template in the bank, and comparing it with the matched filter SNR, if the matched filter SNR exceeds the detection threshold for a particular template, it suggests that the observed data closely matches the predicted waveform of a potential astrophysical event. However, it is important to account also for the false alarm rate (FAR) when setting a detection threshold [30]. The FAR measures how often noise fluctuations in the detector mimic genuine GW events. These false alarms can result from transient noise, environmental disturbances, or random fluctuations within the detector data, collectively known as "glitches" . When setting the SNR threshold, if the value is too low, the detector becomes more sensitive to noise, increasing the probability of false positives.

Once a candidate GW event has been identified in the data, the next step is to perform parameter estimation (PE) to extract key physical properties of the source, such as masses, spins, and distances. This process relies on Bayesian statistics, where the observed data is used to refine our understanding of the source parameters. The next section will explain Bayesian inference in detail, illustrating how this framework allows us to compare observed signals with theoretical models and derive astrophysical insights from the GW event.

1.4 Bayesian statistics

To understand Bayesian statistics, we first need to explore conditional probability. The probability of event A occurring given event B can be written as

$$p(A|B) = \frac{p(A \cap B)}{p(B)}, \tag{1.30}$$

where $\cap$ denotes the intersection of events A and B. Similarly, the probability of event B given event A is

$$p(B|A) = \frac{p(B \cap A)}{p(A)}. \tag{1.31}$$

Since $p(A \cap B) = p(B \cap A)$, we can rearrange both equations, yielding Bayes' theorem

$$p(A|B) = \frac{p(A)p(B|A)}{p(B)}. \tag{1.32}$$

From this expression, Bayes' theorem forms the foundation of Bayesian inference, where a probability model is applied to observed data to update our understanding of the distribution of parameters and make predictions [31]. This framework combines prior knowledge with observed evidence to refine the model's parameters. In the case of Eq.(1.32), we can substitute the parameters of a model we wish to infer, $\boldsymbol{\theta}$, which substitutes the A term, conditioned on a measured data set, defined by X, which replaces the term B. The outcome is a comprehensive probability distribution reflecting the parameters and potential predictions. After making the substitutions, Bayes' theorem becomes:

$$p(\boldsymbol{\theta}|X) = \frac{p(\boldsymbol{\theta})p(X|\boldsymbol{\theta})}{p(X)}, \tag{1.33}$$

where $p(\boldsymbol{\theta}|X)$ represents the posterior, which is our updated knowledge of the parameters $\boldsymbol{\theta}$ given the data X. The term $p(\boldsymbol{\theta})$ is the prior, reflecting our initial belief about $\boldsymbol{\theta}$ before seeing the data. The likelihood, $p(X|\boldsymbol{\theta})$, describes the probability density of observing the data X given the parameter values $\boldsymbol{\theta}$. Finally, $p(X)$ is the evidence, serving as a normalisation factor to ensure the posterior is a valid probability distribution.

The posterior distribution integrates our prior beliefs with newly acquired data to update our knowledge of the paramters $\boldsymbol{\theta}$. This updated belief becomes the basis for further analyses. For example, if we have two independent events, X_1 and X_2, we first update our understanding using the posterior from event 1. The resulting posterior then serves as the

prior for event 2. This can be expressed as

$$p(\theta|X_2,X_1) = \frac{p(\theta|X_1)p(X_2|\theta)}{p(X_2)} \quad = \frac{p(\theta)p(X_1|\theta)p(X_2|\theta)}{p(X_1)p(X_2)}. \tag{1.34}$$

For multiple measurements, this generalizes to a product over all N measurements

$$p(\theta|X_1,\ldots,X_N) = p(\theta)\prod_{i=1}^{N}\frac{p(X_i|\theta)}{p(X_i)}. \tag{1.35}$$

This expression shows how each new piece of data progressively increases our knowledge of the parameters θ, where N represents the number of independent events and i denotes the index of the ith event. When combining multiple events, the likelihood terms (and evidence terms) are multiplied for each event, but the prior is applied only once. The prior represents our initial belief about the parameters before observing any data, so it is not updated with each event [31].

In cases where certain parameters are of less interest, we can marginalise over them. Marginalisation involves integrating out nuisance parameters to focus on the key parameters of interest. For example, given parameters θ_1 and θ_2, if θ_2 is a nuisance parameter, we can marginalise it by integrating the joint posterior

$$p(\theta_1|X) = \int p(\theta_1,\theta_2|X)d\theta_2. \tag{1.36}$$

This process simplifies the inference by reducing the dimensionality of the parameter space, focusing on the parameters most relevant to the analysis. Next, we will briefly describe how PE is performed within this Bayesian framework to infer both GW parameters and cosmological or population parameters.

1.5 Parameter Estimation

When a GW event candidate is observed, the next step is to perform PE on the measured data to extract key information about the source parameters responsible for the event. This process relies on a Bayesian framework, as described in Sec. 1.4. The primary goal is to estimate the joint posterior distribution for the unknown parameters that define the model for the observed data. From this joint distribution, we can derive marginalised posterior probability distribution functions (PDFs) for individual parameters, which are often used for visualisation. However, the joint distribution contains the complete information about the system. In GW PE, we apply Bayesian inference to compute the posterior

distribution, which represents the probability of the model parameters given the observed data. This posterior is proportional to the product of the likelihood, the probability of the data given the model, and the prior, representing our initial knowledge of the parameters. The likelihood function plays a critical role in evaluating how well the model fits the observed data, incorporating key parameters like the masses and spins of the merging objects as well as the noise characteristics of the detector.

The PE process begins by defining the data, modeled as a combination of the GW signal $h(t)$ and detector noise $n(t)$, which can be expressed as $s(t) = h(t) + n(t)$. This relationship holds in both the time and frequency domains, with the latter written as $\tilde{s}(f) = \tilde{h}(f) + \tilde{n}(f)$ due to the linearity of the Fourier transform. After identifying time segments that may contain GW signals of astrophysical origin, the next step is to analyse the source responsible for these signals. To extract the source parameters, we first define the likelihood of the measured data using the Bayesian framework from Eq. (1.33). However, because the noise in detectors is typically colored (i.e., its amplitude varies with frequency), performing the analysis in the frequency domain simplifies the process. In the frequency domain, noise bins are uncorrelated for stationary noise, allowing us to model the likelihood as the product of independent probabilities across frequency bins [32]. We start by modeling the data with Gaussian noise, thus writing the likelihood as

$$\mathscr{L}(\tilde{d}(f)|\tilde{\theta}) \propto \exp\left(-\frac{1}{2}\sum_{i=0}^{N-1}\frac{|\tilde{s}(f_i) - \tilde{h}(f_i,\tilde{\theta})|^2}{\sigma(f_i)^2}\right), \tag{1.37}$$

where $\sigma(f_i)$ is the frequency-dependent standard deviation of the Gaussian noise. Since the PSD, $S(f_i)$, is linearly related to the Gaussian noise, we have $\sigma^2(f_i) \propto S(f_i)$. To properly account for the detector's noise characteristics, we weight the likelihood by the PSD $S_n(f)$. This is done by defining the inner product, which helps to appropriately weight the likelihood by the frequency-dependent noise [33], as

$$\mathscr{L}(d(f)|\tilde{\theta}) \propto \exp\left(-\frac{1}{2}\left\langle s(f) - h(f,\tilde{\theta}) \mid s(f) - h(f,\tilde{\theta})\right\rangle\right), \tag{1.38}$$

where the inner product weighted by the PSD is defined as [33]

$$\langle a(t)|b(t)\rangle = 4\Re\left\{\int_{-\infty}^{\infty}\frac{\tilde{a}^*(f)\tilde{b}(f)}{S_n(f)}, df\right\}, \tag{1.39}$$

where $\Re$ denotes the *real* part of the complex valued result of the right-hand side, the values $\tilde{a}^*(f)$ is the complex conjugate of the Fourier transform of the time-dependent value $a(t)$.

1.5. Parameter Estimation

To estimate the source parameters, $\tilde{\theta}$, we explore the parameter space to obtain samples from the posterior distribution, which reflects our updated knowledge of the parameters given the observed data. This process involves evaluating the likelihood within an N-dimensional parameter space, where N represents the number of parameters in $\tilde{\theta}$. A straightforward approach would be to explore this space using a grid, where each dimension corresponds to a different parameter. However, this method becomes computationally expensive as the number of parameters increases. For example, if n points are evaluated along each dimension, a grid search would require n^N evaluations. For high-dimensional problems, this quickly becomes infeasible. To address this challenge, stochastic sampling techniques, such as Monte Carlo Markov Chain (MCMC) [34], provide a more efficient way to explore high-dimensional posteriors by focusing computational resources on regions with higher probability densities. The method works by generating a sequence of samples (a "Markov chain") from the posterior distribution. Since each sample depends only on the previous one, the algorithm is able to efficiently explore regions of higher probability density. As the chain progresses, it spends more time in areas with higher probability, ensuring the posterior is accurately represented [34]. This approach improves efficiency and allows for accurate calculation of various posterior-derived quantities [35]. Additionally, marginalisation over unwanted parameters becomes more straightforward, as explained in Sec.1.4. However, MCMC can be inefficient when dealing with complex posterior distributions, such as multimodal distributions or those with long, narrow degeneracies. Since MCMC uses random walks to explore the parameter space, this can lead to slow convergence and difficulty in thoroughly exploring all areas, especially in high-dimensional spaces.

Nested sampling (NS) [36] is a technique designed to efficiently explore complex parameter spaces, particularly useful for multimodal distributions. It works by progressively shrinking the volume of the parameter space, focusing on regions with higher likelihoods. Introduced by Skilling in 2004 [36], NS transforms the multi-dimensional integral of the evidence into a one-dimensional integral that is easier to compute. The evidence Z is given by:

$$Z = \int \mathscr{L}(\theta)\pi(\theta)\, d\theta, \tag{1.40}$$

where $\mathscr{L}(\theta)$ is the likelihood function that measures how well the parameters θ explain the observed data, and $\pi(\theta)$ is the prior distribution encoding prior knowledge or assumptions about these parameters. Direct computation of this integral can be challenging due to the complexity of the parameter space. The method of NS addresses this by sorting the parameter space by likelihood values and transforming the integral into a one-dimensional form:

$$Z = \int_0^1 \mathscr{L}(X)\, dX, \tag{1.41}$$

where X represents the cumulative prior mass, defined as the fraction of the prior volume where the likelihood is greater than a given threshold. Mathematically, X can be expressed as:

$$X(\lambda) = \int_{\mathscr{L}(\theta)>\lambda} \pi(\theta)\, d\theta, \tag{1.42}$$

where $\mathscr{L}(\theta)$ is the likelihood function, $\pi(\theta)$ is the prior probability density, and λ is the likelihood threshold.

The NS method operates iteratively. Initially, a set of N random samples, or live points, is drawn from the prior distribution. For each iteration, the likelihood $\mathscr{L}(\theta)$ is evaluated for all live points, and the sample with the lowest likelihood, $\mathscr{L}_{\min}$, is removed. The evidence is updated by adding $\mathscr{L}_{\min}\Delta X$, where ΔX is the reduction in prior volume. The removed sample is then replaced by a new one drawn from the prior, constrained to regions where the likelihood exceeds $\mathscr{L}_{\min}$. This process progressively narrows the parameter space to regions of higher likelihood [36]. The algorithm terminates when the remaining prior volume contributes negligibly to the evidence. This method can be further enhanced by incorporating a machine-learning-based approach known as `nessai`, which significantly accelerates the process.

In `nessai` [37, 38], at each iteration, a NF is trained using the current live points. This flow maps the live points from the sampling space X to samples in the latent space Z (more information about NF in Sec. 1.8). New samples are drawn by sampling from a truncated latent distribution and applying the inverse mapping f^{-1}. Rejection sampling is then used to ensure that the samples adhere to the prior distribution. This approach results in independently and identically distributed samples, eliminating the need for MCMC sampling. Furthermore, since new points are drawn in parallel, likelihood evaluations can also be parallelised, reducing the algorithm's runtime and accelerating convergence. The evidence in `nessai` is estimated by performing a weighted sum of contributions from the nested samples, with weights assigned based on their likelihoods. Posterior samples are derived by resampling these weighted nested samples, providing a detailed representation of the posterior distribution. The process continues until a convergence criterion, such as a negligible remaining prior volume or a fixed number of iterations, is satisfied. The final outputs include the estimated evidence and a comprehensive set of posterior samples, making `nessai` a powerful tool for Bayesian inference in high-dimensional, multimodal parameter spaces.

Beyond these approaches, alternative methods for performing PE, such as variational autoencoders (VAE) [39] and NFs [40], and many others, have been explored to model complex posteriors more efficiently. While these machine learning-based methods show potential for improved computational speed, particularly in high-dimensional parameter spaces, classical techniques still set the benchmark for accuracy and remain the preferred choice for rigorous PE [32].

After performing PE and obtaining estimates of the GW parameters defining the observed strain data, one of the key parameters derived is the luminosity distance. By combining this measurement with the redshift of the GW event and identifying the host galaxy of the GW signal, it becomes possible to estimate the Hubble constant (H_0) using Eq. (1.5). The parameter obtained in the PE process related to mass is the detector-frame chirp mass, which is the observed chirp mass as measured by the detector. This mass is redshifted, meaning it is multiplied by a factor of $(1+z)$, where z is the redshift. This factor accounts for the expansion of the universe, which stretches the wavelengths of GWs and affects the observed mass values. In contrast, the source frame chirp mass represents the intrinsic mass of the GW source, unaffected by the expansion of the universe. This distinction leads to a degeneracy between the redshift and the source frame chirp mass, as the redshifted mass measured in the detector frame cannot independently provide the redshift information. Therefore, external sources of redshift information must be obtained to break this degeneracy. This is where the idea proposed by Schutz [12] comes into play. Schutz suggested combining the calibrated GW luminosity distance estimates with redshift measurements from galaxy catalogues to estimate cosmological parameters, such as the Hubble constant H_0. In the next section, we will explore this concept more thoroughly, discussing how this idea has evolved and been integrated into cosmological analysis pipelines. We will also examine how these pipelines are used to estimate key cosmological parameters, such as H_0.

1.6 Cosmology with Gravitational Waves

1.6.1 A short literature review

In 1986, B. Schutz proposed the groundbreaking idea of using GW events to measure the expansion rate of the universe [12]. This marked the beginning of a new era in cosmology, where GWs could serve as “standard sirens ”for cosmological studies. However, an immediate challenge arose: the redshift information encoded in GWs is completely degenerate

with the redshifted chirp mass, $\mathscr{M}_z$. This means that while GWs can provide a measure of the luminosity distance, they cannot directly give us the redshift needed for cosmological measurements. To overcome this, Schutz proposed using galaxy catalogues, which contain measured properties of galaxies, such as redshift and their uncertainties. By combining the luminosity distance obtained from GW events with the redshift information from these catalogues, cosmological analyses can be performed.

Building on this foundation, Finn and Chernoff [41] extended this idea by exploring how cosmological parameters, particularly H_0 and the deceleration parameter (q_0) [1], could be measured using observations of inspiraling binary systems, such as neutron stars or black holes, detected via GWs. Their work focused on LIGO-like detectors and estimated detection rates based on SNR, H_0, and the binary chirp mass. They estimated that to measure H_0 with 10% precision, around 100 binary inspiral detections with $\rho_0 = 10$ would be required. Similarly, determining the deceleration parameter q_0 with 20% accuracy would require around 3000 observations.

In 2011, Walter Del Pozzo further developed this framework in his work [43], introducing a Bayesian inference framework to address the challenge of extracting cosmological information from GW events. This framework integrated all prior assumptions and information about a GW source into a single, unified data analysis method. It minimised information loss and allowed for the incorporation of event-specific details, such as the sky position in Gamma-Ray Burst (GRB)–GW coincident detections. By applying Bayesian techniques, the method allowed for the seamless inclusion of additional data, such as galaxy surveys, to improve redshift estimation. Building on Schutz's work, Del Pozzo used galaxy catalogues to associate redshifts with GW sources, identifying the host galaxies of GW events. He demonstrated that by combining results from several tens of GW observations using a global network of advanced interferometers, it would be possible to constrain H_0 to within 4–5% accuracy at a 95% confidence level.

Other works, such as those by Samaya Nissanke [44], focused on the potential of BNS mergers as cosmological tools, particularly through their connection with short gamma-ray bursts (SGRBs). Observational evidence strongly suggests that some SGRBs are linked to the mergers of compact objects like BNS systems. These mergers emit GWs during their final inspiral and produce EM counterparts, such as kilonovae, visible in optical and infrared bands due to the radioactive decay of elements formed in the merger. The sim-

1. Currently, the deceleration parameter q_0 is related to the more commonly used cosmological parameters, such as the matter density parameter Ω_m and dark energy density parameter Ω_Λ, through the equation $q_0 = \frac{1}{2}\Omega_m - \Omega_\Lambda$ [42]

ultaneous detection of both GW and EM signals offers a unique opportunity to measure both the luminosity distance and the redshift of the binary system. Nissanke's analysis indicated that detecting 15 simultaneous GW and EM events using a network of advanced detectors, such as LIGO and Virgo, could measure H_0 with 5% precision. With 30 beamed GW-SGRB events, this precision could improve to better than 1%, highlighting the potential of multi-messenger astronomy to refine cosmological measurements.

Unlike traditional distance-ladder methods that depend on astrophysical systematics, Messenger and Read [45] demonstrated that GWs from BNS mergers could, with future third-generation detectors like the Einstein Telescope (ET), enable precise measurement of tidal effects that modify the GW signal. This approach represents a promising avenue for improving cosmological measurements once more advanced detectors become operational. These tidal interactions introduce phase changes that break the degeneracy between mass parameters and redshift, allowing independent estimation of both effective distance and redshift without EM counterparts. By analysing tidal information, their research provides a robust and independent method for cosmological inference, directly measuring rest-frame masses and redshifts. This approach significantly improves cosmological measurements, with studies indicating redshift accuracy of 8-40% for $z < 1$ and 9-65% for $1 < z < 4$, determined from GW observations.

Following these developments, Taylor and Gair [46] proposed a novel approach to measuring H_0 using GW signals from compact binary systems, particularly BNS mergers. Their work capitalized on the relatively narrow distribution of neutron star masses to use these binaries as "standard sirens" for cosmological measurements. While GW observations can directly measure the distance to the source, determining the redshift presents a challenge due to its degeneracy with the system's chirp mass in GW signals. Taylor and Gair explored a method to extract cosmological information from GW observations alone, without relying on EM counterparts. Using a Bayesian formalism and catalogues of BNS inspiral detections, they investigated what could be learned about background cosmology and neutron star mass distributions from GW data alone. Their analysis showed that, under reasonable assumptions, the Hubble constant could be determined to within $\pm 10\%$ accuracy using around 100 observations, provided the Gaussian half-width of the underlying neutron star mass distribution is less than $0.04 M_{\odot}$.

Following the GW detection of GW170817, the first multi messenger event where both EM radiation and GWs were detected [47], initial measurements of H_0 were made using key pipelines, such as `IcaroGW` and `gwcosmo`. The most recent updates on constraining cosmological parameters, particularly H_0, are outlined in the LVK collaboration work [16],

where these pipelines were employed to estimate both cosmological and population parameters using multiple detected events from the GWTC-3 catalogue [48]. In this research from the LVK collaboration, data from 47 GW events were used to estimate H_0. Each GW event provided the luminosity distance to its source, with redshifts inferred through two methods: redshifted BBH masses and a galaxy catalog. The first method used the redshifted BBH masses to simultaneously infer the source mass distribution and the Hubble parameter $H(z)$. A peak in the BBH mass distribution was identified at approximately $34M_\odot$, followed by a decline. Assuming this mass distribution did not evolve with redshift, the study estimated $H_0 = 68^{+12}_{-8}$ km s^{-1} Mpc^{-1} (68% credible interval), combined with data from GW170817 and its electromagnetic counterpart [49]. The second method associated each GW event with its probable host galaxy from the GLADE+ catalog, using the `gwcosmo` pipeline, statistically marginalising over the redshifts of each potential host. This method yielded a similar H_0 value of 68^{+8}_{-6} km s^{-1} Mpc^{-1}. However, this result was significantly impacted by assumptions regarding the BBH mass distribution. The event most informative about H_0, and least affected by these assumptions, was the well-localized GW190814 [16, 50].

The `gwcosmo` method [14, 51, 52] is a well-documented and widely used Bayesian analytical approach for inferring both cosmological and population parameters from GW events, particularly through posterior samples of their parameters and EM data from galaxy catalogues. Since this thesis will discuss a novel method for inferring H_0, which builds on the methodologies and assumptions of `gwcosmo`, we will now briefly outline the key assumptions and methodologies behind this analysis and provide an overview of how it operates.

1.6.2 Cosmology with `gwcosmo`

The `gwcosmo` [14, 51, 52] method is a Bayesian framework used to estimate the Hubble constant and other cosmological parameters from GW data. This framework allows for the incorporation of both GW events and EM counterparts when available, as well as galaxy catalogues, to statistically infer H_0 and other key parameters [51]. At the core of the `gwcosmo` methodology is the computation of the posterior probability on H_0 from a set of $N_{\rm det}$ detected GW events, expressed as

$$p(H_0|\{x_{\rm GW}\},\{D_{\rm GW}\}) \propto p(H_0)p(N_{\rm det}|H_0)\prod_{i=1}^{N_{\rm det}} p(x_{{\rm GW},i}|D_{{\rm GW},i},H_0), \tag{1.43}$$

where, $p(H_0)$ is the prior on the Hubble constant, and $p(N_{\rm det}|H_0)$ represents the probability of detecting $N_{\rm det}$ events, which is related to the intrinsic event rate, R, and the observation volume. In this context, $p(N_{\rm det}|H_0)$ depends on both the intrinsic event rate R and the volume of the universe that can be observed, which is itself dependent on H_0. Normally, this would create a complex dependence of the posterior on H_0 through $N_{\rm det}$. However, by adopting the scale-free prior $p(R) \propto 1/R$, the dependence on the value of H_0 cancels out because the prior does not impose any specific scale for the event rate [53].

The term $p(x_{{\rm GW},i}|D_{{\rm GW},i},H_0)$ denotes the likelihood of the GW data ($x_{\rm GW}$), conditioned on the detectability status ($D_{\rm GW}$) and the Hubble constant. Using Bayes' theorem follows

$$p(x_{\rm GW}|D_{\rm GW},H_0) = \frac{p(x_{\rm GW}|H_0)}{p(D_{\rm GW}|H_0)}. \tag{1.44}$$

This likelihood depends on the observed GW data $x_{\rm GW}$ and accounts for the fact that the event was detected (i.e., passed a SNR threshold). The denominator, $p(D_{\rm GW}|H_0)$, involves integrating over the possible realizations of all detectable GW events. This last term can then be expanded and written as

$$p(D_{\rm GW}|H_0) = \int p(D_{\rm GW}|x_{\rm GW},H_0)p(x_{\rm GW}|H_0)dx_{\rm GW}. \tag{1.45}$$

Then, within the mathematical framework of Ref. [51], the term $p(D_{\rm GW}|H_0)$ only appears in an expanded form, where it is additionally conditioned on redshift z and sky location Ω. Therefore, to calculate $p(D_{\rm GW}|z,\Omega,H_0)$, it needs to be integrated over all realizations of GW events for a range of z, Ω, and H_0 values, with the detection threshold $\rho_{\rm th}$ applied. In practice, this calculation can be performed using Monte Carlo integration

$$p(D_{\rm GW}|z,\Omega,H_0) = \frac{1}{N_{\rm samples}} \sum_{j=1}^{N_{\rm samples}} p(D_{{\rm GW},j}|x_{{\rm GW},j},z,\Omega,H_0), \tag{1.46}$$

where $x_{{\rm GW},i}$ corresponds to a specific GW event, with parameters drawn randomly from the prior distributions of quantities affecting the event's detectability (e.g., mass, inclination, polarization, and sky location). The SNR ρ_i of each event is then calculated for specific values of z and H_0. Then, the quantity $p(D_{{\rm GW},i}|x_{{\rm GW},i},z,\Omega,H_0)$ is defined as

$$p(D_{{\rm GW},i}|x_{{\rm GW},i},z,\Omega,H_0) = \begin{cases} 1, & \text{if } \rho > \rho_{\rm th}, \\ 0, & \text{otherwise.} \end{cases} \tag{1.47}$$

This provides a smooth function for $p(D_{\rm GW}|z,\Omega,H_0)$, which transitions from 1 to 0 over a range of values for z, Ω, and H_0 as the detection threshold is applied.

In the absence of an EM counterpart, the `gwcosmo` framework makes use of galaxy catalogues to infer redshift information. The framework marginalises over the possibility that the host galaxy may or may not be included in the catalog, considering the apparent magnitude threshold of the catalog. The likelihood for each event is written as a combination of two cases:

$$p(x_{\rm GW}|D_{\rm GW},H_0) = \sum_{g=G,\bar{G}} p(x_{\rm GW}|g,D_{\rm GW},H_0)p(g|D_{\rm GW},H_0). \tag{1.48}$$

Here, G denotes the case whether the host galaxy is in the catalogue, and $\bar{G}$ denotes the case where the host is not. This method allows for a statistical inference of H_0 even when the galaxy catalogue is incomplete. In this context, incompleteness refers to the fact galaxy catalogues do not contain all galaxies, as many are too dim to be observed. To address this limitation, a magnitude threshold is applied, which acts as a cutoff to make the catalogue boundaries clearer by excluding galaxies that are too faint to detect, even though the true extent of the catalogue's reach remains uncertain. When incorporating galaxy catalogues, the likelihood term $p(x_{\rm GW}|G,D_{\rm GW},H_0)$ for events within the catalogue (both in redshift and sky location) can be expanded to include redshift information from the galaxies. The likelihood of detecting a GW event with the host galaxy in the catalogue is

$$p(x_{\rm GW}|G,D_{\rm GW},s,H_0) = \frac{\sum_{i=1}^{N} p(x_{\rm GW}|z_i,\Omega_i,s,H_0)p(s|z_i)p(s|M(z_i,m_i,H_0))}{\sum_{i=1}^{N} p(D_{\rm GW}|z_i,\Omega_i,s,H_0)p(s|z_i)p(s|M(z_i,m_i,H_0))}. \tag{1.49}$$

Here, the probability $p(s|M,H_0)$ accounts for the dependence on luminosity

$$p(s|M,H_0) \propto \begin{cases} L(M(H_0)) & \text{if the GW hosting probability is proportional to luminosity,} \\ \text{constant} & \text{if the GW hosting probability is independent of luminosity,} \end{cases} \tag{1.50}$$

where M is the absolute magnitude of the galaxy, and L is the intrinsic luminosity. The term $p(s|z)$ represents the probability of the merger rate as a function of redshift,

$$p(s|z) \propto \begin{cases} f(z) & \text{if the rate evolves with redshift,} \\ \text{constant} & \text{if the rate does not evolve with redshift,} \end{cases} \tag{1.51}$$

where for $f(z)$ in [51], a constant-rate model was assumed. These adjustments to the redshift prior incorporate galaxy properties, enhancing GW event localization and PE by leveraging both GW and EM observations.

When, instead, an EM counterpart is detected, the methodology incorporates both the GW and EM from the event data to refine the H_0 estimate. The joint likelihood for a GW and EM event is given by

$$p(x_{\rm GW}, x_{\rm EM}|D_{\rm GW}, D_{\rm EM}, H_0) = \frac{p(x_{\rm GW}|H_0)p(x_{\rm EM}|H_0)}{p(D_{\rm GW}|H_0)p(D_{\rm EM}|D_{\rm GW}, H_0)}, \tag{1.52}$$

where $x_{\rm EM}$ denotes the data from the detected EM counterpart, and $D_{\rm EM}$ indicates an EM counterpart was observed. This method assumes that both GW and EM data are independent, and only considers events where the EM counterpart has passed a detection threshold.

In Ref. [14], the `gwcosmo` methodology was extended to address challenges related to the assumption of uniform catalogue completeness. This extension focuses on improving the handling of galaxy catalogue data, accounting for variations in completeness across different sky regions, distances, and depths, thereby enhancing the robustness of cosmological parameter estimation. The uniform completeness assumption can lead to biases in the inference of the Hubble constant due to oversimplification across the sky. Specifically, applying a uniform completeness correction to a non-uniform patch of the sky may result in areas where completeness is overestimated, artificially inflating the contribution from galaxies in those regions. This bias can be particularly pronounced if the host galaxy is missing from the catalog. Conversely, in regions where completeness is underestimated, valuable information about the redshift distribution of galaxies at greater distances can be overlooked.

To mitigate these issues, the methodology introduces an extension that pixelates the sky into equally sized regions, performing an independent analysis for each pixel. This approach allows for the use of a line-of-sight (LOS) prior on the luminosity distance for each GW event within a pixel, combined with an estimate of the completeness for that specific region of the sky. By doing so, both the overestimation and underestimation problems are addressed, making the analysis more robust and theoretically more informative. The main difference from their first paper [51] is the introduction of the pixelated sky approach. For a single GW event, the likelihood definition now begins by marginalising over whether the host galaxy is inside or outside the catalog. Additionally, the likelihood's dependence on sky direction, Ω, must be accounted for. The original continuous likelihood expression is approximated by dividing the sky into $N_{\rm pix}$ equal-sized pixels and summing the contributions from each pixel

$$p(x_{\rm GW}|D_{\rm GW}, H_0, I) = \sum_{n=1}^{N_{\rm pix}} p(x_{\rm GW}|\theta_n, D_{\rm GW}, H_0, I)p(\theta_n|D_{\rm GW}, H_0, I). \tag{1.53}$$

One can assume that the probability of detection is uniform across the sky, which is reasonable given the Earth's rotation during long observation periods, causing the dependence on sky direction Ω_i to cancel out. However, this assumption holds only if the detection probability, $p(D)$, is not conditional on the event arrival time. Since the analysis relies on $p(D|H)$ marginalised over all parameters, including time, this effectively marginalises over right ascension, α, supporting the uniform sky assumption. The likelihood then simplifies to

$$p(x_{\rm GW}|D_{\rm GW},H_0,I) = \frac{1}{N_{\rm pix}} \sum_{i=1}^{N_{\rm pix}} p(x_{\rm GW}|\theta_i,D_{\rm GW},H_0,I). \tag{1.54}$$

This assumes that $p(\Omega_i|H_0,I)$ is independent of H_0 in an isotropic universe, allowing the likelihood to be simplified by averaging over pixels, while ignoring minor sky-dependent effects like declination dependence [14].

1.6.3 Motivation for `CosmoFlow`

In Sec. 1.6.2, we showed how `gwcosmo` operates by constructing a Bayesian posterior on H_0 using a combination of GW data, galaxy catalogues, and EM counterparts when available. The methodology allows for marginalisation over the incompleteness of galaxy catalogues and uses detection thresholds for both GW and EM events. This framework has proven effective in constraining H_0 through the combination of multiple GW events [16].

However, despite its successes, the `gwcosmo` method is not without limitations. One potential drawback is its lack of flexibility when extra parameters are introduced into the analysis, such as complex population models or additional cosmological parameters. As the dimensionality of the parameter space increases, the `gwcosmo` framework may become cumbersome, relying on computationally expensive and hard-to-converge integrations. [2] This issue is particularly pronounced when trying to combine thousands of events in a hierarchical manner, where the computational cost of the integration becomes a bottleneck, especially in large-scale studies with expanding data sets.

This leads to the core focus of this thesis: the proposal of a novel approach, called `CosmoFlow`, which builds upon the methodologies of `gwcosmo`. The `CosmoFlow` analysis is designed to estimate cosmological and population parameters, but unlike `gwcosmo`, it leverages machine learning techniques, specifically normalising flows (NFs), to overcome the computational challenges posed by traditional Bayesian methods. As of the pub-

2. Advancements in this topic are being made, with LOS priors being computed using machine learning-driven models, accelerating their analysis.

lication of this thesis, `CosmoFlow` is capable of computing the posterior distribution of cosmological and population parameters using only GW events from "dark sirens "(i.e. those without an EM counterpart), such as BBH, NSBH, and potentially BNS events. Although BNS mergers may sometimes produce EM counterparts, dark siren analysis is applicable to cases where no such signals are detected, though training on BNS populations has not yet been completed [54]. Machine learning techniques offer a more scalable and efficient approach to computing the posterior distribution of cosmological parameters, allowing for the analysis of large datasets while significantly reducing the computational costs associated with traditional methods.

Before introducing how `CosmoFlow` operates and its underlying methodologies in Chapter 2, we will first provide an overview of machine learning techniques and why they are particularly useful in this context.

1.7 Machine Learning

1.7.1 Introduction

One of the early and influential ways to describe machine learning (ML) comes from Arthur Samuel, who, in 1959, defined it as a "field of study that gives computers the ability to learn without being explicitly programmed" [55]. Although somewhat dated, this description still effectively captures the essence and power of ML. As the availability of data continues to grow, data analysis will likely play an important role in advancing technology because it enables more efficient processing, interpretation, and decision-making from vast datasets that would otherwise be too complex or time consuming for traditional methods.

In computer science, algorithms are designed to solve specific problems and perform tasks by analysing data. However, when we lack a complete understanding of the processes that generate the data, the large amount of available data can help fill in the gaps. Even without full knowledge of every detail, it is possible to create approximations that work effectively across large datasets. Although some complexities remain hidden, patterns and regularities can still be identified. This is where machine learning becomes valuable, offering practical solutions when traditional methods may not suffice. For instance, artificial neural networks, such as Multi Layer Perceptrons (MLPs), are non-parametric models consisting of multiple layers of nodes, where each node represents a neuron. A Multi

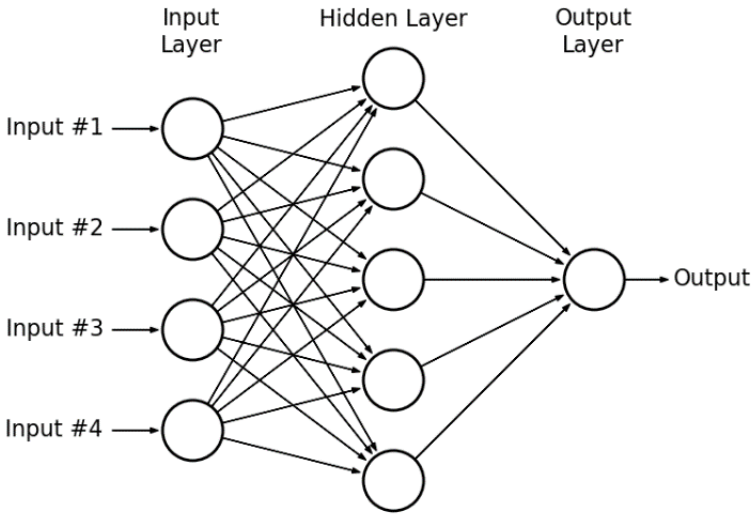

Figure 1.5: A hypothetical example of a MLP network consists of an input layer with multiple feature inputs, one or more hidden layers, and an output layer. Figure taken from Ref. [57].

Layer Perceptron (MLP) typically contains an input layer, one or more hidden layers, and an output layer. Each layer is fully connected to the next, meaning that every neuron in a given layer is connected to every neuron in the following layer. MLPs are powerful tools for both classification and regression tasks, as they can learn complex, non-linear relationships by adjusting the weights of these connections through backpropagation [56].

A neural network (NN) is a series of algorithms that find key patterns in data, working in a way similar to how the human brain processes information. This network is structured into several layers of interconnected nodes, known as neurons, which include input, hidden, and output layers. The input is usually made up of many input features, where a feature is a distinct variable or characteristic of the data being analysed. Each input feature is assigned to a corresponding neuron in the input layer. Neurons within one layer are linked to those in the subsequent layer, each connection characterized by specific weights and biases [56]. In Figure 1.5, an example of a hypothetical MLP is shown, comprised of an input layer, a hidden layer, and an output layer. In any layer of a neural network, the output for layer l is computed as

$$y^{(l)} = f\left(W^{(l)}x^{(l-1)} + b^{(l)}\right). \tag{1.55}$$

In this equation, $x^{(l-1)}$ represents the input to layer l, which is the output from the previous layer $l-1$, or the input data in the case of the first layer. The weight matrix $W^{(l)}$ determines the strength of the connections between neurons from layer $l-1$ to layer l, while the bias vector $b^{(l)}$ adjusts the pre-activation sum for each neuron. The activation function f introduces non linearity by being applied element-wise to the sum of the weighted input and the bias. The output $y^{(l)}$ from layer l serves as the input for the next layer, and this process continues through all layers, including the output layer [56].

The weights and biases are optimised using the *backpropagation* algorithm. Initially introduced by Linnainmaa in the 1970s [58] and later popularized by Rumelhart, Hinton, and Williams in 1986 [59], this algorithm adjusts the weights and biases in response to the observed loss from the output layer. Backpropagation optimises the network by sending the prediction errors from the output layer back through the network, adjusting the weights and biases to minimise the loss function. The loss function quantifies the difference between the predicted and actual values, guiding the network's learning process. During training, the network processes the entire dataset multiple times; each complete pass through the dataset is called an epoch. After each epoch, backpropagation updates the parameters, progressively refining the network's ability to minimise the loss and improve its predictions. The optimisation process involves two main phases:

1. **Forward Pass:** The input features are fed to the initial layer and then propagated through subsequent hidden layers. Each layer applies an activation function, and the process culminates at the output layer.
2. **Backward Pass:** The predicted output of the neural network is compared to the actual target values, which represent the correct outcomes for the given input data. These target values, often referred to as labels in supervised learning, are known from the training dataset. The difference between the predicted output and the true target values is quantified by a loss function, which measures how far off the predictions are from the actual outcomes. The gradient of the loss function is computed with respect to each weight in the network illustrating how a change in each weight would affect the error. This step involves applying the chain rule [58] to effectively propagate the error backward. Subsequently, the weights are adjusted in the direction opposite to the gradient to minimise the loss.

This method allows for the efficient training of neural networks by iteratively reducing the loss, thereby enhancing the model's accuracy over time.

In ML, we calculate the loss function across the entire dataset, which quantifies the difference between the predicted and actual target values. A commonly used loss function is the mean squared error (MSE), defined as

$$\text{MSE} = \mathscr{L}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i(\boldsymbol{\theta}))^2, \tag{1.56}$$

where y_i is the target value, $\hat{y}_i(\boldsymbol{\theta})$ is the predicted value, defined as the output of the last layer in Eq.(1.55), described by the weights and biases grouped together in $\boldsymbol{\theta}$, and n is the number of data points. The goal is to minimise this loss function, reducing the discrepancy between predictions and actual values. When the loss function is differentiable, *gradient descent* is a widely used optimisation method [56]. Gradient descent minimises the loss function, $\mathscr{L}(\boldsymbol{\theta})$, which depends on the model's parameters $\boldsymbol{\theta}$. This is done by iteratively adjusting the parameters in the opposite direction of the gradient, $\nabla_{\boldsymbol{\theta}}\mathscr{L}(\boldsymbol{\theta})$, which points in the direction of the steepest increase in the loss. The size of each adjustment is determined by the learning rate, η. In simple terms, this process can be visualised as rolling down the slope of a valley, with the objective of reaching the lowest point [60]. The learning rate, η, controls how large each step is when updating the parameters. A smaller learning rate makes the updates slower and more precise, reducing the risk of overshooting the optimal value but requiring more iterations. A larger learning rate allows for faster convergence but may lead to instability or missing the optimal solution if the steps are too large. Selecting an appropriate learning rate is critical for the effectiveness of the optimisation process.

A variant of gradient descent, stochastic gradient descent (SGD) [61], approximates the true gradient by using a small subset of the training data, known as a batch, of size n_{batch}. The parameter update at the tth iteration is given by

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \frac{\eta}{n_{\text{batch}}}\sum_{j=1}^{n_{\text{batch}}}\nabla\mathscr{L}(x_j;\boldsymbol{\theta}_t), \tag{1.57}$$

where η is the learning rate, and $\nabla\mathscr{L}(x_j;\boldsymbol{\theta}_t)$ represents the gradient of the loss function with respect to the parameters $\boldsymbol{\theta}_t$, calculated over a batch of data points of size n_{batch}, for x_j input feature parameters.

During training, the dataset is typically shuffled and split into batches. Each batch is processed one by one, where the loss $\mathscr{L}(\boldsymbol{\theta})$ is computed, and the network's parameters are updated accordingly using the gradients. A complete pass through all the batches constitutes an epoch, during which the average loss is computed over all the batches. After each epoch, the model may undergo evaluation on validation data, and the process repeats for

several epochs until the model converges, or the loss function reaches a satisfactory value [62]. This iterative training process allows for more frequent updates to the parameters without having to wait for the entire dataset to be processed, as would be required in standard gradient descent. By using smaller batches in each iteration, SGD is particularly effective for handling large datasets and reduces memory requirements while maintaining reasonable convergence rates.

Several other algorithms exist for optimising gradient descent, with SGD being a foundational method. More advanced optimisers, such as the adaptive moment estimation (Adam) optimiser [63], are also widely used, particularly in large scale ML applications. This optimiser combines the advantages of SGD with adaptive learning rates, which adjusts the learning rate dynamically during the training process, based on the characteristics of the model's parameters or gradients, leading to computational efficiency and lower memory requirements [63]. In this work, the Adam optimiser was employed. The optimisation of the parameters $\boldsymbol{\theta}$ involves iterative updates: for each step, the loss function $\mathscr{L}(\boldsymbol{\theta})$ is computed using a batch of input data. The corresponding gradients of the loss function with respect to each parameter are then determined, allowing for the adjustment of the parameters $\boldsymbol{\theta}$ in the direction that minimises the loss function. The use of adaptive learning rates in Adam further ensures a more stable and efficient optimisation process, particularly when compared to traditional SGD [63].

Using this method of updating the weights and biases, a NN can approximate complex functions that map inputs to outputs, which allows it to generalize across many types of data sets. For example, in a supervised learning task, where each input (such as an image, text, or numerical data) has an associated output (such as a label or target value), the NN learns to predict the correct output for unseen inputs by minimising the error during training. In this case, the data set could consist of labeled images (like recognizing objects in pictures), numerical data (such as predicting house prices from features like square meters), or time series data (like forecasting stock prices). By training on these examples, the NN becomes capable of producing reliable and accurate predictions.

1.7.2 Different types of learning

Depending on the problem at hand, ML algorithms can be broadly classified into three main categories: supervised, unsupervised, and reinforcement learning [62]. These categories are described as follows:

1. **Supervised Learning:** In supervised learning, the algorithm makes predictions based on a given set of examples and corresponding labels. It compares its predictions against the actual outcomes to identify errors and adjust the model accordingly. The input data includes both features and their corresponding labels. This type of learning is typically employed for tasks such as regression or classification.
2. **Unsupervised Learning:** Unsupervised learning involves input data without any corresponding output labels. The primary goal is to model the underlying structure or distribution in the data to discover insights. It achieves this by exploring the data to identify patterns, such as those used in clustering.
3. **Reinforcement Learning:** Reinforcement learning uses agents that learn to make decisions by performing actions and receiving feedback in the form of rewards. The algorithm is not instructed explicitly on which actions to take but instead must determine which actions yield the highest rewards through trial and error [64, 65]. This method is particularly useful in fields such as robotics and gaming.

In addition to these types of learning processes, there are *generative algorithms*, which are often applied in both supervised and unsupervised learning contexts. Generative algorithms are a class of models that aim to learn the underlying data distribution, allowing them to generate new, similar data points. These models are good for tasks like image generation, text synthesis, and probabilistic modeling [66]. These models can be trained in a supervised manner or unsupervised. An exmaple of some generative algorithms are:

- **Generative Adversarial Networks (GANs):** Generative Adversarial Networks (GANs) consist of two neural networks, a generator and a discriminator, competing against each other. The generator attempts to create data that mimics the real dataset, while the discriminator tries to distinguish between real and generated data [67].
- **Variational Autoencoders (VAEs):** VAEs encode data into a latent space and then decode it back, with some variation. They are used to generate new samples from these distributions [68].
- **Normalising Flows (NFs):** NFs are a type of generative model that transform simple distributions into more complex ones using a series of mathematical operations. These models are especially useful for estimating probability distributions and generating new data points [69].

In this thesis, we will focus primarily on supervised learning, which is particularly good for tasks that involve predicting values that may be computationally challenging to derive analytically. Specifically, we will use an MLP in Chapter 2, Sec. 2.6.1, to approximate the SNR calculation. However, the core focus of our work will be on NFs, which we will use to model the probability distributions of cosmological parameters. By training the NF model on GW and EM data from galaxy catalogues, as outlined in Chapter 3, we aim to generate accurate posterior distributions for cosmological parameters.

1.8 Normalising Flows

Normalising Flows (NFs) are a type of generative algorithm in machine learning that transform a simple base distribution into a more complex target distribution by applying a series of invertible functions. This enables NFs to model complex probability distributions with high flexibility, making them useful for tasks like density estimation and generative modeling [69]. The core of how NFs operate is based on the change of variables formula, which maps one probability distribution to another. This is expressed as:

$$p_{\mathbf{x}}(\mathbf{x}) = p_{\mathbf{z}}(\mathbf{z}) \left| \frac{\partial \mathbf{z}}{\partial \mathbf{x}} \right| \tag{1.58}$$

where $p_{\mathbf{x}}(\mathbf{x})$ is the target distribution we wish to approximate, and $p_{\mathbf{z}}(\mathbf{z})$ is a simpler base distribution (often Gaussian) that is easy to sample from, and the random variables $\mathbf{x}$ and $\mathbf{z}$ are vectors with dimensions corresponding to the feature space and the latent space, respectively. The term $|\partial \mathbf{z}/\partial \mathbf{x}|$, known as the *Jacobian* determinant, measures the change in volume between the two distributions, ensuring the preservation of total probability.

The NF models apply a sequence of transformations, each of which is invertible and differentiable. Each transformation is parameterized by a NN, and the invertibility of these transformations allows for both efficient sampling and exact computation of the probability density at each step. This makes NFs particularly powerful for probabilistic modeling, where generating new data points or estimating likelihoods is of interest. This is done by repeatedly applying the change of variables formula, allowing the initial density to "flow" through the sequence of transformations [70]. The utility of NFs has expanded considerably, with various types of mappings [69, 71] being developed for different applications, including GW astronomy [37, 38, 40].

To formalise this, we begin with Eq.(1.58), under a conditional statement $\boldsymbol{\omega}$, we can expand by taking the natural logarithm on both sides. Assuming that the random variables x and z are related as $x = f(z)$ and $z = g(x)$, where $f \equiv g^{-1}$, we arrive at:

$$\log(p_{\mathbf{x}}(\mathbf{x}|\boldsymbol{\omega})) = \log(p_{\mathbf{z}}(f^{-1}(\mathbf{x}, \boldsymbol{\theta}|\boldsymbol{\omega}))) + \log\left(\det\left|\frac{df^{-1}(\mathbf{x}, \boldsymbol{\theta}|\boldsymbol{\omega})}{d\mathbf{x}}\right|\right), \tag{1.59}$$

where $\boldsymbol{\theta}$ represents the "trainable "parameters optimised during the training of the NF. The latent distribution is often chosen as a simple distribution, such as a Gaussian, because it is easy to sample from and has a tractable PDF, $p_{\mathbf{z}}(z)$. From this, two important features of NFs arise:

1. The PDF of the target distribution, $p_{\mathbf{x}}(\mathbf{x})$, becomes tractable.
2. The target distribution can be efficiently sampled by first drawing samples from the latent distribution and then applying the inverse mapping, $x = f(z, \boldsymbol{\theta})$.

These attributes of NFs are only possible if the mapping $x = f(z, \boldsymbol{\theta})$ is invertible (i.e. $z = g(x)$ can be computed) and the Jacobian determinant of the transformation is tractable. The Jacobian ensures that the volume change between the latent space and the target space is correctly accounted for, enabling probability computation and efficient sampling [70].

1.8.1 Loss function for Normalising Flows

To optimise the parameters of the mapping function, $\boldsymbol{\theta}$, a loss function is defined between the output and input target distributions. In the case of NFs, the KL divergence is used as a loss function. The origins of the KL are from information theory [72]. To quantify the amount of information in a dataset with a particular distribution, we use its entropy, defined as

$$H = -\sum_{i=0}^{N} p(x_i) \log p(x_i), \tag{1.60}$$

where $\log$ refers to the natural logarithm (base e). If $\log_2$ is used instead, entropy is measured in *bits*, whereas with the natural logarithm, it is measured in *nats*. The term H is the *entropy*, the minimum number of bits required to encode the information [72]. The KL divergence measures how one probability distribution q diverges from a target distribution p. It quantifies the difference between the two distributions by comparing the logarithms of their probabilities, weighted by the probability of the target distribution p.

For a discrete set of random variables x_i [72], this can be expressed as

$$\mathrm{KL}(p||q) = \sum_{i=0}^{N} p_x(x_i) \log\left(\frac{p_x(x_i)}{q_x(x_i)}\right). \tag{1.61}$$

In this formulation, $p_x(x_i)$ is the probability of the i-th sample under the target distribution, and $q_x(x_i)$ is its corresponding probability under the approximating distribution. The KL divergence is non-negative and equals zero only when the two distributions are identical, making it a useful metric for comparing the similarity between probability distributions. Then, for continuous random variables, we replace the summation with an integral

$$\mathrm{KL}(p_x||q_x) = \int_{-\infty}^{+\infty} p_x(x) \log\left(\frac{p_x(x)}{q_x(x)}\right) dx. \tag{1.62}$$

For a set of data samples x drawn from the target distribution $p_x(x)$, and an approximating distribution $q_x(x)$ modeled by the NF, the loss function is defined as the KL divergence between the two distributions:

$$\begin{aligned}
\mathrm{Loss}(\boldsymbol{\theta}) = \mathscr{L}(\boldsymbol{\theta}) &= \mathrm{KL}(p_x(x)||q_x(x)) \\
&= \mathbb{E}_{p_x}[\log p_x(x)] - \mathbb{E}_{p_x}[\log q_x(x)] \\
&= \mathbb{E}_{p_x}[\log p_x(x)] - \mathbb{E}_{p_x}\left[\log p_z(z) + \log\left(\det\left|\frac{df^{-1}(\mathbf{x},\boldsymbol{\theta}|\boldsymbol{\omega})}{d\mathbf{x}}\right|\right)\right] \\
&= \text{const.} - \mathbb{E}_{p_x}\left[\log p_z(z) + \log\left(\det\left|\frac{df^{-1}(\mathbf{x},\boldsymbol{\theta}|\boldsymbol{\omega})}{d\mathbf{x}}\right|\right)\right].
\end{aligned} \tag{1.63}$$

Since the first term is constant for a given dataset, the expectation value can be approximated as a discrete average for a set of K data points $\{x_i\}_{i=1}^{K}$, yielding:

$$\mathscr{L}(\boldsymbol{\theta}) \approx -\frac{1}{K}\sum_{i=0}^{K}\left[\log p_z(f^{-1}(x_i,\boldsymbol{\theta}|\boldsymbol{\omega})) + \log\left|\frac{df^{-1}(x_i,\boldsymbol{\theta}|\boldsymbol{\omega})}{dx}\right|\right]. \tag{1.64}$$

We can approximate the target distribution by optimising the parameters of a NN, denoted as $\boldsymbol{\theta}$, using standard optimisation algorithms such as gradient descent. Importantly, the presence of a conditional variable, $\boldsymbol{\omega}$, does not affect this process. The mapping remains invertible, and the Jacobian determinant remains tractable, ensuring that the overall transformation is efficient and valid, as outlined in Ref. [70]. With the loss function defined in Eq.(1.64), we can begin training the NF by iteratively adjusting the parameters $\boldsymbol{\theta}$ through gradient descent, minimising the loss function to improve the accuracy of the approximation.

1.8.2 Coupling Flows

In the implementation of a NF, certain constraints must be satisfied regarding the types of functions used. The function that maps $f : x \to z$ (from input data space to latent space) must be invertible, and the computation of the Jacobian determinant needs to be tractable [69]. Three primary methods are commonly employed in constructing NFs to meet these criteria: coupling flows [73], autoregressive flows [74], and spline flows [75].

Coupling flows introduced by [73] for NFs divide the input data into two subspaces. Consider partitioning the input data x into two subsets: $x^{1:d}$ and $x^{d:D}$. Here, $x^{1:d}$ represents data dimensions up to d, and $x^{d:D}$ spans from dimension d (excluded) to D. To create a bijective function, one that matches each input to exactly one unique output (and vice versa), while keeping it flexible and easy to compute, simple functions called affine coupling layers are used. These layers modify part of the input step by step to achieve the desired transformation. The change is easy to reverse, and it depends on the unmodified part of the input, $x^{1:d}$. For a D-dimensional input x with $d < D$, the output y of an affine coupling layer is determined by the following equations [73, 76]:

$$\begin{aligned} y_{1:d} &= x_{1:d} \\ y_{d+1:D} &= x_{d+1:D} \odot \exp\left(s(x_{1:d}, \theta)\right) + t(x_{1:d}, \theta) \end{aligned} \tag{1.65}$$

where s and t are scale and translational functions parameterised by trainable parameters θ, and $\odot$ is the element-wise product. The reason for the construction of the bijective functions Eq.(1.65) is so that the Jacobian is guaranteed to be invertible and tractable. In coupling layers, this is achieved by designing the transformation in a block-wise manner, making the Jacobian matrix easier to compute. Specifically, the Jacobian determinant can be calculated analytically as

$$\det\left|\frac{\partial y(x;\theta)}{\partial x}\right| = \begin{vmatrix} \frac{\partial y_1}{\partial x_1} & \cdots & \frac{\partial y_1}{\partial x_D} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_D}{\partial x_1} & \cdots & \frac{\partial y_D}{\partial x_D} \end{vmatrix} = \begin{vmatrix} \mathscr{I}_{1:d} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ \frac{\partial y_{d+1:D}}{\partial x_{1:\mathrm{d}}} & \cdots & \mathrm{diag}\left(\exp\left[s\left(x_{1:\mathrm{d}}, \theta\right)\right]\right) \end{vmatrix} \tag{1.66}$$

where $\mathscr{I}_{1:\mathrm{d}}$ represents the identity matrix of size $d \times d$, and $\mathrm{diag}\left(\exp\left[s\left(x_{1:\mathrm{d}}, \theta\right)\right]\right)$ refers to a diagonal matrix containing the exponentiated values of the scale function $s(x, \theta)$.

Since the calculation of the Jacobian determinant in the coupling layer does not require evaluating the full Jacobian of the functions s or t, these functions can be as complex as necessary. Although $s(x, \theta)$ appears in the transformation, we do not need to compute the full Jacobian (i.e., the derivatives of all elements of s). Instead, only the value of

$s(x,\theta)$ is needed to compute the determinant. This is because the Jacobian matrix has a structure where only the diagonal elements (associated with s) contribute to the determinant. Therefore, s and t can be modeled as deep neural networks, parameterized by θ, allowing for highly flexible and expressive transformations without incurring significant computational overhead [76].

1.8.3 Spline Flows

Another implementation of NFs is *spline flows*. Work described in Ref. [77], based on Ref. [75], shows that this development significantly amplifies the adaptability of coupling and autoregressive transformations while maintaining the property of analytic invertibility. This methodology revealed that the implementation of neural spline flows remarkably enhances the accuracy of density estimation [75, 77].

The main idea behind spline flows is the construction of an affine mapping function, such as $y = g_\theta(x)$ using monotonic rational-quadratic splines, where, parameterised by trainable parameters θ_i. A spline is defined by a series of points called *knots*, which divide the input space into bins. Each bin between knots is characterized by a monotonic rational-quadratic function, which is a ratio of two quadratic polynomials. The knots serve as boundaries for these intervals, allowing the spline to smoothly and flexibly map the data by adjusting the function within each defined segment [78]. These functions are straightforward to differentiate and analytically invertible [77]. Using notation from [77], lets assume the input data x is defined between $[-B,B]$. For values outside this interval, the transformation defaults to the identity function, introducing linear tails so that the overall transformation can handle unconstrained inputs.

The construction then involves K distinct rational-quadratic functions, defined by $K+1$ coordinates, $\{(x^{(k)},y^{(k)})\}_{k=0}^{K}$, referred to as knots. These knots are arranged monotonically from $(x^{(0)},y^{(0)}) = (-B,-B)$ to $(x^{(K)},y^{(K)}) = (B,B)$. To define the spline's behavior further, [77] assigns $K-1$ positive values, $\{\delta^{(k)}\}_{k=1}^{K-1}$, to represent the derivatives at the internal knots, ensuring smooth transitions between segments. The derivatives at the boundary knots, $\delta^{(0)}$ and $\delta^{(K)}$, are set to 1 to match the gradients of the linear tails. Figure 1.6 illustrates this setup [77].

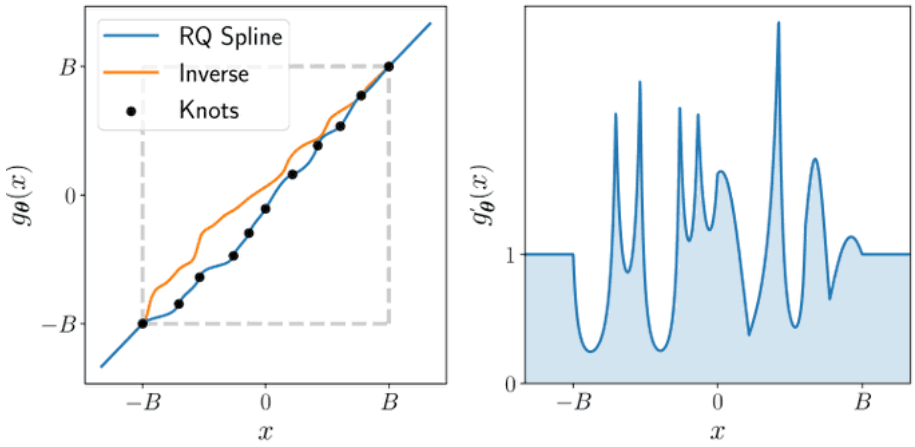

Figure 1.6: (Left): random monotonic rational-quadratic transform with $K = 10$ bins and linear tails, defined by $K+1$ knots. Each interval between knots is described by a rational-quadratic function. (Right): The derivative of the transform on the left with respect to x, illustrating the smooth and monotonic nature of the transformation. Image credit [77]

The slope between adjacent points (x_k, y_k) and (x_{k+1}, y_{k+1}) is defined as

$$s_k = \frac{y_{k+1} - y_k}{x_{k+1} - x_k}, \tag{1.67}$$

and the position of x between two knots is normalised as

$$\zeta(x) = \frac{x - x_k}{x_{k+1} - x_k}. \tag{1.68}$$

The rational-quadratic spline for the k-th segment is

$$y = \frac{\alpha^{(k)}(\zeta)}{\beta^{(k)}(\zeta)} = y^{(k)} + \frac{\left(y^{(k+1)} - y^{(k)}\right)\left[s^{(k)}\zeta^2 + \delta^{(k)}\zeta(1-\zeta)\right]}{s^{(k)} + \left(\delta^{(k+1)} + \delta^{(k)} - 2s^{(k)}\right)\zeta(1-\zeta)}, \tag{1.69}$$

where $\delta^{(k)}$ and $\delta^{(k+1)}$ are the derivatives at the k and $k+1$ knots, defining the shape of the mapping spline function. Since the rational-quadratic transformation is monotonic and applied element by element to the input vector, the Jacobian determinant can be calculated by taking the derivative of the transformation for each input value. The logarithm of the determinant is then the sum of the logarithms of these individual derivatives [77]. The derivative of the rational-quadratic transformation with respect to x is given by:

$$\frac{d}{dx}\left(\frac{\alpha^{(k)}(\zeta)}{\beta^{(k)}(\zeta)}\right) = \frac{s^{(k)}\left[2\delta^{(k+1)}\zeta^2 + 2s^{(k)}\zeta(1-\zeta) + \delta^{(k)}(1-\zeta)^2\right]}{\left(s^{(k)} + \left(\delta^{(k+1)} + \delta^{(k)} - 2s^{(k)}\right)\zeta(1-\zeta)\right)^2}. \tag{1.70}$$

To invert the rational-quadratic transformation, we solve for $\zeta(x)$ by solving a quadratic equation derived from Eq.(1.69), as

$$a\zeta^2 + b\zeta + c = 0, \tag{1.71}$$

where the coefficients a, b, and c are given by

$$a = (y^{(k+1)} - y^{(k)}) \left(s^{(k)} - \delta^{(k)} \right) + (y - y^{(k)}) \left(\delta^{(k+1)} + \delta^{(k)} - 2s^{(k)} \right), \tag{1.72}$$

$$b = (y^{(k+1)} - y^{(k)}) \delta^{(k)} - (y - y^{(k)}) \left(\delta^{(k+1)} + \delta^{(k)} - 2s^{(k)} \right), \tag{1.73}$$

$$c = -s^{(k)} (y - y^{(k)}). \tag{1.74}$$

The quadratic formula then gives two potential solutions for $\zeta(x)$

$$\zeta(x) = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}. \tag{1.75}$$

Since the rational-quadratic spline is monotonic, we can always determine the correct root, ensuring that the inverse transformation is well-defined, and therefore find solutions for x [77].

The rational-quadratic spline is applied elementwise to each component of the input vector, allowing for flexible transformations while ensuring computational efficiency. Since the transformation is performed elementwise, the log-determinant of the Jacobian can be computed as a sum over the transformed components. This makes the model efficient at both transforming and inverting data distributions, which is especially useful for tasks such as density estimation and generative modeling.

1.9 Conclusion

In this chapter, we covered a range of key topics that connect cosmology, GW, and ML. We began by addressing the Hubble tension, the discrepancy between different measurements of the universe's expansion rate, which is a central challenge in modern cosmology. To provide context, we discussed how the universe is expanding and why this tension arises from inconsistencies in the observed expansion rate using different methods.

From there, we explored GWs, their theoretical background in general relativity, how they are detected by observatories like LIGO and Virgo, and the challenges posed by noise, along with techniques like matched filtering to detect GW signals.

We then introduced Bayesian statistics, which provides a powerful framework for inferring parameters from observed data. This approach is especially useful for studying GWs, as it allows us to estimate the parameters describing the GW source, such as mass and distance. From these parameters, we can measure the calibrated distance to the event, which, in turn, enables cosmological measurements without relying on the traditional cosmic distance ladder. However, due to the degeneracy between redshift and mass, it is impossible to measure the Hubble constant solely using GW observations. To resolve this, galaxy catalogues must be used with GW data, a technique proposed by Schutz [12].

We then provided a brief literature review on the advancements in cosmology through GW observations, beginning with the foundational proposal by Schutz, who first suggested the use of GWs to measure the Hubble constant with galaxy catalogues. This was followed by the contributions of Finn and Chernoff, and later by Walter Del Pozzo. Samaya Nissanke, Messenger and Read continued this progress, with further developments from Taylor and Gair. These efforts ultimately culminated in the first direct GW measurements, where pipelines such as `gwcosmo` and `IcaroGW` were employed to estimate cosmological and population parameters. These developments laid the foundation for using GWs as a powerful tool to extract valuable cosmological insights.

We then explored how Bayesian methods are applied to address the challenges of inferring cosmological parameters from GW observations. In particular, we focused on `gwcosmo`, a key tool for estimating cosmological and population parameters by combining GW data with galaxy catalogues. We also present `CosmoFlow`, which builds on the assumptions made in `gwcosmo` but exploits ML techniques to predict cosmological parameters, which is the main topic of this thesis.

In the ML section, we introduced the different types of learning, supervised, unsupervised, and reinforcement and their applications in processing large datasets. We then focused on NFs, an ML technique designed for modeling complex probability distributions efficiently. We explored how NFs operate, emphasizing the role of coupling and spline flows, which make these models both powerful and computationally feasible. These techniques will play a central role in the methodologies of `CosmoFlow` in this analysis.

Chapter 2

CosmoFlow: data generation

Declaration: the majority of the material in this chapter is based on the methodology of my first-author paper [54].

2.1 Introduction: What is CosmoFlow and why is it important?

As highlighted in the introductory Chapter 1, the use of GW for inferring cosmological parameters has advanced significantly since the first detection in 2015 [30]. A major breakthrough occurred in 2017 with the detection of a BNS merger event [49]. This event, notable for its multimessenger nature, where both EM and GW signals were observed, provided one of the most precise cosmological measurements to date using GW events. Subsequent detections of other GW events have continued to contribute incrementally to our understanding of cosmology. This progress has involved using a Bayesian approach to estimate the posterior distributions of cosmological parameters by leveraging GW posterior data samples, particularly the luminosity distance, in combination with EM data such as galaxy surveys.

Current methods, while effective at estimating cosmological parameters, rely heavily on the computational speed of evaluating complicated integrals. This becomes increasingly inefficient as the number of GW events rises due to enhanced detector sensitivity. Additionally, these pipelines lack the flexibility to expand the analysis to multiple cosmological parameters, which will become important as more distant events are detected, where parameters like the dimensionless density parameters have greater influence on the luminosity distance estimation.

There is a clear need for faster and more flexible pipelines to manage the growing number of detectable events by the LIGO/VIRGO and future KAGRA detectors. This requires much quicker computation of likelihoods to estimate cosmological parameters efficiently.

This chapter introduces `CosmoFlow`, a machine learning tool designed for the estimation of cosmological parameters through the application of simulation-based inference. Simulation-based inference methods address complex problems by using simulations to approximate the likelihood function, which can then be incorporated into traditional Bayesian frameworks to infer the posterior distribution. This likelihood-based approach is particularly useful when the likelihood is difficult to calculate analytically but still feasible to estimate numerically or approximate with computational methods, making it tractable. Alternatively, likelihood-free inference techniques directly estimate the posterior distribution without the need to compute the likelihood explicitly. Methods like Approximate Bayesian Computation (ABC) [79] generate simulated data and compare it to observed data, adjusting the model parameters to achieve a close match. Likelihood-free methods are especially valuable in situations where calculating the likelihood is impractical or impossible [80]. In the case of `CosmoFlow`, we use the likelihood-based approach, where we approximate parts of the likelihood function to estimate the posterior distribution of cosmological parameters using NFs.

In simpler terms, our objective is to generate synthetic data of GW events under different cosmological scenarios, representing potential GW events from BBHs observable under various cosmological parameters. We aim to identify which events can be detected above a certain SNR threshold to account for GW selection effects. The SNR measures the strength of the GW signal relative to background noise, with higher SNRs indicating more reliable detections and accurately measured parameters. Events with lower SNRs are more susceptible to noise, leading to uncertainties and biases in parameter estimation. By setting an SNR threshold, both `gwcosmo` and `CosmoFlow` include only high-confidence events, improving the reliability and accuracy of inferred cosmological parameters.

This chapter focuses on three major aspects of the `CosmoFlow` analysis: the theoretical Bayesian approach to estimate the posterior distribution from GW and EM data; the generation of GW event data and integration of galaxy surveys, for which we use GLADE+ [81]; and finally, the training of the NF within the data-generation process.

2.2 CosmoFlow: Bayesian Framework

In this section, we will define the Bayesian framework developed for the analysis of `CosmoFlow` [82]. In developing this framework, we aimed to remain as consistent as possible with the established Bayesian framework of `gwcosmo` [14, 51, 52, 83]. This consistency allows us to compare the methods more directly and, if needed, change `CosmoFlow`'s underlying methodologies to align more closely with `gwcosmo`.

We start by defining the posterior distribution of the cosmological parameters, Ω, conditioned by the GW strain data with n events, $\mathbf{h} = [h_1, h_2, \ldots, h_n]$ as

$$p(\Omega|\mathbf{h},\mathbf{D},I) = p(\Omega|I)\prod_i \frac{p(h_i,D_i|\Omega,I)}{p(h_i,D_i|I)}, \tag{2.1}$$

where $\mathbf{D} = [D_1, D_2, \ldots, D_n]$ is the binary state of detection, 1 for detected and 0 for not detected for the ith event, and I represents all other assumed information. The detectability of a GW signal accounts for the selection effects that arise when applying SNR thresholds on candidate events [84, 85]. It is important to ensure the likelihood term is properly normalised, such that

$$\sum_{D_i=0,1} \int p(h_i,D_i|\Omega,I)dh_i = 1. \tag{2.2}$$

This normalisation allows us to properly evaluate the likelihood over all detectable and non-detectable events, marginalising over all events. This ensures that the analysis covers the entirety of the GW parameter space. However, encoded within I is the information that only events exhibiting an SNR greater than some threshold ρ_{th} are considered.

To elaborate further, although we have measured strain data for all times, not just for detected events, our analysis focuses on the subset of events that exceed the detection threshold. This implies that for detected events, we only consider the data where $D_i = 1$. This ensures that our likelihood function is conditioned on the event being detectable, effectively normalising the likelihood to the subset of detectable events. This is why we can

simplify $p(h_i|\Omega, D_i = 1, I)$ to $p(h_i|\Omega, I)$, as $D_i = 1$ by definition for these cases. Therefore, by only considering detectable events (i.e., $\rho \geq \rho_{\rm th}$), we find that

$$p(h_i, D_i|\Omega, I) = \frac{p(h_i|\Omega, I)}{p(D_i|\Omega, I)} \tag{2.3}$$

where $p(h_i|\Omega, D_i, I) \equiv p(h_i|\Omega, I)$. This equivalence holds because, for detected strain data h_i, the event is certainly detected by definition.

Therefore, the normalisation term $p(D_i|\Omega, I)$ ensures that we correctly account for the selection bias introduced by only considering events above the $\rho_{\rm th}$ threshold, allowing us to properly evaluate the likelihood over all detectable events and marginalise appropriately. This approach ensures that the analysis encompasses the entirety of the GW parameter space relevant to detectable events. This result allows us to write our cosmological parameter posterior as

$$p(\Omega|\mathbf{h}, \mathbf{D}, I) = p(\Omega|I) \prod_i \frac{p(h_i|\Omega, I)}{p(h_i, D_i|I) p(D_i|\Omega, I)}. \tag{2.4}$$

Then the numerator term can be expanded by marginalising over the GW detector-frame parameters ($\theta_{\rm GW}$) as follows

$$p(\Omega|\mathbf{h}, \mathbf{D}, I) = p(\Omega|I) \prod_i \frac{\int p(h_i|\theta_{i,\rm GW}, I)\, p(\theta_{i,\rm GW}|\Omega, I)\, d\theta_{i,\rm GW}}{p(h_i, D_i|I)\, p(D_i|\Omega, I)}. \tag{2.5}$$

We continue our derivation by focusing on expanding the second term in the numerator within the integrand, namely $p(\theta_{i,\rm GW}|\Omega, I)$, which describes the prior distribution of the GW parameters conditioned on the cosmological parameters.

This term is important because it accounts for the distribution of GW parameters given our cosmological model and prior information. Conditioning the GW parameters on the detectability of the event is essential to account for selection effects. Without this consideration, our analysis could be biased by the fact that some events are more easily detectable than others, which could skew the results. By incorporating detectability into the prior, we ensure that our analysis accurately reflects the underlying population of GW events and avoids biases introduced by selection effects.

This can be achieved by applying Bayes' rule, as defined in Eq. (1.33), to the term $p(\theta|\Omega,D,I)$ as follows:

$$p(\theta_{i,\mathrm{GW}}|\Omega,D_i,I) = \frac{p(D_i|\theta_{i,\mathrm{GW}},I)p(\theta_{i,\mathrm{GW}}|\Omega,I)}{p(D_i|\Omega,I)}. \tag{2.6}$$

We note that the detectability of an event is unaffected by Ω once the GW parameters are specified in the detector frame, i.e., $p(D_i|\theta_{i,\mathrm{GW}}) \equiv p(D_i|\Omega,\theta_{i,\mathrm{GW}},I)$. This holds true because $\theta_{i,\mathrm{GW}}$ are the detector frame parameters. In the detector frame, parameters such as mass are already redshifted, and the luminosity distance is fixed for a given set of GW parameters. Therefore, additional cosmological information does not influence the detectability further.

We can define the probability of detection for a single detector, given the matched-filtering SNR ρ, which depends on the GW parameters θ_{GW}, as:

$$p(D|\theta_{\mathrm{GW}},I) = \begin{cases} 1, & \text{if } \rho(\theta_{\mathrm{GW}}) > \rho_{\mathrm{th}}, \\ 0, & \text{otherwise.} \end{cases} \tag{2.7}$$

By rearranging Eq. (2.6), we can solve for $p(\theta_{i,\mathrm{GW}}|\Omega,I)$, and subsequently substitute this expression into Eq. (2.5). After some mathematical manipulation, we obtain:

$$\begin{aligned} p(\Omega|\mathbf{h},\mathbf{D},I) &= p(\Omega|I)\prod_i \int \frac{p(h_i|\theta_{i,\mathrm{GW}},I)\,p(\theta_{i,\mathrm{GW}}|\Omega,D_i,I)\,\cancel{p(D_i|\Omega,I)}}{p(h_i,D_i|I)\,\cancel{p(D_i|\Omega,I)}\,p(D_i|\theta_{i,\mathrm{GW}},I)}\,d\theta_{i,\mathrm{GW}} && \text{(a)} \\ &= p(\Omega|I)\prod_i \int \frac{p(\theta_{i,\mathrm{GW}}|h_i,I)\,p(h_i|I)\,p(\theta_{i,\mathrm{GW}}|\Omega,D_i,I)}{p(h_i,D_i|I)\,p(D_i|\theta_{i,\mathrm{GW}},I)\,p(\theta_{i,\mathrm{GW}})}\,d\theta_{i,\mathrm{GW}} && \text{(b)} \\ &= p(\Omega|I)\prod_i \frac{p(h_i|I)}{p(h_i,D_i|I)} \int \frac{p(\theta_{i,\mathrm{GW}}|h_i,I)\,p(\theta_{i,\mathrm{GW}}|\Omega,D_i,I)}{p(D_i|\theta_{i,\mathrm{GW}},I)\,p(\theta_{i,\mathrm{GW}})}\,d\theta_{i,\mathrm{GW}}. && (2.8) \end{aligned}$$

In these series of equations, we have considered the posterior distribution of the GW parameters, $p(\theta_{i,\mathrm{GW}}|h_i,I)$. These parameters reflect the probabilistic model of GW parameters conditioned on the observed strain data h_i from each detected GW event, as detailed in [86] and [87]. Then, by only considering terms that depend on the variation of the cosmological parameters, and noting that the prior on the GW parameters is conditioned on the cosmological model and parameters used, we can write $p(\theta_{i,\mathrm{GW}}|I) \equiv p(\theta_{i,\mathrm{GW}}|\Omega_0,I)$, where $p(\theta_{i,\mathrm{GW}}|\Omega_0,I)$ represents the GW parameter priors used in the parameter estimation assuming a fixed cosmology Ω_0 [86, 87]. This follows the proportionality relationship:

$$p(\Omega|\mathbf{h},\mathbf{D},I) \propto p(\Omega|I)\prod_i \int \frac{p(\theta_{i,\mathrm{GW}}|h_i,I)p(\theta_{i,\mathrm{GW}}|D_i,\Omega,I)}{p(D_i|\theta_{i,\mathrm{GW}},I)p(\theta_{i,\mathrm{GW}}|\Omega_0,I)}\,d\theta_{i,\mathrm{GW}}, \tag{2.9}$$

where we have omitted the term outside the main integral, as it is not a function of the cosmological parameters and will thus be eliminated during the normalisation process. From the definition of the expected value of a PDF of a random variable $g(x)$ sampled from a continuous distribution $f(x)$, the expected value is given by:

$$\mathscr{E}[g(x)] = \int_{-\infty}^{\infty} g(x) f(x)\, dx. \tag{2.10}$$

If we only have access to a set of samples from the distribution $f(x)$, the expected value can be approximated using these samples. This approximation is given by:

$$\mathscr{E}[g(x)] \approx \frac{1}{N} \sum_{i=1}^{N} g(x_i) = \langle g(x) \rangle_{x \sim f(x)}, \tag{2.11}$$

where $\{x_i\}_{i=1}^{N}$ are the sampled values from $f(x)$, and $\langle \cdot \rangle$ denotes the expectation value or average over the distribution $f(x)$. This approximation becomes increasingly accurate with a larger number of samples and is particularly useful in Monte Carlo simulations, where the expected value is computed by averaging over these samples.

By examining Eq. (2.9), we can use the definition of the expectation value described in Eq. (2.11) to compute the necessary integrals via Monte Carlo sampling. This ensures accurate estimations of the posterior distributions. For a fixed number of sampled GW parameters from the posterior defined as $p(\boldsymbol{\theta}|h,I)$, it is possible to approximate the integral over the GW parameters using a Monte Carlo summation. This approximation results in the following expression:

$$p(\Omega|\mathbf{h},\mathbf{D},I) \propto p(\Omega|I) \prod_i \left\langle \frac{p(\boldsymbol{\theta}_{i,\mathrm{GW}}|\Omega,D_i,I)}{p(D_i|\boldsymbol{\theta}_{i,\mathrm{GW}},I)\, p(\boldsymbol{\theta}_{i,\mathrm{GW}}|\Omega_0,I)} \right\rangle_{\boldsymbol{\theta}_{i,\mathrm{GW}} \sim p(\boldsymbol{\theta}_{i,\mathrm{GW}}|h_i,I)}. \tag{2.12}$$

The equation shows that the expectation is calculated over the GW parameters sampled from the posterior distribution $p(\boldsymbol{\theta}_{i,\mathrm{GW}}|h_i,I)$.

In Eq. (2.12), the numerator includes the probability density function of the GW parameters given the detection status of the ith event, D_i, the cosmological parameters $\boldsymbol{\Omega}$, and prior information I. This term is important to our analysis as it describes the prior distribution of the GW parameters conditioned on the cosmological model $\boldsymbol{\Omega}$ and the detectability of the event. Within the prior distribution of the parameters, EM data will be included through the use of galaxy catalogues, specifically GLADE+ [81].

The denominator contains two terms: 1) $p(D_i|\theta_{i,\mathrm{GW}},I)$, the probability mass of detecting an event given the GW parameters derived from the posterior samples for the ith event. 2) $p(\theta_i|\Omega_0,I)$, the prior distribution of GW parameters used in computing the posterior samples for the ith event, which is influenced by the baseline cosmological model Ω_0 and prior information I.

2.2.1 Probability of detection

The probability of detection, $p(D|\theta_{\mathrm{GW}},I)$, as shown in the denominator of Eq. (2.12), serves as a weighting factor for each posterior distribution sample associated with every event. Each event in the GWTC-2 [86] and GWTC-3 [88] GW catalogues consists of a set of posterior samples representing the inferred physical parameters of the detected GW events. These samples are weighted according to their detectability when used in Eq. (2.12).

The matched-filter SNR of an event, ρ, is compared against a threshold ρ_{th} to determine detectability. This approach follows the analysis presented in [16]. For the matched-filtering SNR of a detector network, we assume that different detectors' data can be combined in quadrature

$$\rho_{\mathrm{net}} \equiv \sqrt{\sum_i \rho_i^2}, \tag{2.13}$$

where ρ_i is the matched-filtering SNR of the ith detector, as described in Eq. (1.28), in the network. This formula is valid when the noise of each detector is statistically independent [89].

Due to the coincident analysis of multiple detectors, the noise-dominated matched-filter SNR of the detector network can be described using a χ^2 distribution. This arises from the central limit theorem and the typically Gaussian nature of noise in GW detectors.

When no signal is present, the sum of squared independent Gaussian noise variables follows a central χ^2 distribution. This distribution arises because Gaussian noise in each detector contributes to a squared sum, and in the absence of a signal, the expected value of each squared term remains centered around zero.

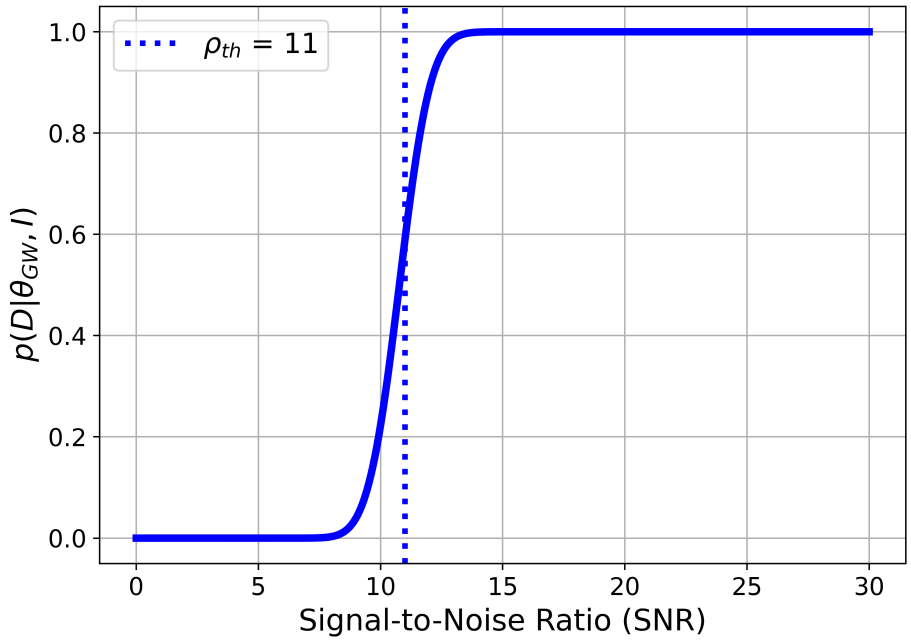

Figure 2.1: Survival function of the χ^2 distribution, as detailed in Eq. (2.14), with $\rho_{\rm th} = 11$.

When a signal is present, however, the distribution changes to a non-central χ^2 distribution. In this case, the SNR follows a non-central χ^2 distribution with a non-centrality parameter $Q(\theta_{\rm GW}) = \rho^2$, where ρ is the matched-filter SNR. The non-centrality parameter $Q(\theta_{\rm GW})$ represents the contribution from the detected GW signal, which depends on the GW parameters of the event, denoted by $\theta_{\rm GW}$. Essentially, when a signal is present, the SNR has an offset (given by $Q(\theta_{\rm GW})$), which indicates that the distribution is shifted away from zero, reflecting the presence of an actual GW signal. This is expressed as follows:

$$p(D|\theta_{\rm GW}, I) = \int_{Q>\rho^2_{\rm th}}^{\infty} \chi^2(Q(\theta_{\rm GW}), k = 2n)\, dQ, \tag{2.14}$$

where $k = 2n$ represents the degrees of freedom, accounting for n detectors and doubled to include both the real and imaginary parts of the strain data. The strain is represented as a complex quantity because it includes both amplitude and phase information of the GW signal. This integral function describes the survival function of the non central χ^2 distribution, and can be easily evaluated using the built-in function in `scipy.stats.ncx2.sf` [90]. In Fig. 2.1 the probability distribution of detection, following a non-central χ^2 distribution, is plotted over different values of combined detector matched-filter SNR. In this case, the $\rho_{\rm th} = 11$ was set as the detector threshold SNR, which is the same value used in [16].

In this framework, highly detectable samples (with $\rho(\theta) \gg \rho_{\rm th}$) will have $p(D|\theta,I) \approx 1$, meaning they contribute directly to the posterior without additional weighting. Conversely, samples with $\rho(\theta) \approx \rho_{\rm th}$ will have $p(D|\theta_{\rm GW},I) \approx 0$, meaning they are less likely to be detected and thus are up-weighted in the analysis to account for their lower detectability. This weighting ensures a balanced contribution from all samples.

2.2.2 Prior distribution

The prior distributions for GW parameters, represented by $p(\theta_{\rm GW}|\Omega_0,I)$, define the expected range of values for these parameters, given the cosmological model Ω_0 and any relevant prior information I. Among the various GW parameters, the luminosity distance $D_{\rm L}$ plays a pivotal role in cosmological analysis.

In a flat universe (where $\Omega_k = 0$), the luminosity distance $D_{\rm L}$ is related to the comoving distance D_C through the redshift factor $(1+z)$. Specifically, the luminosity distance accounts for both the expansion of the universe and the increase in distance due to the redshift, which can be expressed as:

$$D_{\rm L} = (1+z)D_C, \tag{2.15}$$

where D_C is the comoving distance. In a more detailed form, the comoving distance D_C can be written as:

$$D_C = \frac{c}{H_0}\int_0^z \frac{dz'}{E(z')}, \tag{2.16}$$

where $E(z)$ describes the evolution of the expansion rate of the universe and depends on cosmological parameters, such as the matter density parameter $\Omega_{\rm m}$, curvature density parameter Ω_k, and dark energy density parameter Ω_Λ:

$$E(z) = \sqrt{\Omega_{\rm m}(1+z)^3 + \Omega_k(1+z)^2 + \Omega_\Lambda}. \tag{2.17}$$

In the case of a flat universe ($\Omega_k = 0$), this simplifies to:

$$E(z) = \sqrt{\Omega_{\rm m}(1+z)^3 + \Omega_\Lambda}. \tag{2.18}$$

Thus, the luminosity distance can be expressed as:

$$D_{\rm L} = \frac{c(1+z)}{H_0}\int_0^z \frac{dz'}{E(z')}. \tag{2.19}$$

This relationship highlights how the luminosity distance D_{L} depends directly on the redshift z, the Hubble constant H_0, and the evolution of the expansion rate $E(z)$, which is determined by the cosmological parameters [1].

In the GW events listed in GWTC-3, a prior uniform in comoving volume was used, assuming a constant merger rate in the source's comoving frame with $H_0 = 67.9\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$ and $\Omega_{\mathrm{m}} = 0.3065$ [47, 48, 91]. The comoving volume, V_{C}, is defined under the assumptions of isotropy and homogeneity, implying that matter is uniformly distributed on large scales (greater than 100 Mpc). The differential comoving volume is

$$p(z) \propto \frac{dV_{\mathrm{C}}}{dz} = \frac{c^3}{H_0^3}\left(\int_0^z \frac{dz'}{E(z')}\right)^2 \frac{1}{E(z)}. \tag{2.20}$$

When normalised, this volume becomes independent of H_0, since it is only a function of redshift, making it suitable for modeling galaxy distributions as a function of redshift [92]. To maintain a uniform distribution in comoving volume, the corresponding prior in terms of luminosity distance is

$$p(D_{\mathrm{L}}) = p(V_{\mathrm{C}})\left|\frac{dD_{\mathrm{L}}}{dV_{\mathrm{C}}}\right| \propto D_{\mathrm{L}}^2. \tag{2.21}$$

This approach aligns with the analysis in Ref. [16].

Apart from luminosity distance, other GW parameters, such as spin magnitudes, component masses, spin orientation, sky location, and binary orientation, are assigned uniform or isotropic priors [48]. For instance, spin magnitudes and redshifted component masses have uniform priors, while spin orientation and sky location follow isotropic distributions, meaning they are equally likely in all directions. This approach allows for an unbiased sampling of the parameter space [48]. Therefore, the term in the denominator of Eq. (2.12) that accounts for the prior probability of the GW parameters used during PE is

$$p(\boldsymbol{\theta}_{\mathrm{GW}}|\Omega_0) \propto p(D_{\mathrm{L}}) \propto D_{\mathrm{L}}^2, \tag{2.22}$$

where the priors on the other GW parameters are uniform and thus contribute as constants, which cancel out upon normalisation [48]. Further details on these priors will be discussed in Sec. 2.5.

2.3 CosmoFlow: Data Generation

2.3.1 Short overview

In the Bayesian framework outlined in Sec. 2.2, our primary focus is evaluating Eq. (2.12). To do so, we plan to approximate the numerator using a machine learning strategy. A NF, as introduced Sec. 1.8, is the ideal choice due to its capability to model complex distributions, which is essential for incorporating GW selection effects and EM information into the posterior distributions of cosmological parameters.

Our objective is to achieve a data distribution that mirrors the probability distribution $p(\boldsymbol{\theta}_{\mathrm{GW}}|\boldsymbol{\Omega}, \boldsymbol{D})$. This term reflects the prior probability of GW parameters conditioned on the cosmological parameters and the detectability of each GW event. This term is of high importance, as it will efficiently handle complex features inherent in cosmological datasets, such as galaxy clusters present in galaxy surveys and selection effects caused by the detectability of GW signals, which require careful consideration. For instance, the detectability of an event can be influenced by cosmological parameters. For a fixed redshift, sampled from either a galaxy catalogue or a comoving volume prior, higher values of $\boldsymbol{H}_0$ result in shorter luminosity distances, thereby increasing the SNR and, consequently, the probability of detection. Other parameters that could influence the results include population parameters, such as the maximum mass ($\boldsymbol{M}_{\mathrm{max}}$). Allowing for different maximum mass levels could increase the probability of detecting events at greater distances.

This section will give a brief overview of how the data generation process is performed, varying only one cosmological parameter, $\boldsymbol{H}_0$, while keeping other parameters, such as the mass model parameters and the dimensionless density parameter, $\boldsymbol{\Omega}_{\mathrm{m}}$, fixed. More details about the mathematical descriptions of all priors used in the data generation will be provided in the following sections.

The data generation process begins by choosing a specific cosmological model. In this case, a flat $\boldsymbol{\Lambda}$CDM model is used, the same as in Ref. [16], with $\boldsymbol{\Omega}_{\mathrm{m}} = 0.3$. We then uniformly sample the cosmological parameters, $\boldsymbol{\Omega}$, which, for the one-dimensional case, would only be $\boldsymbol{H}_0$. We proceed by selecting redshifts sampled from a comoving volume prior distribution, reparameterized to account for the merger rate in the detector frame, as defined in [93].

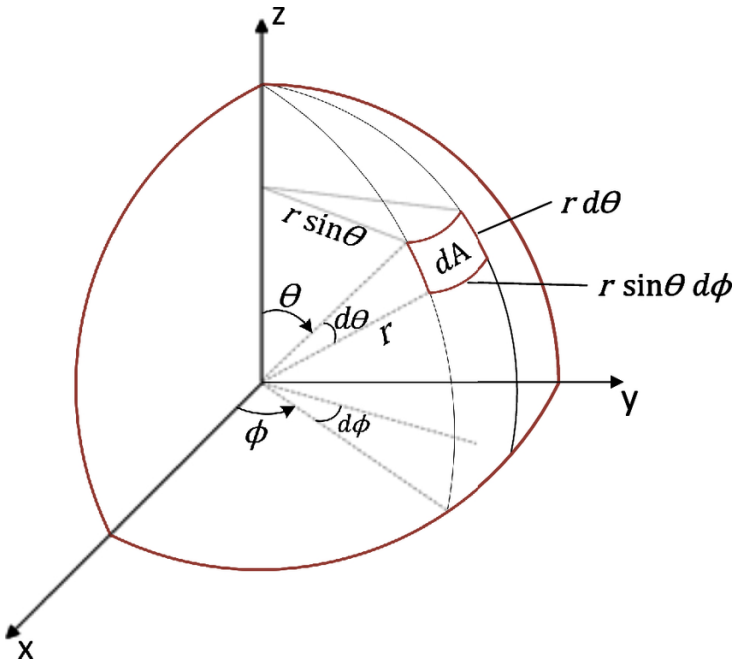

Figure 2.2: A schematic representation of spherical coordinates. The polar angle θ starts from the north pole, and the azimuthal angle ϕ is the longitudinal coordinate. The small surface element dA is shown with sides $r\,d\theta$ and $r\sin(\theta)\,d\phi$. Image credit [94].

We then sample sky locations over the 2D sphere by uniformly sampling right ascension, α, from a uniform distribution between 0 and 2π, and declination, δ, from a uniform distribution in cosine between -1 and 1. This method ensures uniform sky coverage, avoiding clustering at the poles. Mathematically, the area element on a sphere in spherical coordinates is given by:

$$dA = r^2 d\theta \sin(\theta)\, d\phi. \tag{2.23}$$

Here, ϕ and θ correspond to the longitudinal and latitudinal polar coordinates, with θ starting from the pole, and r is the radial component. In Fig. 2.2 a schematic representation of spherical coordinates is provided, showing the longitudinal and latitudinal coordinates with radius r. A small element of the surface area is also illustrated, with sides $r\,d\theta$ and $r\sin(\theta)\,d\phi$. Since $\sin(\theta) = \cos(\delta)$, the prior distribution in declination, normalised between $-\pi/2$ and $\pi/2$, is given by:

$$p(\delta)d\delta = \frac{\cos(\delta)d\delta}{\int_{-\pi/2}^{\pi/2}\cos(\delta)\,d\delta} = \frac{1}{2}\cos(\delta)d\delta. \tag{2.24}$$

Following the selection of cosmological parameters, redshifts, and sky locations, absolute magnitudes are sampled using a luminosity function, also known as the Schechter function [95], described as

$$p(L)dL = \phi^*\left(\frac{L}{L^*}\right)^{\alpha}\exp\left(-\frac{L}{L^*}\right)dL. \tag{2.25}$$

In this function, ϕ^* represents the normalisation factor, L^* denotes the characteristic luminosity, and α is the faint-end slope. To convert this luminosity function to absolute magnitudes, we use the relation

$$M = M^* - 2.5 \log_{10}\left(\frac{L}{L^*}\right), \tag{2.26}$$

where M^* is the absolute magnitude corresponding to L^*. Since we are interested in using a specific galaxy catalog, the luminosity function we will use follows the one fitted for the galaxy catalog. To align with [16], we will use the GLADE+ catalog. For the K-band, the Schechter function parameters are $M_K^* = -23.39$ and $\alpha_K = -1.09$ [96]. The bright cut-off is set to $M_{\min,K} = -27.00$ and the faint cut-off is set to $M_{\max,K} = -19.0$, corresponding to galaxies with luminosity $L > 0.017 L_K^*$.

In preprocessing, the catalog is divided into individual pixels using the `HealPy` [97, 98] module. `HealPy` is based on the HEALPix (Hierarchical Equal Area isoLatitude Pixelation) framework, which divides the entire sky into equal-area pixels. Each pixel represents a discrete area of the sky and is identified by a unique index, allowing for efficient storage, manipulation, and analysis of the catalog corresponding to different sky regions. Consequently, from each sky sample, we can determine the pixel location on the sky and correlate this with the preprocessed pixelated galaxy catalog. With the galaxy catalogue pixelated, we can compute the magnitude threshold map of the galaxy survey. A magnitude threshold map is a tool used to filter galaxies based on their apparent magnitudes. In `gwcosmo`, this map is created by taking the median of apparent magnitudes within each pixel and discarding all galaxies with magnitudes above this threshold. The median is used because it is a robust statistic that effectively represents the central tendency of the apparent magnitudes within each pixel. Unlike the mean, the median is not heavily influenced by outliers or extreme values, making it a more reliable measure for this purpose. This approach ensures only the most significant and detectable galaxies are considered in the analysis [14]. By focusing on galaxies that are bright enough to be reliably detected, the magnitude threshold map helps improve the accuracy and efficiency of the analysis.

After selecting a host galaxy, we determine whether the galaxy should be sampled from the catalogue or remain as a sample from the prior. To make this decision, we use the magnitude threshold map. We compute the apparent magnitudes of the sampled galaxies, with

$$m = M + 5 \log_{10}\left(\frac{D_{\mathrm{L}}}{[\mathrm{Mpc}]}\right) + 25, \tag{2.27}$$

where m represents the apparent magnitude of the galaxy, M is the absolute magnitude of the galaxy, and D_{L} is the luminosity distance to the galaxy in megaparsecs (Mpc). The decision then to select a galaxy from the catalogue depends on the comparison between the apparent magnitude of the sampled host galaxy and the magnitude threshold provided by the catalogue for that galaxy's specific pixel location. Essentially, this step determines whether the sampled galaxy meets the visibility criteria set by the catalogue. If the galaxy's apparent magnitude is brighter than the catalogue's threshold at that pixel, it suggests that the galaxy would realistically be part of the catalogue under typical observational conditions. Therefore, we sample a galaxy from the catalogue pixel, weighting it by its intrinsic luminosity and down-weighting by a factor of $(1+z)$ to account for the time delay between different host galaxies at different redshifts. This means that the weighting factor for each galaxy within a pixel is $L/(1+z)$. Otherwise, if the apparent magnitude is higher than the magnitude threshold map value at the specific pixel, we retain the simulated galaxy as originally sampled from the priors.

Then we proceed to place synthetic compact binary coalescence (CBC) events, such as BBHs or NSBHs systems, within these galaxies. This is done by sampling the GW parameters that describe the strain-data waveform model (14 in total, excluding the phase parameter) from uniform priors, except for the masses, and associating each set of GW parameters with a selected host galaxy.

In this section, we introduce the population models of the events, describing the probability distribution of the primary and secondary masses. To align our results with [16], we use the Power Law Plus Peak mass distribution for m_1 and m_2, which is described in detail in Section 2.5.1. It is important to note that the events we generate will be in detector-frame mass. Therefore, the masses are coupled with the respective redshift value obtained from the host galaxy, which could be a galaxy from the catalogue or otherwise, thus obtaining the detector frame masses as

$$m_{\mathrm{detector}} = m_{\mathrm{source}}(1+z), \tag{2.28}$$

where m_{detector} are the detector-frame masses, m_{source} are the source-frame masses and z is the redshift. The parameter-estimation process involves 15 parameters that describe the strain-data waveform model. These parameters include the component masses m_1 and m_2, the spin magnitudes a_1 and a_2, the spin tilt angles θ_1 and θ_2, the inclination angle θ_{JN}, the luminosity distance D_{L}, the right ascension α, the declination δ, the polarization angle ψ, the time of coalescence t_c, and the phase ϕ at coalescence.

Therefore, for the remaining 10 parameters, the spin magnitudes a_1 and a_2 are assumed to follow uniform distributions between 0 and 1, while the spin tilt angles θ_1 and θ_2 are uniformly distributed between 0 and π. The inclination angle $\theta_{\rm JN}$ is uniformly distributed in cosine between 0 and π. The polarization angle ψ is uniformly distributed between 0 and π. The time of coalescence t_c is uniformly distributed over the observation period of one sidereal day (86164.0905 seconds). The azimuthal angle ϕ_{12}, which describes the position compared to the orbital plane, and ϕ_{jl}, which describes the orientation of the total angular momentum relative to the line of sight, are uniformly sampled between 0 and 2π [99]. Finally, the phase ϕ at coalescence is omitted because it does not influence the matched-filtered SNR and therefore has no impact on the detectability of the events we synthetically generate.

After sampling GW parameters from their priors, we compute the optimal SNR associated with the resulting waveforms. This step is recognized as the primary bottleneck in the data generation process. The bottleneck arises because the computation of the optimal SNR for a synthetic event relies on the `bilby` package [100]. This package allows us to pass the sampled GW parameters, along with other initialization components and a defined power spectral density (PSD) for the detectors used, to compute the optimal SNR.

A key challenge is that this computation may not be easily parallelizable, requiring the evaluation of millions of SNRs individually, which significantly impacts performance. Consequently, this significantly slows down the process. Following the consideration of selection effects, only GW events with an SNR above a certain threshold, $\rho_{\rm th}$, are retained along with their associated cosmological parameters. This leads to inefficiencies, as many of the sampled parameters fall within regions of the parameter space that are not easily detectable, resulting in the majority of synthetic events being discarded. To mitigate this inefficiency, a machine learning approach is employed. Specifically, a Multi Layer Perceptron (MLP) is used to approximate the optimal SNR calculation. This method significantly accelerates the process by providing a fast, approximate computation of the SNR.

Events that are not detected undergo resampling of all parameters except for the cosmological parameter. This is then fed back into the initial stage of the data generation process, where all host galaxy and GW parameters are resampled. This iterative process continues until a combination of intrinsic and extrinsic GW parameters results in a detectable

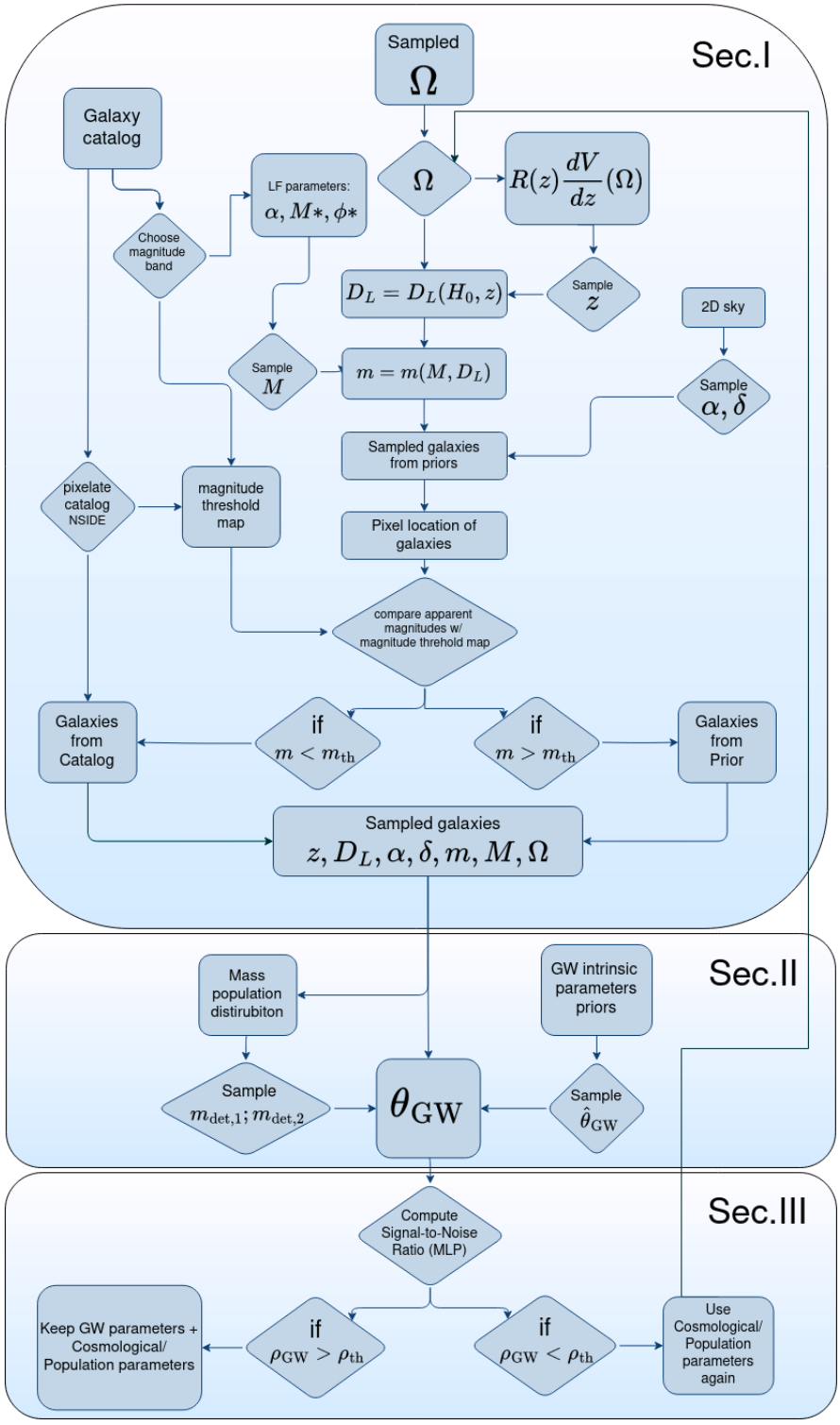

Figure 2.3: Flow chart representing the simulation process used for GW event parameter generation conditional on cosmological and population parameters. Here the $\hat{\theta}_{\rm GW}$ correspond to the spin magnitudes, angular orientations and geocentric time of arrival parameters.

event. This approach ensures that the generated cosmological data ultimately matches the chosen prior distribution of cosmological parameters. A detailed flowchart illustrating the data generation process is presented in Fig. 2.3. The data generation process can be split into three sections.

In Section I, cosmological and population parameters are sampled uniformly, followed by sampling redshifts of galaxies hosting GW events. These parameters are combined to compute the luminosity distance for each event. A galaxy catalogue provides luminosity function parameters, from which absolute magnitudes are sampled. With the luminosity distance, apparent magnitudes are computed. Host galaxies are placed on a 2D sky map, assigned to pixels based on resolution. If a galaxy's magnitude is below the threshold, we sample from the galaxy catalogue within that pixel; otherwise, the previously selected host galaxies are retained. In Section II, GW parameters are sampled, with source masses following a power law plus peak distribution [87]. Luminosity distance, redshifts, and sky locations are taken from the host galaxies. Remaining GW parameters are sampled from uniform priors. In Section III, the GW parameters are passed as inputs into the SNR approximator defined in Sec.2.6.1. If the SNR exceeds the threshold, all GW and associated cosmological parameters are retained. If below the threshold, cosmological parameters are reused, while other parameters are resampled until a GW event is detected. The next section delves into the data generation process, split into three parts:

- Selecting Host Galaxies, Sec.2.4, Fig. 2.3 (Sec.I);
- Selecting GW parameters, Sec.2.5, Fig. 2.3 (Sec.II);
- Selecting detected GW events, Sec.2.6.1, Fig. 2.3 (Sec.III);

2.4 Selecting Host Galaxies

In this section, we will look at the initial step of the data generation process: the selection of host galaxies for synthetic CBC events. By selection of galaxies, we refer to the determination of specific astronomical parameters, as in, right ascension (α), declination (δ), redshift and luminosities. These parameters for each galaxy are either drawn from predefined priors or directly sourced from the galaxy catalogue.

First, we will discuss the prior distributions for host galaxy redshifts. Next, we will examine the GLADE+ galaxy catalogue, focusing on its characteristics, particularly the luminosity function and its fitting parameters. We follow the methodology from [16] to compute a *magnitude threshold* map, which is important due to the flux-limited nature of the GLADE+ catalogue. This map helps differentiate between galaxies that are detectable in the catalogue and those that are not within our data generation process. Finally, we will explain our approach for deciding whether to retain a sampled galaxy from the priors or select it from the galaxy catalogue.

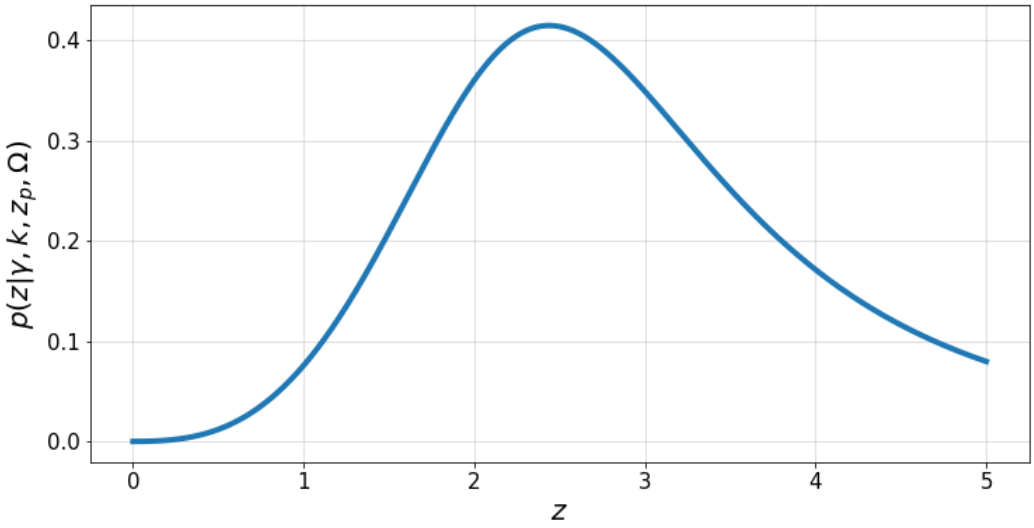

Figure 2.4: 1D distribution of GW redshifts in host galaxies given the Madau and Dickinson phenomenological model expressed in Eq. (2.29). The values used are $\gamma = 4.59$, $k = 2.86$ and $z_p = 2.47$ with a Flat ΛCDM universe with $\Omega_{\rm m} = 0.3$.

2.4.1 Redshift distribution for host galaxies

We first choose a prior on H_0 and fix the baryonic density parameter at $\Omega_{\rm m} = 0.3$ within a flat ΛCDM model, and then we sample the cosmological parameters. Next, we sample the redshifts of the host galaxies. To do so, we use the assumption that dark-sirens BBH GW events are localized within galaxies hosting them, thus it is possible to model the binary merger rate using a phenomenological model introduced by Ref. [101], and parameterized by Ref. [93], as follows:

$$p(z|\gamma,k,z_p,\Omega) = \frac{dN}{dtdz} = \mathcal{N} R_0 \left[1+(1+z_p)^{-\gamma-k}\right] \frac{1}{1+z} \frac{\partial V_c(\Omega)}{\partial z} \frac{(1+z)^{\gamma}}{1+\left(1+z/1+z_p\right)^{\gamma+k}}, \quad (2.29)$$

where R_0 is the merger rate today, z_p is the characteristic redshift point between the two power-law regimes (corresponding to the maximum of the function), γ is the slope of the power-law regime for the rate-evolution before the point z_p, and k is the slope of the power-law regime for the rate-evolution after the point z_p, and $\mathcal{N}$ is the normalisation factor. The $1/(1+z)$ factor allows for the transformation from the source rest frame to the detector frame in the time dimension, while the factor $(1+z_p)$ ensures that the merger rate today is R_0 by normalising the equation such that terms involving $(1+z_p)^{-\gamma-k}$ cancel out at $z = 0$, thereby simplifying the expression to R_0 when evaluated at $z = 0$ [101].

The phenomenological distribution described in Eq. (2.29) for GW events is hypothesized to correlate with various astrophysical processes, including the rate of binary system formations and star formation rates within host galaxies [101]. This distribution is essential for understanding the evolution of GW event rates over cosmic time. By linking GW event rates with astrophysical processes such as star formation, the model provides valuable insights into the distribution of GW events across different redshifts.

The population parameters that describe Eq. (2.29) will be held fixed at the following values: $\gamma = 4.59$, $k = 2.86$, and $z_p = 2.47$ as described in [87]. These are the same values used in the analysis by [16]. A representation of the redshift distribution of GW events is shown in Fig. 2.4. Using the PDF described in Eq. (2.29), we sample redshift values corresponding to host galaxies of BBH mergers.

2.4.2 Galaxy catalogue, GLADE+

The GLADE+ [81] catalogue is an extended version of the GLADE [102] galaxy catalogue. It is comprised of six distinct yet interconnected astronomical catalogues: the GWGC [103], 2MPZ [104], 2MASS XSC [105], HyperLEDA [106], and WISExSCOSPZ [107, 108] galaxy catalogues, alongside the SDSS-DR16Q quasar catalogue [109].

The GLADE+ catalogue covers the entirety of the sky, with exception of the region covered by the galactic band of the Milky Way. A sky map of the GLADE+ catalogue is shown in Fig. 2.5, where the galactic band of the Milky way is clearly visible. The map was generated using the K-band galaxies from the catalogue with a resolution of NSIDE = 64. Overdensities, such as the central region depicted in Fig. 2.5, originate from the HyperLEDA catalogue. These overdensities arise due to more in-depth and sensitive surveys, like SDSS and GAMA, conducted in the corresponding sky areas [81]. GLADE+ contains approximately 22 million galaxies and it covers the full 2D sphere of the sky, with a completeness of 20% up to 800 Mpc [81]. Catalogue completeness refers to how well a catalogue represents the actual population of galaxies. For a flux-limited catalogue like GLADE+, completeness assesses the extent to which the catalogue includes galaxies of varying brightness across different distances [81]. Higher completeness indicates a more comprehensive inclusion of galaxies, especially those that are more distant and fainter in magnitude. Due to the flux-limited nature, as the distance increases, the catalogue increasingly misses fainter galaxies, which impacts the overall completeness.

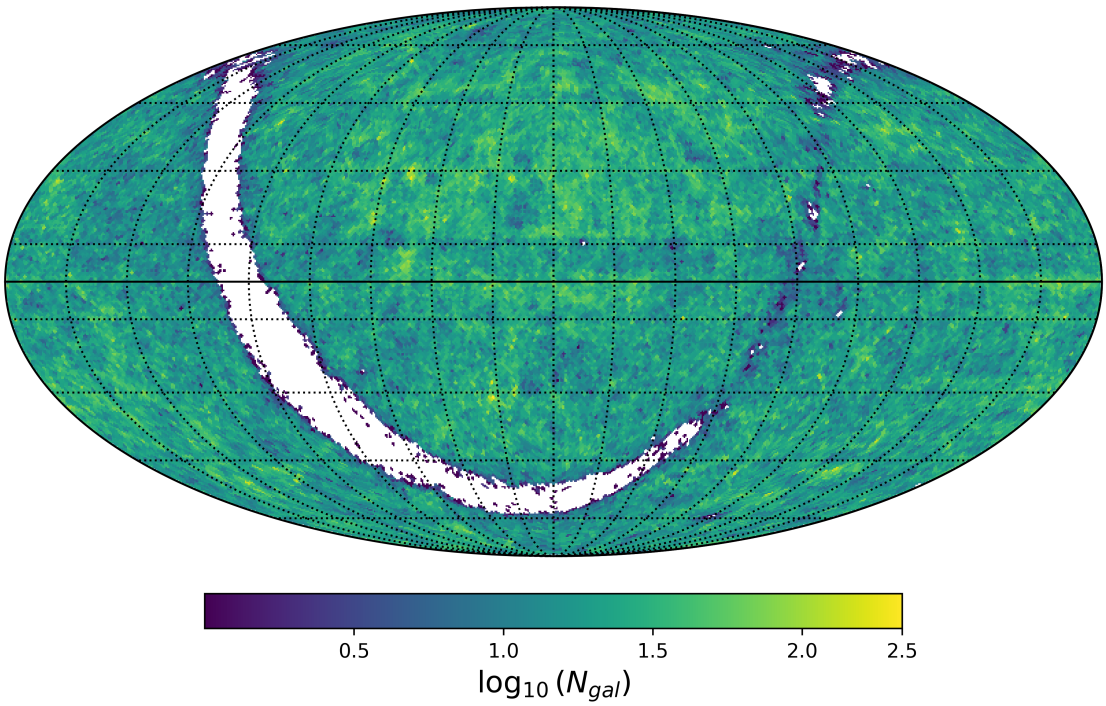

Figure 2.5: Mollweide projection of GLADE+ catalogue showing bin count on each pixel with a resolution of NSIDE = 64, with a total number of pixels of $N_{\rm pix} = 12 \times \mathrm{NSIDE}^2 = 49152$. The galaxies shown here are only for the K-band and a mask with a minimum galaxy per pixel has been applied of $N_{\rm th} = 1$, setting the count to zero if there are 1 or less galaxies in that pixel, for plotting purposes.

To address the incompleteness of GLADE+, we generate a *magnitude threshold map* from the catalogue. The completeness of a galaxy catalogue is typically assessed based on an apparent magnitude threshold [110], denoted as $m_{\rm th}$. This threshold is determined by calculating the median apparent magnitude from the distribution of galaxies for each pixel, establishing a limiting magnitude in every direction. We adopt the methodology described in Ref. [111], where the median apparent magnitude of all galaxies in each pixel is used to set the threshold, discarding galaxies dimmer than this threshold. The magnitude threshold map aims to approximate a fully complete catalogue, meaning it includes all galaxies above a certain apparent brightness and ignores those that are dimmer than the threshold. This method simplifies the analysis by removing the need to account for varying levels of completeness across different regions, ensuring that only galaxies bright enough to meet the threshold in each pixel are considered. Figure 2.6 illustrates this concept, showing the median K-band apparent magnitude from each pixel as the threshold for the apparent luminosity of galaxies within each pixel. This magnitude threshold map is employed in the data generation process to distinguish between galaxies sampled from the prior or from the GLADE+ catalogue.

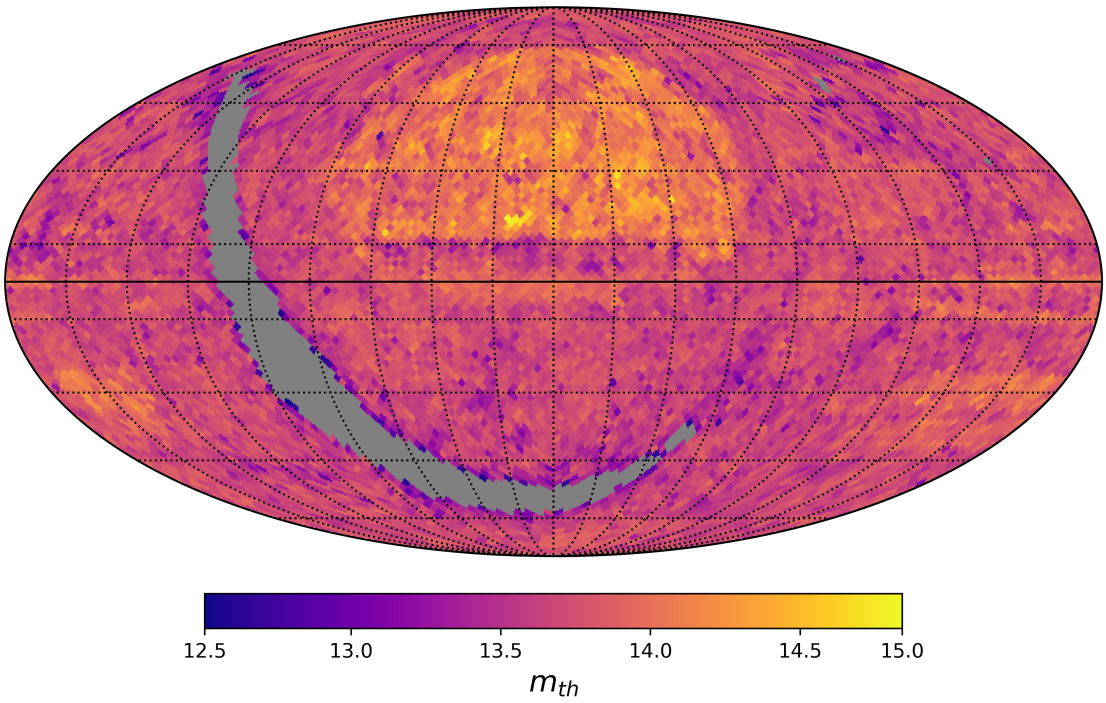

Figure 2.6: Mollweide projection of the magnitude threshold map of GLADE+ catalogue using the K-band. The resolution is set to NSIDE = 32. The shaded grey area represents the galactic band of the Milky Way, where the values of $m_{\rm th} = 0$ for a better figure readability.

Owing to its extensive sky coverage, GLADE+ stands out as one of the best catalogues for following up on GW events. This advantage, however, comes with the trade-off of having somewhat limited depth in redshift measurements. Other catalogues, such as the Dark Energy Survey (DES) Year 1 [9] and the DESI Legacy Imaging Survey [112], are expected to be complete up to redshift $z \sim 1$. Nevertheless, we choose to employ the same all-sky galaxy catalogue for all our events, following the analysis in Ref. [16]. Future work will explore combining data from different catalogues to perform data generation over various galaxy surveys, allowing for more in-depth information about redshifts.

Most of the redshifts in the GLADE+ catalogue are photometric and are estimated using a machine learning model described in Ref. [113]. The photometric redshifts are determined by analyzing the magnitudes across various color bands to discern the extent of redshifting of the source. To obtain the redshift information, Ref. [113] employs an artificial NN, named ANNz, designed to estimate photometric redshifts from apparent magnitudes across different color bands. After training, the model achieves a relative error of $\sigma \approx 0.033(1+z_{\rm ph})$, where $z_{\rm ph}$ represents the photometric redshift [114]. The catalogue also includes corrections for peculiar velocities, obtained using a Bayesian technique as described in Ref. [115].

2.4. Selecting Host Galaxies

The catalogue consists of two main magnitude bands: the B_J band and the K_s band (hereafter referred to as the K-band). In this analysis, we use all galaxies with measured K–band luminosity reported in the Vega system. This band was chosen because of a good match between the galaxy luminosity function and the galaxy number density in the GLADE+ catalogue [16]. Each spectral band has its own luminosity distribution, which follows the Schechter function, described in Eq. (2.25). This function was first described in the Press-Schechter formalism [95]. It is mathematically described by a Gamma function (for more information about Gamma functions, see Ref. [116]), as

$$p(L)dL = \phi^* \left(\frac{L}{L^*}\right)^{\alpha} \exp\left(-\frac{L}{L^*}\right) dL, \tag{2.30}$$

and the parameters are $p(L)dL$, which is the number of galaxies per unit volume within a luminosity interval L and $L+\Delta L$, ϕ^* is the normalisation constant and L^* is a characteristic luminosity, indicating the luminosity at which the exponential cutoff of the distribution occurs [95]. We can also transform this distribution to absolute magnitudes, M, which are more practical for our purposes. By using the absolute magnitude in Eq. (2.27) (the distance modulus equation) together with the luminosity distance, we can obtain the apparent magnitudes. To do so, we can use the change of variable equation, where $p(L)dL = p(M)dM$, where the relationship between absolute magnitudes and intrinsic luminosity can be described as

$$M = -2.5\log_{10}(L) + \text{const.}, \tag{2.31}$$

where the constant can be associated with any known luminosity for which we have a corresponding absolute magnitude (i.e. for the Sun, const. $= M_{\odot,\mathrm{K}} + 2.5\log_{10}(L_\odot)$), where $M_{\odot,\mathrm{K}} = 3.28$ is the absolute mangitude of the Sun in the K-band [117]. By applying the relationship in Eq. (2.31) to the change of variable in Eq. (2.30), we obtain the Schechter function described in terms of absolute magnitude

$$p(M) = 0.4\ln(10)\,\phi^*\,10^{0.4(M^*-M)(\alpha+1)} \exp\left(-10^{0.4(M^*-M)}\right). \tag{2.32}$$

For the K-band, parameters from [118] are adopted, with $\phi^* = 1.16102 \times h^3\mathrm{Mpc}^3$, $h \equiv H_0/100$, $M^* = 23.39$, and $\alpha = 1.09$, where the fit has been applied in the luminosity range of $M_K = [-19, -27]$. These parameters were fitted using a value of the Hubble constant fixed at $H_0 = 100\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$. Color and evolution correction terms have been applied, modifying the distance modulus relationship in Eq. (2.27) to:

$$m_{\mathrm{corr}} = m + K, \tag{2.33}$$

where $m_{\rm corr}$ is the corrected apparent magnitude, K is the correction factor given by $K = -6.0\log_{10}(1+z)$ for redshifts $z > 0.25$, as detailed in [118].

In the data generation process, we assume that the luminosity function of the entire universe, or at least up to the maximum distance of detectable events, can be described by the Schechter function as detailed in Eq. (2.32). We set a bright cut-off high enough to include all the bright galaxies supported by the Schechter function: $M_{\min,K} = -27.00$ and consider all the galaxies no fainter than $M_{\max,K} = -19.0$, for an $h = H_0/100 = 1$. These choices correspond to all galaxies with luminosity $L > 0.017L_K^*$, where L^* is the characteristic galaxy luminosity of the Schechter function [16].

An important consideration for this study, particularly because we focus on host galaxies of GW events, is the assumption of *luminosity weighting* as used in [16]. The assumption is that galaxies more likely to host CBCs events are typically more massive and, as a result, more luminous in certain bands. The underlying reasoning is that galaxies with a higher number of compact binaries are generally heavier, leading to increased luminosity. Therefore, more massive and luminous galaxies are assumed to have a higher probability of hosting GW events. However, given the absence of a definitive empirical relationship between galaxy properties and the occurrence of CBC events, we adopt a conservative approach by applying a linear weighting. Consequently, the distribution we are interested in is represented as

$$p(L)L = \phi^* L^* \left(\frac{L}{L^*}\right)^{\alpha+1} \exp\left(-\frac{L}{L^*}\right). \tag{2.34}$$

When expressed in terms of absolute magnitude, Eq. (2.34) becomes

$$p(M) \propto 10^{0.4(M^*-M)(\alpha+2)} \exp\left(-10^{0.4(M^*-M)}\right). \tag{2.35}$$

This equation modifies Eq. (2.32) by incorporating an additional term in the exponential factor. Specifically, an extra factor of L introduces an additional factor of $10^{-0.4*M}$. The distribution described in Eq. (2.35) will be used as the prior distribution for sampling galaxy luminosities that host CBC throughout the entire analysis. From this distribution, we sample absolute magnitudes before determining whether the sampled galaxy should be included in the catalogue or not.

Therefore, after sampling the cosmological parameters and the redshifts, and computing the luminosity distance, we proceed to sample the absolute magnitude from the luminosity function defined in Eq. (2.35). One important aspect of the luminosity function described in Eq. (2.35) is that it remains mathematically invariant when varying H_0. This

is because H_0 affects the distance measurements and, consequently, the luminosity calculations. Changing H_0 scales the luminosities but does not alter the relative distribution of galaxy luminosities. In this case, since the shape of the distribution is invariant under changes in H_0, we can sample the Schechter function with a fixed value of $h = H_0/100 = 1$. We then shift the magnitude by $M(h) = M_{100} + 5\log_{10}(h)$, where M_{100} is the absolute magnitude sampled under the condition that $h = 1$.

To derive this equation, we start with the distance modulus formula, using the distance in parsecs:

$$m - M = 5\log_{10}(D_{\mathrm{L}}) - 5. \tag{2.36}$$

The luminosity distance D_{L} depends on the Hubble constant H_0. For a given redshift z, $D_{\mathrm{L}} \propto 1/H_0$. Let $D_{\mathrm{L},100}$ be the luminosity distance for $H_0 = 100\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$, then $D_{\mathrm{L}} = D_{\mathrm{L},100}/h$, then

$$\log_{10}(D_{\mathrm{L}}) = \log_{10}(D_{\mathrm{L},100}) - \log_{10}(h). \tag{2.37}$$

Substituting back into the distance modulus formula in Eq. (2.36) and rearranging, we get

$$m - M = 5\log_{10}(D_{\mathrm{L},100}) - 5\log_{10}(h) - 5. \tag{2.38}$$

For $H_0 = 100$, the absolute magnitude is M_{100}, so $m - M_{100} = 5\log_{10}(D_{\mathrm{L},100}) - 5$. Combining these, we obtain

$$M = M_{100} + 5\log_{10}(h). \tag{2.39}$$

Therefore, when sampling from Eq. (2.35), the absolute magnitude can be shifted to the correct luminosity given by Eq. (2.39).

This adjustment changes the boundaries of possible luminosities that can be sampled. Specifically, the minimum (dimmer) boundary is calculated as $-19 + 5\log_{10}(140/100) = -18.85$ and the maximum (brighter) boundary as $-27 + 5\log_{10}(20/100) = -27.69$, assuming a range of H_0 from $20\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$ to $140\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$.

Next, by uniformly assigning a sky position ($\alpha \sim U[0, 2\pi]$ and $\sin(\delta) \sim U[-1, 1]$), we can associate each sampled galaxy with a specific pixel location. With the absolute magnitude (correctly shifted) and the luminosity distance, we compute the apparent magnitude, from which we then compare with the threshold magnitude m_{th} of the specific pixel in which the galaxy was sampled, and decide whether to retain the prior sampled galaxy or to select one from the catalogue.

2.4.3 Selecting a galaxy from the catalogue

When sampling a galaxy from the catalogue, we follow these steps: first, identify all galaxies that meet the selection criteria based on their computed apparent magnitude being less than the specified magnitude threshold, second, access the pixelated GLADE+ catalogue corresponding to the sky locations we sampled, and finally sample from the pixel catalogue giving weights to each galaxy within each pixel by the luminosity and redshift.

From each sky sample, an associated pixel can be obtained, along with the cosmological parameters related to that pixel. We then count the number of galaxies within each pixel based on the catalogue data and compute the luminosity distance D_{L} for each galaxy using the cosmological parameters associated with the pixel. Using the apparent magnitudes of the galaxies, m_i, we can compute their absolute magnitudes given the luminosity distances using Eq. (2.27). From the absolute magnitudes, we can then determine their luminosities, L_i using Eq. (2.31). Within each pixel, galaxies are weighted by their luminosity and redshift, with the weight for each galaxy defined as:

$$w_i = \frac{L_i}{1+z_i}, \tag{2.40}$$

where L_i is the luminosity of the ith galaxy and z_i is its redshift. This weighting method follows Ref. [16], where luminosities are used to upweight galaxies, reflecting a linear relationship $w \propto L$. Conversely, the time dilation factor, $(1+z)^{-1}$, is used to downweight more distant galaxies, accounting for the reduced contribution of high redshift sources.

One important point is that the selection of the pixel from which to sample a galaxy from the galaxy catalogue is done before introducing any catalogue information. Since the sky is sampled uniformly, each pixel has the same probability of being selected, which can bias the results. In reality, galaxy catalogues have a well-defined distribution of luminosities, making some regions more likely to host GW events due to higher galaxy densities. This issue has been studied by implementing a weighted sky sampling approach, where pixels are weighted by the luminosity of galaxies within them, assigning higher probabilities to regions with more luminosity. However, this method was computationally slow due to the dependence of the luminosity map on cosmological parameters and high pixel variability. Therefore, we proceeded with the uniform sky distribution approach, which was deemed sufficient. Future work will explore weighting sky locations by galaxy catalogue luminosity.

To sample a galaxy, we use a multinomial distribution. Each galaxy within the pixel is assigned a weight based on Eq. (2.40), and the probability of selecting a specific galaxy is proportional to its weight. This ensures that galaxies with higher luminosity and lower redshift are more likely to be selected to host GW events. We then select one galaxy from the pixel using the weights in Eq. (2.40) and maintain its properties for further analysis. Accounting for the uncertainty in redshift in the catalogue, we select the redshift value z from the catalogue, along with its associated 1σ uncertainty σ_z. To ensure that the sampled redshift value remains within realistic bounds (i.e., non-negative), we employ a truncated normal distribution. The PDF for the truncated normal distribution is

$$f(z|\mu,\sigma) = \frac{1}{\sigma_z\sqrt{2\pi}} \exp\left(-\frac{(z-\mu)^2}{2\sigma_z^2}\right) \Bigg/ \left[\operatorname{erf}\left(\frac{b-\mu}{\sigma_z}\right) - \operatorname{erf}\left(\frac{a-\mu}{\sigma_z}\right)\right], \tag{2.41}$$

where μ is the measured redshift value from the catalogue, σ_z is the standard deviation from the catalogue, erf is the error function of the standard normal distribution, and a and b are the lower and upper bounds of the truncation, respectively. In our case, we set $a = 0$ to ensure non-negative redshifts and $b \to \infty$. This method ensures that all sampled values are valid and remain within the defined physical constraints. The truncated normal distribution was implemented using `scipy.stats.truncnorm` [90].

Thus, we compute the PDF of in-catalogue redshifts after selecting a pixel, expressed as

$$p(z|G, \mathrm{pix}(\alpha,\delta)) \propto \sum_{i=1}^{N_{\mathrm{gal}}} w_i f(z|\mu_i,\sigma_i), \tag{2.42}$$

where G denotes the conditional sampling from the catalogue, and $\mathrm{pix}(\alpha,\delta)$ refers to the pixel indexed by right ascension α and declination δ. Equation (2.42) is a weighted sum over all galaxies within the pixel, weighted by w_i, combining all the truncated Gaussian distributions within the pixel, $f(z|\mu_i,\sigma_i)$, where μ_i and σ_i are the observed redshift and its standard deviation of the ith galaxy within the selected pixel.

2.4.4 Note on sampling from the luminosity function

In this section, we will shed light on how to sample from the luminosity function (i.e., the Schechter function) defined in Section 2.4.2. To sample efficiently from the Schechter function fitted for GLADE+, we first need to understand the distribution it describes in Eq. (2.35).

The Gamma (factorial) function is defined as

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t}\, dt. \tag{2.43}$$

This function extends the concept of factorial to non-integer values. For more information, refer to the *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables* by Abramowitz and Stegun [119].

From Eq. (2.43), we can describe a PDF, namely the Gamma distribution, as

$$f(x;k,\theta) = \frac{x^{k-1} e^{-x/\theta}}{\theta^k \Gamma(k)}, \tag{2.44}$$

where k is the shape parameter and θ is the scale parameter [120]. For $\theta = 1$ and $k = L/L^*$, we can recover the Schechter function described in Eq. (2.30). Since we are interested in the luminosity-weighted Schechter function, as described in Eq. (2.34), we need the following exponent to be defined as $k - 1 = \alpha + 1$. Therefore, the shape parameter for describing the luminosity-weighted Schechter function is $k = \alpha + 2$.

We also define the *upper incomplete gamma* function, which is a generalization of the gamma function that allows for integration over a finite range starting from a non-zero lower bound [119]. It is defined as

$$\Gamma(a,x) = \int_x^\infty t^{a-1} e^{-t}\, dt. \tag{2.45}$$

To sample the luminosity-weighted Schechter function described in Eq. (2.34), we need to compute its cumulative distribution function (CDF). Considering the cutoffs in luminosities, defined in Section 2.4.2, the sampling can only be done starting from a minimum luminosity cut off, $L_{\min}$, corresponding to the minimum absolute magnitude value of $M_{\min} = -19$.

We begin the derivation from the un-normalised luminosity-weighted Schechter function, given by:

$$p(L)L \propto \phi^* L^* \left(\frac{L}{L^*}\right)^{\alpha+1} e^{-L/L^*}. \tag{2.46}$$

To find the CDF from $L_{\min}$ to L, we need to integrate the weighted luminosity function:

$$\mathrm{CDF}(L) \propto \int_{L_{\min}}^{L} \phi^* L^* \left(\frac{L'}{L^*}\right)^{\alpha+1} e^{-L'/L^*}\, dL'. \tag{2.47}$$

Factoring out the constants, we get:

$$\mathrm{CDF}(L) \propto \phi^* L^{*2} \int_{L_{\min}}^{L} \left(\frac{L'}{L^*}\right)^{\alpha+1} e^{-L'/L^*} \frac{dL'}{L^*}. \tag{2.48}$$

Substituting $x = L'/L^*$, hence $dL' = L^* dx$, we have:

$$\mathrm{CDF}(L) \propto \phi^* L^{*2} \int_{L_{\min}/L^*}^{L/L^*} x^{\alpha+1} e^{-x} dx. \tag{2.49}$$

Recognizing that the integral represents the incomplete gamma function, we can use the upper incomplete gamma function $\Gamma(s,x)$, to express the CDF as:

$$\mathrm{CDF}(L) \propto \phi^* L^{*2} \left[\Gamma(\alpha+2, L_{\min}/L^*) - \Gamma(\alpha+2, L/L^*)\right]. \tag{2.50}$$

Finally, to properly normalise Eq. (2.50), we need to integrate the starting distribution in Eq. (2.46) between the minimum and maximum values of the luminosities, which, in the case outlined in Section 2.4.2, follows:

$$\mathrm{CDF}(L) = \frac{\Gamma(\alpha+2, L_{\min}/L^*) - \Gamma(\alpha+2, L/L^*)}{\Gamma(\alpha+2, L_{\min}/L^*) - \Gamma(\alpha+2, L_{\max}/L^*)}, \tag{2.51}$$

where $L_{\max}$ corresponds to the maximum absolute magnitude of $M_{\max} = -27$. The final expression described in Eq. (2.51) is the analytical expression for the CDF of the luminosity-weighted Schechter function, which will be used to sample luminosities.

An interesting characteristic of Eq. (2.51) is that it depends on the ratio of luminosities, L/L^*. This term is independent of the Hubble constant, since, as described in Sec. 2.4.2, by keeping all cosmological parameters fixed except for H_0, both the luminosity L and the characteristic luminosity of the Schechter function, L^*, will be scaled the same way for a given value of H_0 (since we know that $M = M_{100} + 5\log_{10}(h)$, where M is the absolute magnitude). Therefore, if we use $h = 1$, we use the L^* fitted for that value, and when we sample a luminosity, we then shift its absolute magnitude by $5\log_{10}(h)$. Therefore, Eq. (2.51) is independent of H_0 (but not other cosmological parameters) and can be used to sample luminosities regardless of the H_0 value.

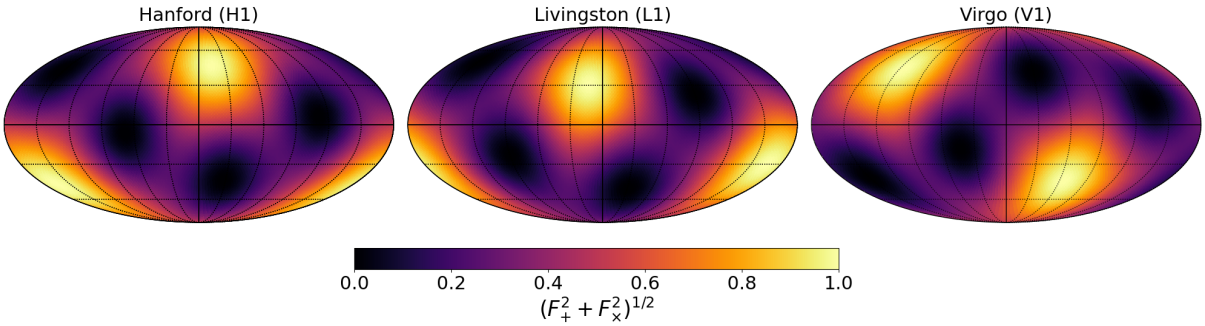

Figure 2.7: The antenna response function plotted over a Mollweide projection of the sky for each detector (H1, L1, and V1), at time $t = 0$ and $\psi = 0$. The color bar represents the total magnitude of the antenna response from both F_+ and $F_\times$, combined as $(F_+^2 + F_\times^2)^{1/2}$, spanning values between 0 and 1.

2.5 Selecting GW parameters

After selecting the host galaxies and deciding whether each galaxy would be from the galaxy catalogue or not, we can now begin the second stage of the data generation process, which is to sample the GW parameters for BBH type events.

The GW strain data is described by a waveform, parameterized by a set of 15 parameters. From the context of a geocentric reference frame, the strain $h^{(k)}(t|\vec{\theta})$, where t is the time series and $\vec{\theta}$ represent the 15 GW parameters, observed by the kth detector for a source with polarization amplitudes h_+ and $h_\times$, located at sky coordinates (α, δ), where α represents the right ascension and δ the declination, can be expressed as [121]

$$h^{(k)}(t|\vec{\theta}) = F_+^{(k)}(\alpha, \delta, \psi) h_+(t|\vec{\theta}) + F_\times^{(k)}(\alpha, \delta, \psi) h_\times(t|\vec{\theta}), \tag{2.52}$$

where $F_{+,\times}^{(k)}(\alpha, \delta, \psi)$ are the antenna response functions for GW detectors. These functions depend on the source's sky location (α, δ) and the GW's polarization angle ψ. These functions also vary with the time of coalescence and GPS time due to the Earth's rotation and orbital motion, which alter the detector's orientation and position relative to the source. Consequently, the sensitivity of the detectors to the GW's polarizations changes over time. For detailed calculations and models, see [122].

The antenna response functions can be described in polar coordinates (θ, ϕ) as

$$F_+ = -\frac{1}{2}\left[(1 + \cos^2\theta)\cos 2\phi \cos 2\psi - 2\cos\theta \sin 2\phi \sin 2\psi\right], \tag{2.53}$$

Name	Description	Distribution
m_1	Source frame mass of the first object in a binary system	PowerLaw+Peak [87]
m_2	Source frame mass of the second object in a binary system	PowerLaw+Peak ($m_2 < m_1$)
θ_1	Tilt angle of the first object	$\sin(\theta_1)$ for $\theta_1 \in [0, \pi]$
θ_2	Tilt angle of the second object	$\sin(\theta_2)$ for $\theta_2 \in [0, \pi]$
ϕ_{12}	Azimuthal angle compared to orbital plane	$U[0, 2\pi]$
ϕ_{jl}	Azimuthal position compared to orbital plane	$U[0, 2\pi]$
a_1	Dimensionless spin magnitude of the first object	$U[0, 0.999]$
a_2	Dimensionless spin magnitude of the second object	$U[0, 0.999]$
$D_{\rm L}$	Luminosity distance to the source	Comoving volume + EM
ι	Inclination angle relative to the line of sight	$\sin(\iota)$ for $\iota \in [0, \pi]$
ϕ	Phase of the wave at coalescence	$U[0, 2\pi]$
t_c	Time of coalescence	$U[0, 1]$ Sidereal days
α	Right ascension of the source	$U[0, 2\pi] + \text{EM}$
δ	Declination of the source	$\cos(\delta)$ for $\delta \in [-\frac{\pi}{2}, \frac{\pi}{2}] + \text{EM}$
ψ	Polarization angle	$U[0, \pi]$

Table 2.1: Gravitational wave parameters with their descriptions and the priors from which they are sampled. Where EM is present, some of the parameters are sampled from the EM catalogue as well.

and $F_\times$ is expressed as

$$F_\times = \frac{1}{2}\left[(1+\cos^2\theta)\cos 2\phi \sin 2\psi - 2\cos\theta \sin 2\phi \cos 2\psi\right], \tag{2.54}$$

where longitudinal coordinate ϕ is related to the right ascension as $\alpha = \phi + \text{GMST}$, where GMST stands for the Greenwich Mean Sidereal Time of the signal's arrival, and the latitudinal component is related to the declination as $\delta = \pi/2 - \theta$ [122]. A visual representation of the antenna response function is depicted in Fig. 2.7, where the antenna response functions for the H1, L1, and V1 detectors are plotted over the sky. This figure highlights regions of high and low response, indicating areas with strong and weak sensitivity, respectively.

The strain data parameters are summarised in Table 2.1. The first column lists the parameter symbols, the second explains the parameters, and the third outlines the prior distribution assumption on the GW parameters. The parameters $D_{\rm L}$, α, and δ incorporate EM data from the galaxy catalogue. The other parameters are derived from priors consistent with existing literature, such as Refs. [87, 99, 123].

Depending on the observing run for which we are generating data, the operational durations of the detectors vary significantly, affecting the potential detection times. For instance, the O1 observing run lasted for four months, O2 spanned from January to November 2016, and O3a and O3b extended from April 2019 until March 2020 [124]. Sampling timestamps across these entire periods for data generation would be highly inefficient. Instead, we restrict the coalescence time of arrival parameter to a range between 0 and 86164.0905 seconds, corresponding to one sidereal day. This approach is based on the assumption that every sidereal day is identical if the PSD is constant. By considering only a single day, we simplify the computational process without impacting the accuracy of our simulations, as extending the time modulation beyond one day would be analogous to extending the right ascension over multiple years.

It's important to highlight that the mass parameters, m_1 and m_2 follow the power law plus peak (PLP) distribution. The PLP distribution used for the source frame masses m_1 and m_2 combines a power-law component, which captures the majority of the mass distribution, with a Gaussian peak that accounts for a possible excess of events around a certain mass scale [125]. Selection effects play a significant role in GW astronomy, where the detectability of events depends on the properties of the source. Specifically, detectors are more sensitive to higher mass systems due to the stronger GW signals they produce. This means that the observed distribution of masses is influenced not only by the underlying astrophysical population but also by the sensitivity of the detectors.

In the next section, we will delve deeper into the specifics of the PLP distribution, explaining its parameters and how it models the mass distribution of binary systems observed by GW detectors.

2.5.1 Power Law Plus Peak Model for BBH Mergers

The PLP distribution is a phenomenological population model, which accurately describes the mass distribution of binary black hole mergers detected in the first, second, and third observing runs completed by Advanced LIGO and Advanced Virgo [87, 99, 125]. This mass model is the same one used in Ref. [16].

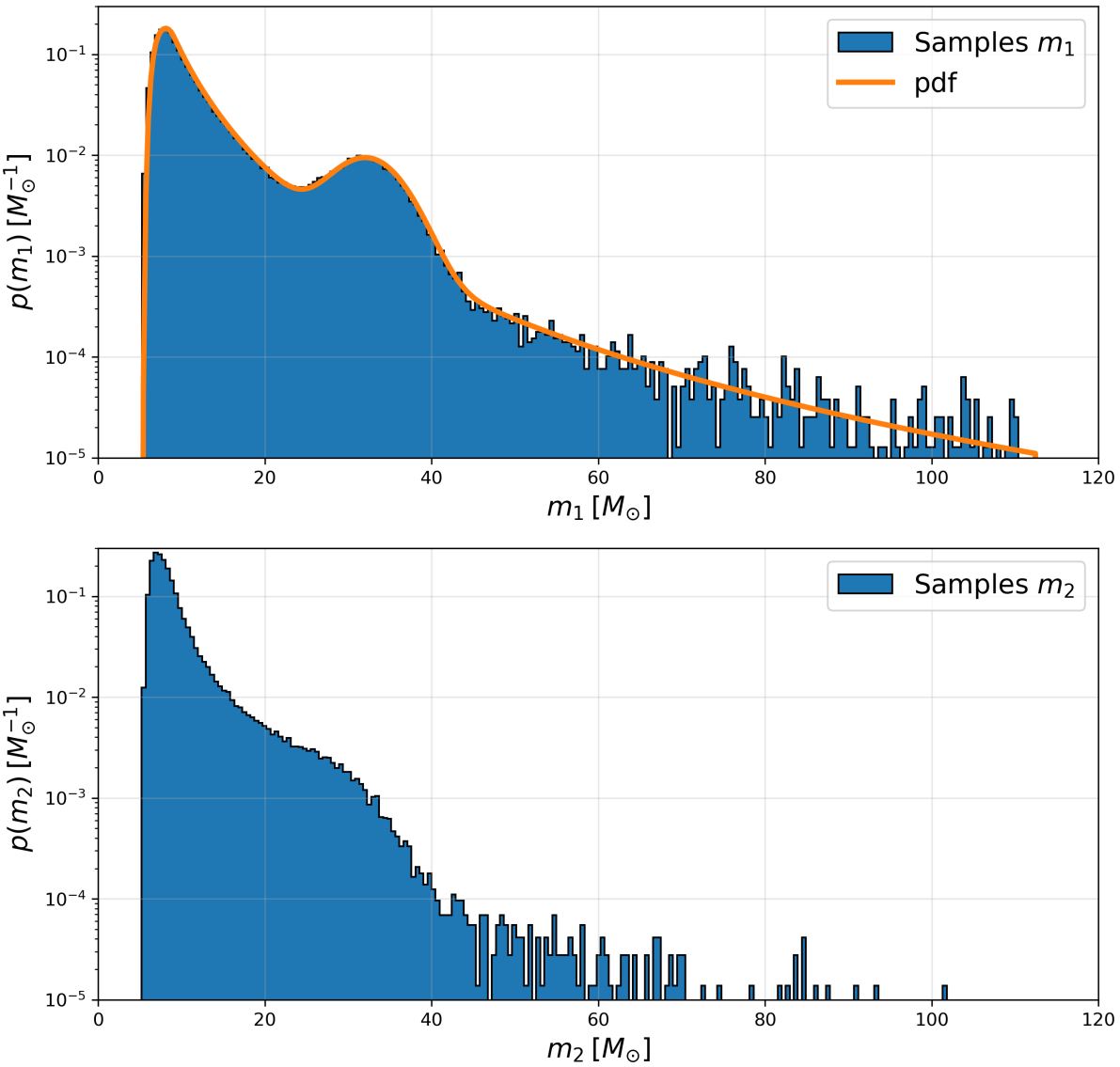

Figure 2.8: (Top): 1D distribution of primary masses from the PLP distribution with the probability density function overlaid using Eq. (2.55). (Bottom): Distribution of secondary masses, m_2, sampled from the conditional probability $p(m_2)$, marginalised over all m_1 sampled from the top plot, as described in Eq. (2.58). Both plots show the histograms of 150.000 samples. The hyperparameters used are $\alpha = 3.78$, $\beta = 0.81$, $m_{\max} = 112.5M_\odot$, $m_{\min} = 4.98M_\odot$, $\delta_m = 4.8M_\odot$, $\mu_g = 32.27M_\odot$, $\sigma_g = 3.88M_\odot$, and $\lambda_g = 0.03$.

The distribution consists of a power-law component, which describes the high frequency of low-mass black holes in the GWTC-3 catalogue [99], and a Gaussian peak designed to capture a potential accumulation of high-mass black holes resulting from pulsational pair-instability supernovae [125, 126]. This process refers to a particular type of supernova event that occurs in very massive stars, typically with initial masses ranging from about 70 to 140 solar masses. In these events, the core of the star contracts, ignites burning of elements like oxygen or silicon, expands and cools, and then contracts again to reignite. Eventually, the core's mass stabilizes, typically within a narrow range of about 35 to 50 solar masses [126]. A third component of the distribution is a smoothing function, which

accounts for factors like metallicity that may obscure the boundaries of the lower mass gap, a hypothesized range in the black hole mass spectrum between the most massive neutron stars and the least massive black holes, if such a gap exists [125]. The mathematical formulation of this distribution is as follows:

$$p(m_1|\vec{\theta}) = (1-\lambda_g)A(\theta)m_1^{-\alpha}\Theta(m_{\max}-m_1) + \lambda_g B(\theta)\exp\left(-\frac{(m_1-\mu_g)^2}{2\sigma_g^2}\right)S(m_1, m_{\min}, \delta_m), \tag{2.55}$$

where $\vec{\theta}$ represents the hyperparameters describing the prior distribution in m_1, including $m_{\max}$, the maximum mass, α, the power-law index, μ_g and σ_g, the mean and standard deviation of the Gaussian component, respectively, and λ_g, the fraction of the distribution modeled by the Gaussian component. Also, Θ is the Heaviside step function, which is zero for any mass greater than the maximum mass, and $S(m_1, m_{\min}, \delta_m)$ is a smoothing function described as

$$S(m|m_{\min}, \delta_m) = \begin{cases} 0 & \text{for } m < m_{\min}; \\ [f(m-m_{\min}, \delta_m)+1]^{-1} & \text{for } m_{\min} \leq m < m_{\min}+\delta_m; \\ 1 & \text{for } m \geq m_{\min}+\delta_m; \end{cases} \tag{2.56}$$

with the f function described in Eq.(B5) in Appendix B in [87] as

$$f(m'|\delta_m) = \exp\left(\frac{\delta_m}{m'} + \frac{\delta_m}{m'-\delta_m}\right), \tag{2.57}$$

where δ_m represents the mass range over which the black hole mass spectrum transitions. The terms $A(\vec{\theta})$ and $B(\vec{\theta})$ are the normalisation factors for the two components of the distribution, ensuring the total probability density integrates to one.

Given the primary mass as defined in Eq. (2.55), we then characterise the conditional distribution for the secondary mass under the assumption that the secondary mass is always smaller. The secondary mass distribution is characterised as follows

$$p(m_2|m_1, \theta) = C(\theta)\, m_2^{\beta}\, S(m_2|m_{\min}, \delta_m) \text{ for } m_1 > m_2, \tag{2.58}$$

where β is the power law index for the secondary mass and $C(\vec{\theta})$ is the normalisation factor of the distribution. Therefore, the full mass distributions for the BBH coalescences can be described as $p(m_1, m_2|\vec{\theta}) = p(m_1|\vec{\theta})p(m_2|m_1, \vec{\theta})$, with $\vec{\theta}$ incorporating all the hyperparameters of the joint distributions. The hyperparameters are: $\alpha = 3.78$, $\beta = 0.81$, $m_{\max} = 112.5M_\odot$, $m_{\min} = 4.98M_\odot$, $\delta_m = 4.8M_\odot$, $\mu_g = 32.27M_\odot$, $\sigma_g = 3.88M_\odot$, and $\lambda_g = 0.03$, which are the same values used in Ref. [16].

2.5.2 A note on sampling with multiple CDFs

The `CosmoFlow` analysis relies heavily on the ability to efficiently sample from multiple distributions, each defined by a set of hyperparameters. These hyperparameters can be adjusted to explore different astrophysical scenarios and their impacts on the results. For instance, hyperparameters might influence the shape, scale, and location of the distribution, reflecting various underlying physical processes or conditions. Sampling from these distributions with varying hyperparameters is essential because it enables the generation of diverse data sets that represent a broad range of possible situations. For example, in our data generation approach, we might vary the hyperparameters described in Sec.2.5.1. This allows us to sample primary and secondary source masses corresponding to each set of hyperparameters, where each set is drawn from a prior distribution.

To do so, the inverse cumulative sampling method [127] plays a crucial role in this process. This technique uses the PDF to derive the CDF through integration. Then, by interpolating the CDF, it is possible to map samples in $U[0,1]$ (the domain of the CDF) to the distribution's random variables. This allows consistent sampling from the PDF, as long as the interpolation is accurate. However, adapting the CDF for each hyperparameter set can be computationally demanding. Vectorising this process by pre-computing and chaining the CDFs for each PDF set can enhance efficiency. This method involves constructing a series of CDFs, each corresponding to a different set of hyper-parameters, $\vec{\lambda} = [\lambda_0, \lambda_1, ..., \lambda_{N-1}]$, where $\vec{\lambda}$ represents the set of hyperparameters describing the PDFs.

We chain together a sequence of CDFs, each one specific to a particular hyperparameter set. This approach forces individual samples to be drawn between integer ranges, effectively interpolating across the different CDFs. By doing so, we efficiently sample from all the separate CDFs within a single interpolation scheme. To model the process for N random variables, each associated with a unique set of hyperparameters, we compute the CDF for each. This computation can be vectorised for computational efficiency. We then concatenate these CDFs, adding one multiplied by the index of the hyper-parameter.

The final combined CDF, $F(N_x)$, where N_x is the sequential element value after repeating the X values from each CDF defined (i.e., for two CDFs, each defined over a range of L elements of the same X axis, then the range of N_x is $[0, 2L]$), is constructed by sequentially concatenating each individual CDF while adding a corresponding incremental unit value. With each CDF defined over a range of X values, we can then define a function $F(N_x)$.

The expression is given by:

$$F(N_x) = \mathrm{CDF}_{i=\lfloor \frac{N_x}{L} \rfloor}(X_{j=N_x \bmod L} | \lambda_{i=\lfloor \frac{N_x}{L} \rfloor}) + \left\lfloor \frac{N_x}{L} \right\rfloor, \tag{2.59}$$

where $i = \lfloor \frac{N_x}{L} \rfloor$ is the ith CDF and $j = N_x \bmod L$ is the jth element within the ith X range of values.

From Eq. (2.59), the new function $F(N_x)$ undergoes interpolation, enabling efficient sampling. The process involves concatenating the individual CDFs into a single, continuous function. Then we sample from the i'th CDF by inputting a CDF value in the range $[i, i+1]$, where each range $[i, i+1]$ corresponds to a different CDF. This approach allows us to draw samples from all the individual CDFs within a single interpolation of a combined CDF, ensuring a seamless and efficient sampling process.

Algorithm 1: draw_snake function (Pseudocode)

Input: Nsamples, cdfs, x_arr
Output: Nsamples from each CDF

```
def draw_snake(Nsamples, cdfs, x_arr):
    N = number of CDFs in cdfs
    cdfs_snake = horizontally stack cdfs
    xlist = convert x_arr to list
    xlist = N * xlist
    x_array = convert xlist to array
    cdfs_snake = cdfs_snake + incremental unit values for each CDF
    t = random values between 0 and 1 + incremental units for each CDF
    return interpolate(t, cdfs_snake, x_array)
```

A representation of the Python code used to enable this method is presented in Algorithm 1, where the entire process is described line by line, using only the well known mathematical package `NumPy` [128]. This method streamlines the process of sampling across multiple distributions, each characterised by unique hyperparameters, facilitating efficient analysis across different distributions.

We demonstrate a simple application of this method using the following example. Consider five Gaussian distributions from which we intend to sample multiple times. We begin by computing the CDFs for each distribution, as illustrated in Fig. 2.9. Subsequently, we concatenate these CDFs and combine each with an incremental unit vector to form a *snake-like* function, $F(N_x)$, depicted in the bottom left plot in Fig. 2.9, where it shows the concatenated CDFs spanning 5000 elements, with 1000 array elements allocated to each function. As described in Algorithm 1, by interpolating the snake-like function and sampling accordingly between integer step ranges of $[i, i+1]$, we effectively vectorise the

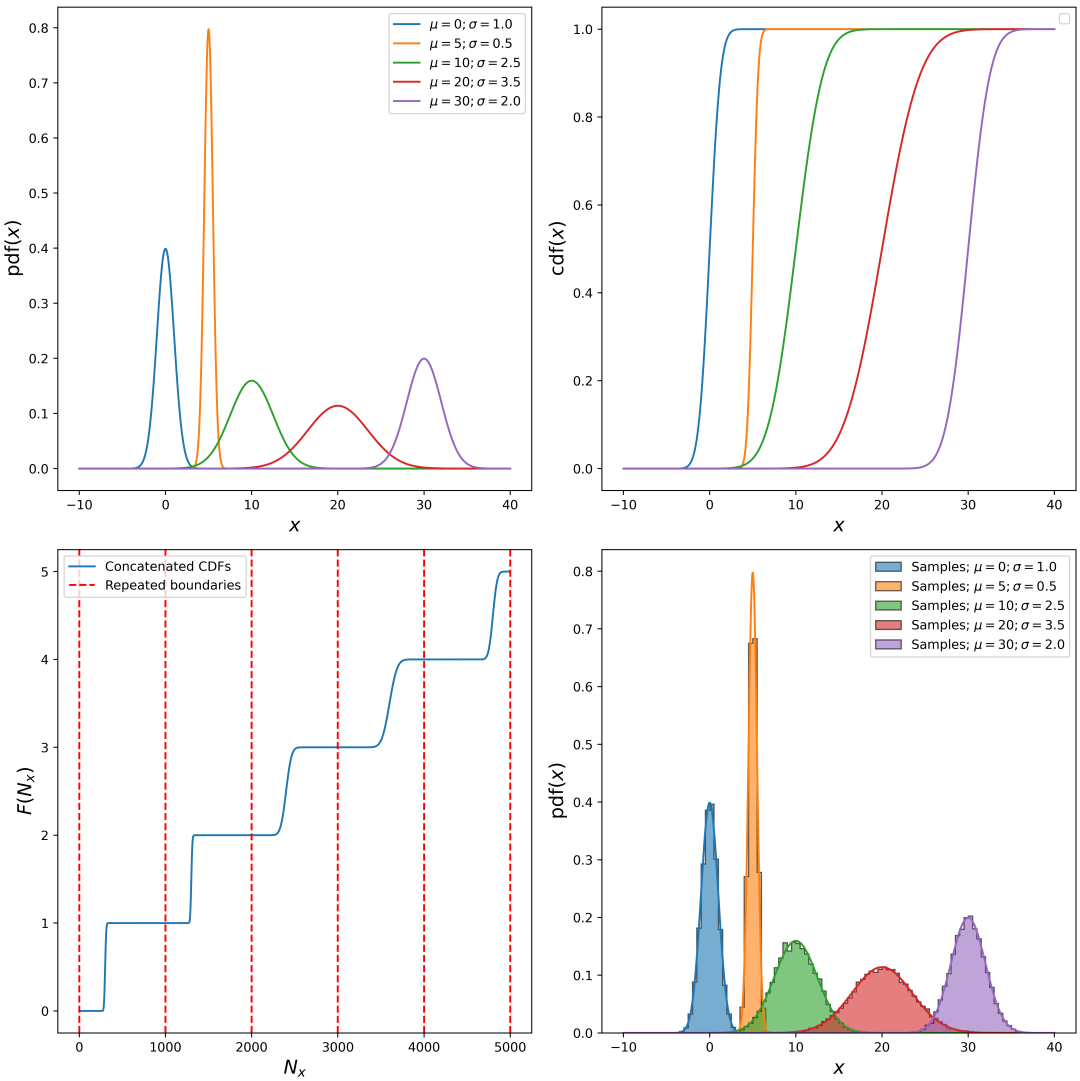

Figure 2.9: (Top Left): PDFs of five different Gaussian distributions, each with its corresponding mean (μ) and standard deviation (σ). (Top Right): CDFs computed from each of the Gaussian distributions. (Bottom Left): Concatenated CDFs from the top right plot, with the x-axis representing the number of elements in each CDF array and red dashed lines indicating the junctions between CDFs. (Bottom Right): Histogram of 10000 samples drawn from each distribution using the 'snake' method, overlaid with the combined total PDF from each Gaussian.

sampling process across multiple distinct distributions. The bottom right plot of Fig. 2.9 illustrates the distributions of the Gaussians from the top left plot, each sampled 10,000 times. The results shows the feasibility of drawing samples from each PDF, enabling the sampling from different realizations of hyperparameter sets.

This sampling technique, named *snakes*, is integrated into the `CosmoFlow` analysis to conduct sampling routines across various realizations of 1D probability distributions. We employ this technique to sample from Eq. (2.58), where for each sampled m_1, a different PDF is computed and sampled to obtain the m_2 random variable. In the final chapter, we will present the results of the data generation process, where all cosmological and population parameters are varied. This technique of sampling from multiple different distributions is extensively used in that process.

2.6 Selecting detected GW events

2.6.1 Signal-To-Noise Ratio calculation and MLPs

After determining the host galaxy locations for the GW events and sampling their intrinsic parameters from specified priors, a key step is assessing the detectability of each synthetic event. This is achieved by applying an SNR threshold, denoted as $\rho_{\rm th}$. Events whose matched-filter SNR, ρ, exceeds this threshold are considered detectable. Specifically, we perform an incoherent combination of SNR values from multiple detectors, similar to the coincident analysis used in the primary all-sky search pipelines [129].

Each detector SNR is estimated using the `bilby` package, which calculates it for each event via Eq. (1.28). Subsequently, the SNR values from each detector are incoherently combined quadratically to yield the full network SNR value, as

$$\rho_{\rm network} = \sqrt{\sum_i \rho_i^2}, \tag{2.60}$$

with i corresponding to the ith detector. Events that exceed $\rho_{\rm th}$ are retained for further analysis, while non-detectable events are discarded. This methodology is consistent with the approach used by `gwcosmo` as detailed in [16, 51], incorporating GW selection effects related to detectability.

Events that are discarded undergo resampling for all parameters except the Hubble constant, ensuring that this parameter is reused until a detectable event is confirmed. This process, illustrated in the flowchart shown in Fig. 2.3, preserves the detectable events and their corresponding Hubble constant values. By keeping the Hubble constant unchanged for non-detectable events, while resampling the other parameters, we ensure that the prior distribution on H_0 (the distribution of H_0 after marginalising over all other parameters) remains consistent with the initially chosen prior distribution.

However, the SNR calculation performed by `bilby` is not inherently optimised for vectorization, which can slow down the sampling and evaluation of SNR for synthetically generated events. While some optimisations might be possible, the current implementation does present challenges in computational efficiency. During the data generation process, in our search for detectable events from the prior parameter space, it is highly unlikely that each sample we draw will surpass the SNR threshold. Consequently, many of the events for which we compute the SNR will be discarded, further slowing down the data generation process. With an average computational time of approximately 0.1 seconds to compute the associated SNR for each set of parameters from each detector, the total computational time for the data generation process would extend into months.

To overcome this bottleneck, a machine learning approach has been implemented. Specifically, an MLP model has been implemented with 13 input parameters, excluding the phase ϕ which does not influence the SNR calculation. The luminosity distance is not directly used as an input but rather appears as part of the output of the MLP, which produces the product $\rho \times D_{\rm L}$. Analytically, this formulation allows us to reduce the dimensionality of the problem by leveraging the inverse relationship between SNR and distance. Thus, we define a new quantity, ξ, as:

$$\xi = \rho \times D_{\rm L}. \tag{2.61}$$

This approach effectively reduces the dimensionality of the output while capturing the relationship between optimal SNR (ρ) and luminosity distance ($D_{\rm L}$). Thus, to reduce the dimensionality of the problem that the MLP must learn, the network is designed to output three separate values: specifically, the product of the optimal SNR and the luminosity distance for each of the three detectors, H1, L1, and V1. We define this product as the variable ξ_i for each detector i, simplifying the modeling process by allowing the network to directly predict ξ_i instead of separate parameters. The three outputs ξ_1, ξ_2, and ξ_3

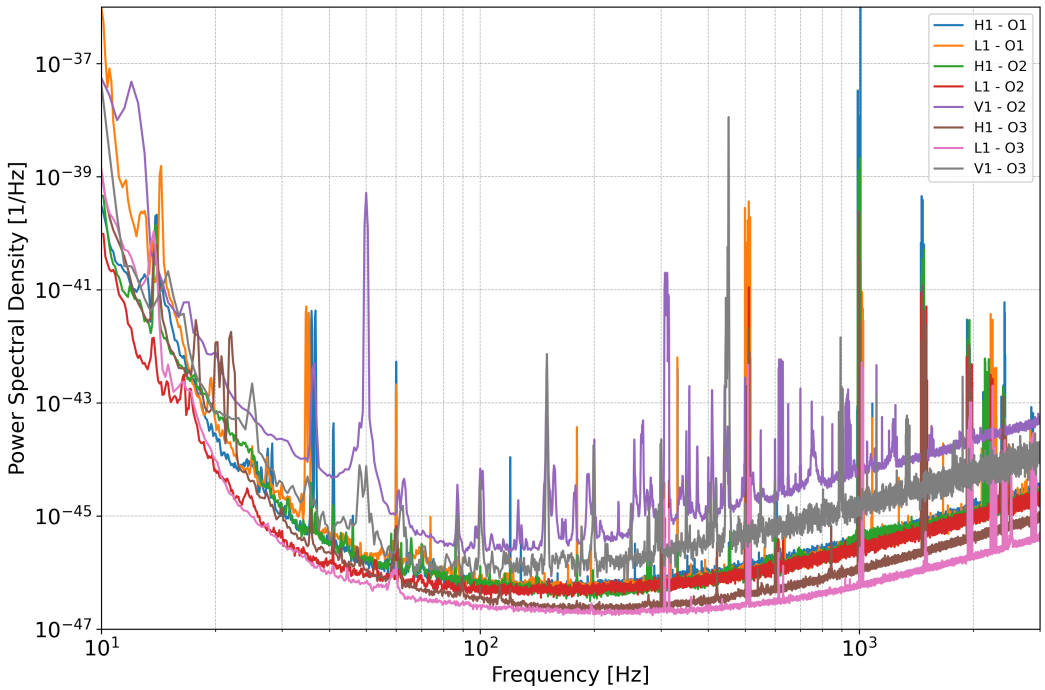

Figure 2.10: Time-averaged PSD curves for O1, O2, and O3 observing runs for each GW detector, adapted from [131]. These curves represent the sensitivity of the detectors across different runs, showing the noise performance and the frequency-dependent sensitivity limits of the instruments during each observation period.

correspond to the predicted values for each of the detectors in the network. From these outputs, the individual SNR for each detector can be calculated by dividing ξ_i by the luminosity distance D_{L}. The total network SNR is then computed by combining these individual SNR values. The MLP was implemented using the `poplar` package [130].

In this analysis, three MLPs were trained, one for each detector run: O1, O2, and O3. To align our analysis with the `gwcosmo` analysis, we used the same sensitivity curves provided in their repository Refs. [83, 92]. Each sensitivity curve corresponds to a PSD file for each detector in each run. The time averaged sensitivity curves were taken from Ref. [131] and are plotted in Fig. 2.10. Using these sensitivity curves ensures consistency with the `gwcosmo` analysis and accurately reflects the noise characteristics of each detector during the corresponding runs. The primary change in input for each MLP is the PSD, which directly influences the SNR by modifying the noise levels. By assuming that the PSDs are constant throughout each run, we simplify the training process of the MLPs while ensuring that the network can effectively model the SNR calculations for each specific detector and observing run.

The `bilby` waveform generator [100] is then configured to use the waveform approximant IMRPhenomXPHM, the same approximator employed in [16]. This waveform represents a phenomenological model that describes the GW signals from quasi-circular, precessing binary black hole systems in the frequency domain. Notably, IMRPhenomXPHM extends the analysis beyond the dominant quadrupole to incorporate higher multipoles within the precessing frame [132].

To begin the process of generating data for the MLP, we start by setting the sampling frequency and minimum frequency at $4096\,\mathrm{Hz}$ and $20\,\mathrm{Hz}$ respectively. The first is important for determining the time resolution of the waveform and directly influences the highest frequency component that can be accurately captured by the Nyquist frequency [133], which is half the sampling rate. The second setting defines the minimum frequency from which to start the waveform analysis, thus considering only frequency contributions above this threshold.

Since the MLP requires training on every realization of events, both detected and undetected, prior bounds were placed on the GW parameters, but no further constraints were applied within those bounds. This approach ensures uniform performance across the entire parameter space defined by the prior bounds. We did not attempt to optimise performance within specific regions of the parameter space, allowing the MLP to generalize well across all possible GW events within the defined bounds.

The parameters were sampled from the same priors as described in Tab. 2.1 (excluding EM information), except for the detector frame mass priors, which were set to a uniform distribution between $U[4,350]\,M_{\odot}$. This range adequately covers the source masses from $[5\,M_{\odot}, 112.5\,M_{\odot}]$ when multiplied by the $(1+z)$ factors, appropriate for detector frame masses. Additionally, the prior on the luminosity distance was set to uniform between $U[10,11000]\,\mathrm{Mpc}$. Adopting these priors allows the MLP to be optimised consistently across the defined regions of the GW parameter space. Both the input and target output data are scaled using Z-score normalisation, which involves subtracting the mean and dividing by the standard deviation of each feature, as

$$z = \frac{x-\mu}{\sigma}, \tag{2.62}$$

where μ is the mean of the feature values and σ is the standard deviation of it. In this case, the features are the GW parameters. This rescaling ensures that the learning algorithm treats each feature equally, facilitates faster convergence, and improves overall performance by preventing features with larger ranges from dominating the learning process. It is important to note that the choice of priors is an important aspect of the modeling process. While uniform priors are commonly used, they are not always the most general or uninformative choice.

For each observing run, the training dataset for the MLP consisted of one million data points. Each data point included 13 features representing the GW parameters, and the target data was ξ, for each detector. Therefore, the dataset comprised 13 input features and 3 target columns (the ξ values), resulting in a total of 16 columns and one million rows. The parameter samples were then drawn from their priors, then used with `bilby`, employing the PSD corresponding to each detector, to estimate the optimal SNR. The MLP underwent training with 90% of the data, with 128 neurons across 8 layers. The ReLU (Rectified Linear Unit) activation functions were used in each layer. ReLU is an activation function defined as $f(x) = \max(0,x)$, which outputs the input directly if it is positive; otherwise, it outputs zero. ReLU introduces non-linearity to the network, allowing it to learn complex patterns. It is widely used in NNs due to its simplicity and computational efficiency, helping to mitigate issues like the vanishing gradient problem commonly encountered with other activation functions [134].

The training proceeded for 200,000 epochs with a learning rate of 0.001 and used MSE as the loss function. The Adam optimiser was employed for training. No early stopping criterion was used; instead, the model with the lowest validation loss at the end of training was selected. The entire training process took approximately 15 hours using an RTX GeForce 2080 graphics processing unit (GPU). The loss trends for both the training and validation datasets are displayed in Fig. 2.11, where clear reductions in loss values are observed, indicating that the model increasingly generalizes better to the data. However, as the loss is still decreasing towards the end of the training, minor improvements could potentially be achieved with additional training.

To evaluate the network's performance, 10% of the data, which was not seen by the network during training, was used as validation data. The results are depicted in Fig. 2.12. The figure illustrates the overall accuracy of the MLP model in predicting SNR values. The top left plot of Fig. 2.12 shows the predicted versus true SNR values, with a color bar representing the absolute magnitude of the residuals. The bottom left plot of Fig. 2.12 displays the residuals against the true SNR values, indicating that the magnitude of

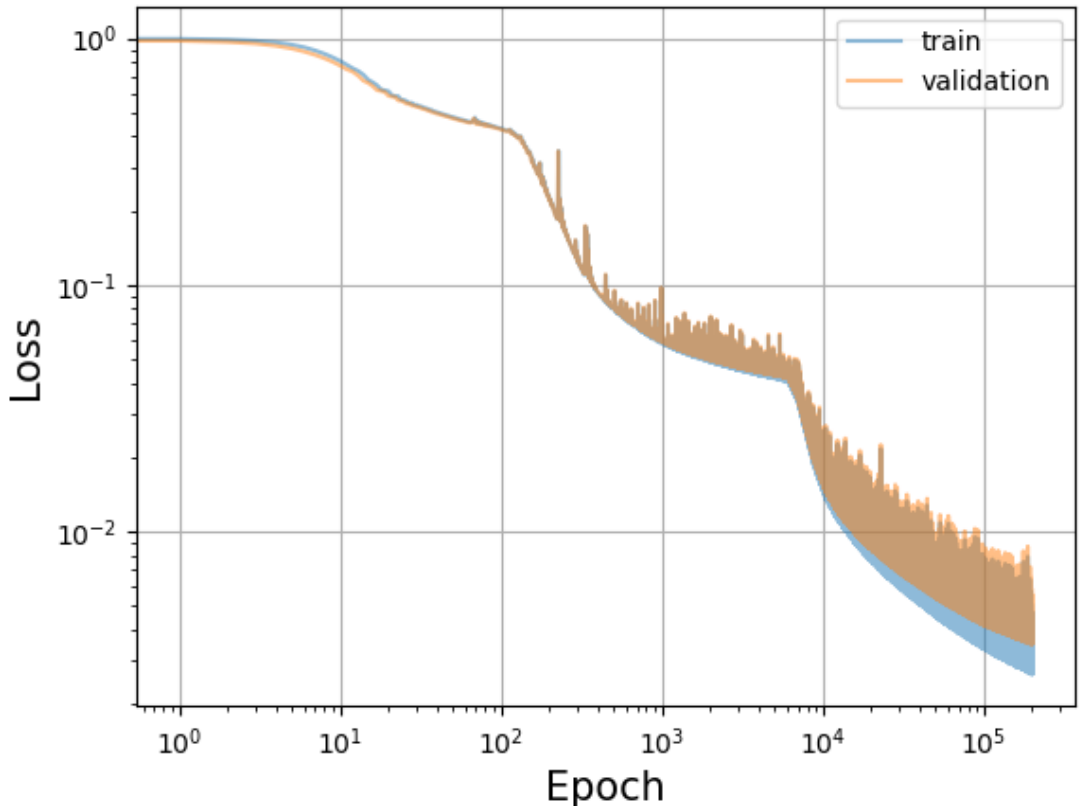

Figure 2.11: Loss values of the training and validation data as a function of training epoch.

errors increases with higher SNR. This increase in error magnitude is primarily due to the distribution of SNR values in the training data, which contains fewer data points at higher SNR levels, as shown in Fig. 2.13. Consequently, the MLP is less exposed to high SNR cases during training, leading to diminished performance and less accurate predictions in these regions. It is important to note that this trend indicates the magnitude of the errors and not a bias, as we are plotting the absolute residuals. The right plot of Fig. 2.12 presents a histogram of the residuals, highlighting the distribution and frequency of prediction errors, showing no obvious biases.

We employed this method to train a total of three MLPs, one for each detection run from O1 to O3. Although it is possible to train a single MLP on data from all detection runs, this would require adding extra features to the input data. Specifically, we would need to include features for the three detectors (H1,L1,V1) and the observing run (O1, O2, or O3). This would increase the number of input features from 13 to 19, with three binary features for the detectors and three for the observing runs. While this could streamline

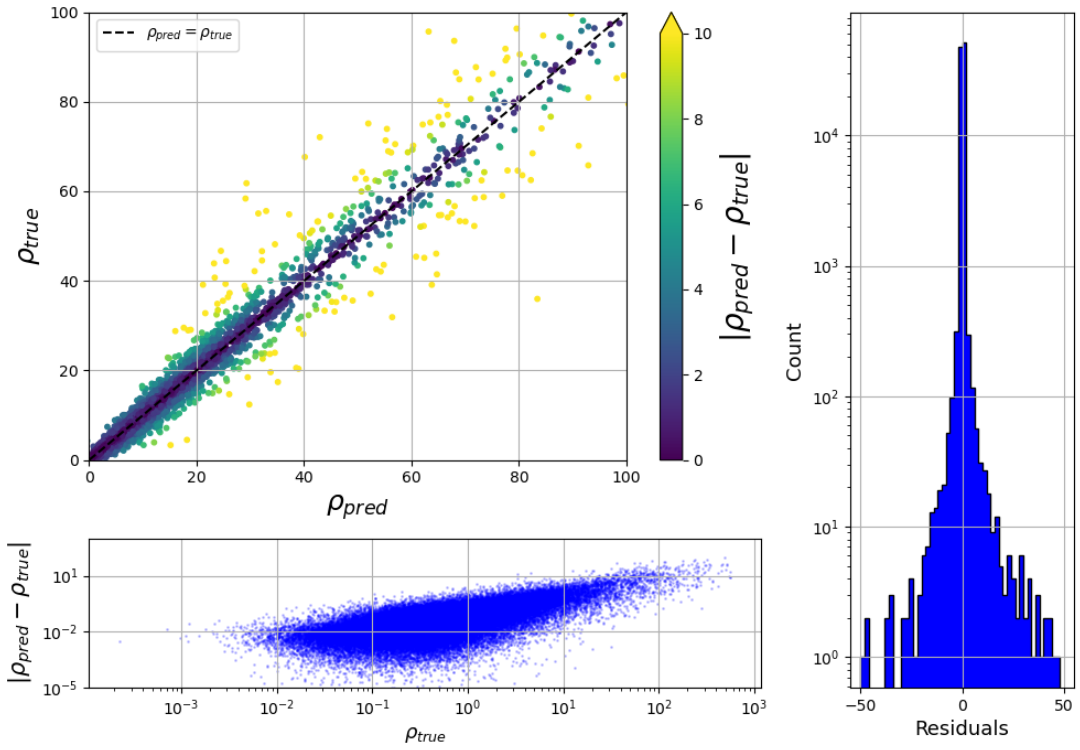

Figure 2.12: (Top left): Scatter plot of the true SNR against the predicted SNR from the MLP model for the H1 dataset. The color scale represents the absolute residuals $|\rho_{\text{pred}} - \rho_{\text{true}}|$, indicating the error magnitude. The dashed line represents the ideal case where the predicted SNR equals the true SNR. (Bottom left): Scatter plot of the absolute residuals $|\rho_{\text{pred}} - \rho_{\text{true}}|$ against the true SNR values on a log-log scale, showing how the prediction errors vary with the true SNR. (Right): Histogram of the residuals $|\rho_{\text{pred}} - \rho_{\text{true}}|$, illustrating the distribution and frequency of prediction errors. The histogram is plotted on a logarithmic scale to highlight the distribution's tails. All results are specific to the H1 dataset.

the training process, it would also make the input data more complex and increase the training time. Therefore, we chose to train separate MLPs for each detection run to keep things simple and reduce the computational burden. The results from the trained MLP with O1 and O2 PSDs are found in Appendix A.

Within the data generation code, we use the trained MLP to estimate the optimal SNR for each generated event and retain the optimal SNRs from the specific detectors chosen for use during the data generation process. The optimal SNR values from each detector are then combined quadratically to obtain the detector network's optimal SNR. To add realism to the observed SNR for these GW events, the optimal detector network SNR is used as the non-central parameter of a χ^2-distribution with $k = 2n$ degrees of freedom, where n is the number of detectors used. We then sample from this distribution into obtain the matched-filtered network SNR. The distribution is defined in Eq. (2.14), with a threshold of $\rho_{\text{th}} = 11$.

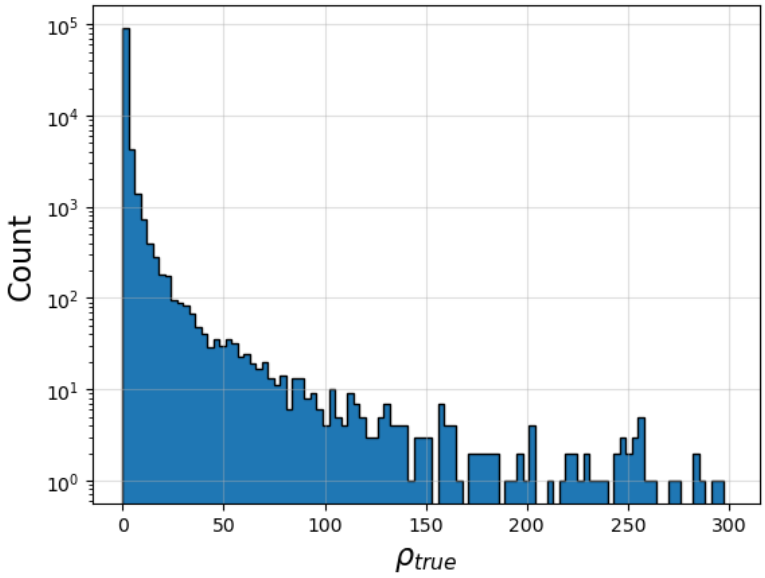

Figure 2.13: SNR distribution of the validation data for the H1 detector. This is from the O3 H1 SNR values, showcasing the distribution and tail of the SNR values.

For the purposes of this analysis, obtaining accurate estimates of the optimal SNR is very important. To quantify the accuracy of the MLP predictions, we analyse the error distribution. The bottom left plot of Fig. 2.12 shows that the residuals, or errors, between the predicted and true SNR values range from 0.01 to 10 units, with a mean error of approximately 1 unit of SNR. Having a mean error of 1 unit is acceptable for our analysis, particularly given the SNR detection threshold of 11. This level of error demonstrates that the MLP operates reliably within the margin of uncertainty associated with the detection criterion, effectively distinguishing between detected and non-detected events. While achieving perfect accuracy is ideal, the current mean error demonstrates the model's robustness and effectiveness in practical applications. Future work will look into further reducing this error by incorporating more data, refining the model, and potentially using more sophisticated techniques, building on the strong foundation established by this analysis. The trained MLPs significantly increases the SNR calculation speed, achieving a speed increase of $10,000\times$. This enhancement is very important for bypassing the computational bottleneck associated with SNR calculations.

2.6.2 Optimising Redshift Threshold for Maximum Detection Efficiency

The data generation code, designed to sample from the chosen priors and select only events that are detectable, is efficient yet could be enhanced by optimising the parameter space. To avoid biases while optimising the search for detectable events, it is important to carefully select the regions from which we sample. By minimising the parameter space, we can avoid regions where events are nearly impossible to be detect, thereby improving efficiency. One effective strategy is to implement a redshift threshold, which focuses the search on a constrained range. This ensures that the final set of output samples remains unbiased and representative of detectable GW events.

The maximum observable distance for GW events primarily depends on the luminosity distance and the chirp mass of the events, with additional influences from spins and sky positions. Initially, we sample GW parameters to cover the entire prior space defined in Tab. 2.1. We adopt a uniform prior for the luminosity distance, ranging from 10 to 11000 Mpc, which encompasses the entire detectable distance for GW events for O1, O2 and O3 sensitivities. The primary and secondary masses are adjusted for the detector frame mass, spanning from $4.8\,M_{\odot}$ to $337.5\,M_{\odot}$, with a maximum redshift of 2. The lower and upper mass bounds are defined by $M_{\rm min}$ and $M_{\rm max} \times (1+z)$, respectively. This choice extends the range from the smallest possible redshifted minimum mass to the highest possible redshifted maximum mass, ensuring coverage across all potential values due to redshift. For each GW parameter sample, we then evaluate the SNR and store the highest possible SNR. We then evaluated 1000 iterations, each with 10,000 samples, calculating the product of SNR and luminosity distance for each sample using the optimal setup with all detectors active, focusing on the most detectable events. This computation was expedited using the previously trained MLP. For each iteration, the maximum sampled value of the product was retained.

The distributions in Fig. 2.14 show the maximum values of the product of SNR and luminosity distance for the O1, O2, and O3 observing runs in the first row. Each histogram represents the counts of these maximum values across the 1000 iterations, with the mean and 99.99% percentile values from the total dataset, comprising 10 million data points, indicated by the red and black dashed lines, respectively. By evaluating in 1000 iterations of 10,000 samples each, we aim to capture a more representative distribution of high SNR events, marginalising over all parameters in each batch. This approach avoids relying solely on the single most extreme event, instead providing a broader view of loud events under realistic conditions. The 99.99% quantile was selected as a conservative threshold

to identify very loud events without overemphasizing outliers. This quantile captures the most significant events while avoiding extremes that could be influenced by noise or statistical anomalies. It reflects a balance between capturing rare but plausible optimal events and maintaining statistical robustness. Additionally, given the large sample size (10 million data points), the 99.99% quantile still provides a meaningful set of 10,000 events for analysis, ensuring that the results are both statistically significant and representative of realistic high SNR events. These results demonstrate that there is a range of distances within which GW events can be detected with an SNR above a defined threshold. This threshold helps to rule out high values of ξ (the product of optimal SNR and luminosity distance) that would almost certainly result in non-detections, thereby refining the detectability criteria for GW events.

To estimate this range, we can consider an event at the detection threshold $\rho_{\rm th}$ and place it at the maximum possible distance $D_{L,\rm max}$. We would expect to detect it, optimising over all other parameters. Then, by taking the same source and changing its distance, its SNR will change according to the following relation:

$$\rho_{\rm th} D_{\rm L,max}(z_{\rm max}, H_0) = \rho D_{\rm L}. \tag{2.63}$$

This equation relates the product of SNR and luminosity distance at the detection threshold to the same product at any other distance, thereby allowing us to estimate the maximum detectable distance for events given a specific SNR threshold.

In Eq. (2.63), the maximum luminosity distance, $D_{\rm L,max}(z_{\rm max}, H_0)$, on the left-hand side is a function of redshift $z_{\rm max}$ and the Hubble constant H_0. By setting $\rho_{\rm th} = 11$ in Eq. (2.63) and choosing a conservative value for the maximum $\rho \times D_{\rm L}$, we selected the 99.99% quantile from each total distribution across the three histograms, we can solve for $z_{\rm max}$ by inverting the equation, which then just becomes a function of H_0. Initially, we opted to use the mean of the maxima, but quickly realized this value was imposing a cutoff at high Hubble constants, even if minimal. Therefore, we decided to use the 99.99% quantile from the combined dataset of 10 million samples.

These quantiles, calculated from the entire datasets, correspond to values of $\rho \times D_{\rm L} = [64,859.84\,{\rm Mpc}, 89,547.08\,{\rm Mpc}, 109,010.27\,{\rm Mpc}]$ for the O1, O2, and O3 detector runs, respectively. Using these values, we determine the maximum distance at which a GW event is detectable. This distance is then converted into redshift values for a given H_0, by inverting Eq. (1.5). In practice, this approach effectively narrows down the parameter space to include only those distances where GW events are likely to be detectable, ac-

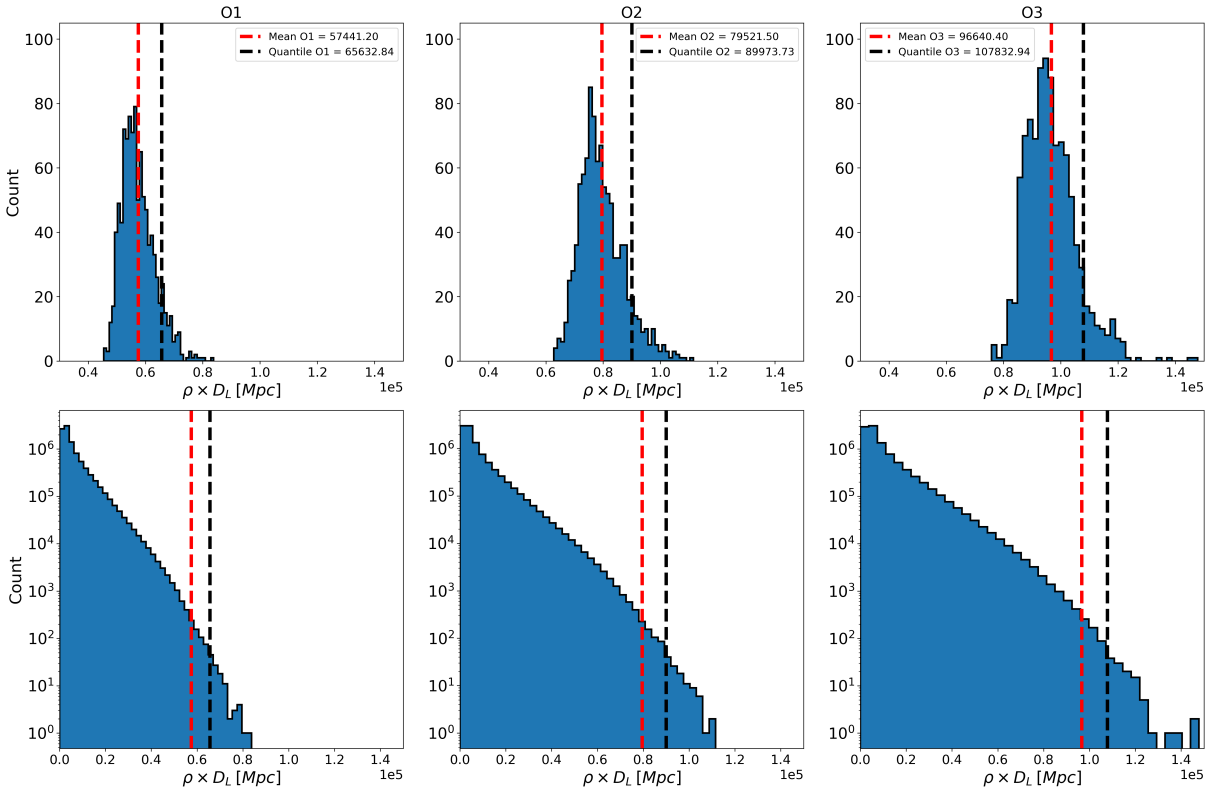

Figure 2.14: Plots showing the distributions of the maximum SNRs multiplied by the luminosity distance. The top row displays the distribution of the maximum SNRs times the luminosity distance for each batch of 10,000 samples over 1,000 iterations. In contrast, the bottom row shows the distribution for the total amount of data, encompassing all 10 million samples. The red dashed line represents the mean of the distribution of the maxima, while the black dashed line indicates the 99.99% quantile. The plots are organized sequentially with O1 on the left, O2 in the center, and O3 on the right.

counting for variations in the rate of cosmic expansion. By limiting the maximum redshift as a function of the Hubble constant in the redshift prior when sampling z, we avoid wasting time generating high-distance events that are almost certainly undetectable. Importantly, this does not alter the data generation distributions but optimises the sampling process to focus on potentially detectable events.

An example of the prior distribution, reflecting this modification, is shown in Fig. 2.15. The left plot displays samples drawn from the joint prior distribution $p(z,H_0)$, for an O3 run and $\rho_{\rm th} = 11$, highlighting the variation in sample density between regions of high and low H_0. The right plot of Fig. 2.15 shows the ratio of the comoving volume, V_C, with a fixed maximum redshift at $z_{\rm max} = 2.86$ (the maximum redshifts achieved by setting $H_0 = 140\,{\rm km\,s^{-1}\,Mpc^{-1}}$ in $z_{\rm max}(H_0)$), and a varying maximum comoving volume computed using the $z_{\rm max}(H_0)$ at each H_0 value. The ratio shows that for a specific value, say $H_0 = 25\,{\rm km\,s^{-1}\,Mpc^{-1}}$, the reduction factor in the sampled volume is $V_C/V_{C,z_{\rm max}} \approx 15$, which corresponds to a speedup of about the same value. With this modification to the

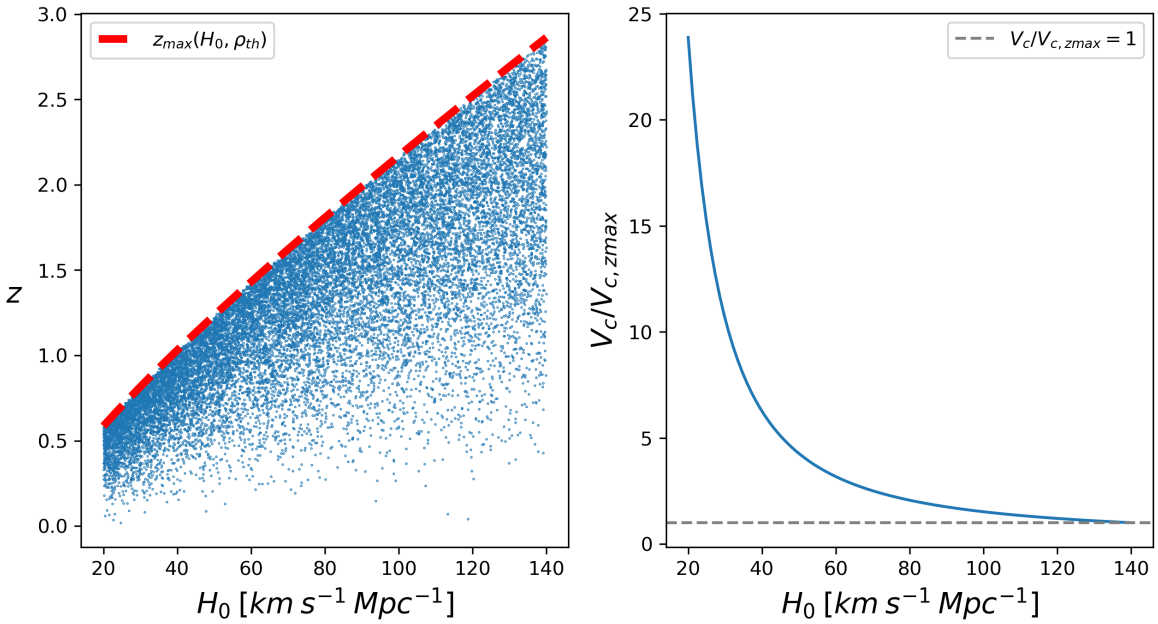

Figure 2.15: (Left): 20,000 redshift samples from the prior distribution with cutoff at z_{max} defined in Eq. (2.63), with the overlaid dashed red line showing the z_{max} boundary as a function of H_0, for an O3 event distribution and an $\rho_{\rm th} = 11$. (Right): Ratio of comoving volume between a set $z_{max} = 2.86$ for an O3 data generation process compared to the modified reduced comoving volume, $V_{c,z_{\rm max}}$, as a function of $z_{\rm max}(H_0, {\rm SNR}_{\rm th})$.

prior redshift distributions, we achieve a significant speedup in our data generation code, on average about 5 times faster. This speedup can be quantified by integrating the area under the curve in the right plot of Fig. 2.15 and then dividing over the range of H_0 the integration was performed, resulting in the average value of the function. The ratio of fixed to varying comoving volume approximately corresponds to the speedup factor (or reduction factor, as we are reducing the search volume).

2.7 Data generation results

In the previous sections, we discussed the components of the data generation process, with a focus on the SNR MLP approximator. This tool allows for rapid and vectorized calculations of SNR across multiple events simultaneously, enabling efficient data generation within practical time frames.

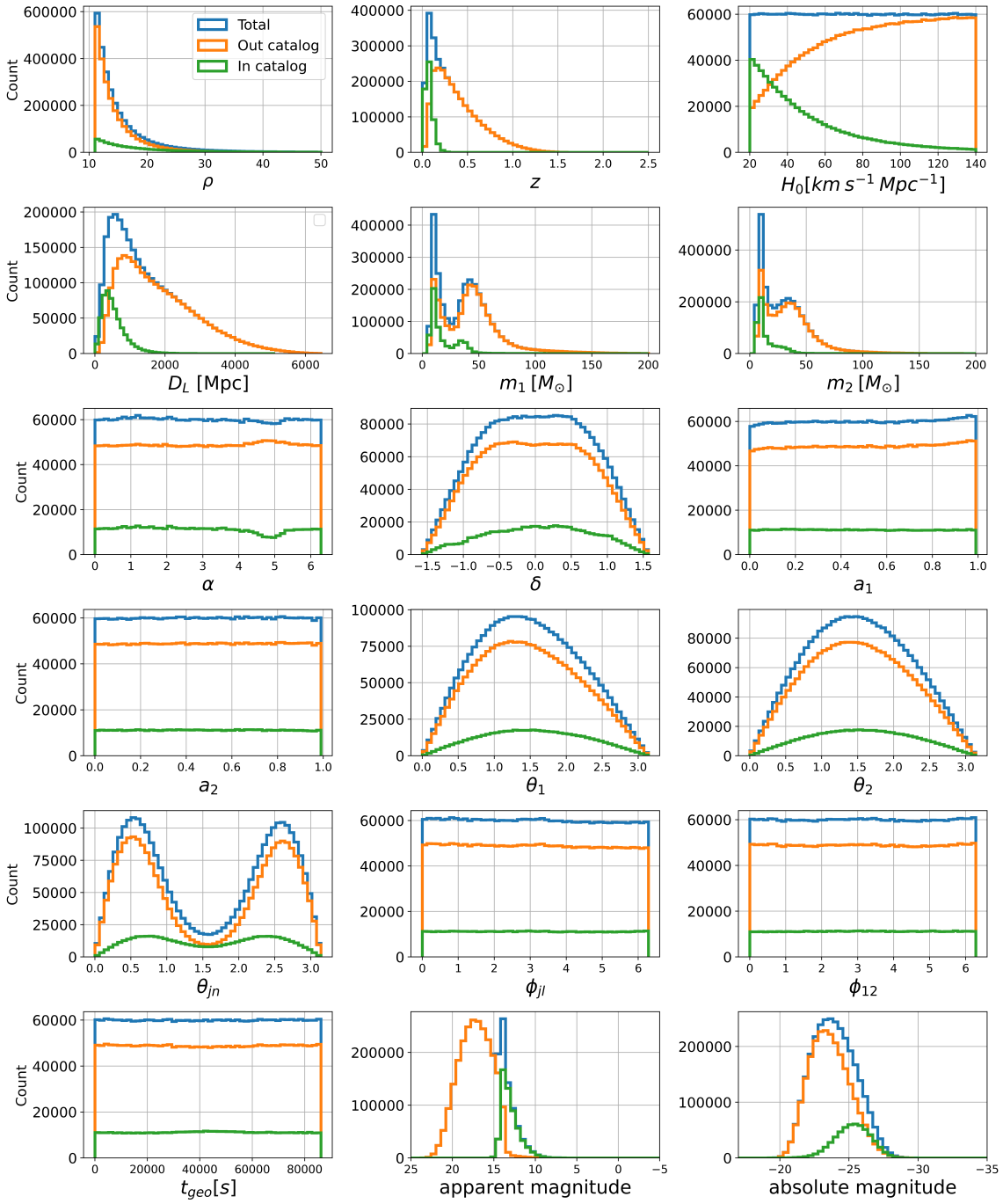

Figure 2.16: Multiple histograms show the distributions of 3 million synthetic GW source parameters associated with galaxy characteristics and cosmological parameters. These include the SNR (ρ), redshift, the Hubble constant (H_0), luminosity distance ($D_{\rm L}$), primary and secondary masses redshifted masses ($m_1(1+z)$ and $m_2(1+z)$), sky location coordinates (α, δ), spin magnitudes (a), tilt angles (θ_1, θ_2, θ_{jn}), phase angles (ϕ_{jl}, ϕ_{12}), geocentric time of arrival ($t_{\rm geo}$) and both apparent and absolute magnitudes. The color coding helps differentiate the total dataset in blue, with subsets categorized as in-catalogue (in green) and out-of-catalogue (in orange), thereby providing a clear visual distinction among the groups. This particular data set is for a O3 run sensitivity with H1, L1 and V1 detector setup.

Following the data generation framework outlined in Fig 2.3, we have successfully completed the data generation phase. The decision to sample cosmological parameters uniformly ensures consistent performance across the entire data space during the NF training, thereby preventing performance biases. The results of this process are depicted in Fig. 2.16, which displays a series of histograms detailing the distributions of GW source parameters for the H1, L1, and V1 detector setup, corresponding to LIGO Hanford, LIGO Livingston, and Virgo Pisa, respectively, using the O3 run PSD. Additionally, when generating the data, we record the origin of the simulated event, indicating whether the galaxy from which the signal was simulated is within the catalogue or outside of it. In Fig. 2.16, orange represents samples outside the catalogue, while green indicates those within the GLADE+ catalogue.

This visual comparison highlights the characteristic differences between samples in and out of the catalogue, as seen in the distribution of various histograms such as the Hubble constant. The distribution of the Hubble constant (H_0) shows that in-catalogue events are concentrated towards lower values, while out-of-catalogue events spread across a broader range. The detector frame masses of the binary components (m_1 and m_2) also show a noticeable difference, with in-catalogue events having a slightly different distribution compared to out-of-catalogue events, favouring lower masses since in-catalogue galaxies are usually closer than out-of-catalogue galaxies.

The apparent magnitude plot shows a clear difference between in-catalogue and out-of-catalogue events, with a noticeable spike at the threshold for categorizing galaxies. We do not have a definitive explanation for this phenomenon, but our best estimate is as follows: when determining the apparent magnitude of galaxies outside the catalogue, there is a smooth trend. However, in the magnitude range between 14 and 15, where we start selecting galaxies from the catalogue, this trend is interrupted by the actual observations from the catalogue. This results in a discontinuity in the overall apparent magnitude distribution. The spin magnitude parameter a_1 tends to favor higher spins due to detectability, as higher spins make events more likely to be detected. We would expect a similar effect for the secondary spin magnitude parameter a_2, but this is not observed. This discrepancy may be because we are generating synthetic BBH events where the spin of the secondary mass has less influence compared to the primary mass. The primary mass, having a higher mass, contributes more significantly to the total angular momentum. The sky location parameters exhibit interesting features. The right ascension shows distinct galaxy catalogue features in the distribution for in-catalogue events, with a noticeable dip

in the region between $4-5$ rads due to the galactic band. The declination, particularly for out-of-catalogue events, displays two distinct bulges. These bulges reflect the average antenna pattern, depicted in Fig2.7, for the three detectors (HLV) averaged over one sidereal day.

Out of all the parameters presented in Fig. 2.16, the one with the most informational impact is the redshift distribution, specifically the in-catalogue distribution. This is where the EM information (and the sky locations) come into the analysis, allowing us to incorporate EM information from the catalogue. To better understand how this information varies within the analysis, Fig. 2.17 shows the variation of redshift over 32 random pixels across the sky for different sky locations. The subplots show in-catalogue and out-of-catalogue distributions. Specifically, in the in-catalogue distribution, it is possible to observe variation over the distribution for each pixel, showcasing the variation and clustering of the catalogue in different sky directions. Other pixels, such as locations $\alpha, \delta = [(146.74, -55.87), (285.47, 14.48)]$, show no catalogue contributions, suggesting that the redshifts for those pixels are located in the galactic band.

It is also important to note that the samples in Fig. 2.16 are split in approximately 20/80 between in-catalogue and out-of-catalogue events. This significant difference is an important consequence of the selection effects. If the selection effects were not incorporated into the data generation, the dataset would be dominated by out-of-catalogue events. This dominance would prevent the network from learning the structure of the catalogue effectively, as the catalogued events provide key information about the distribution and characteristics of detectable sources.

The angles ϕ_{jl} and ϕ_{12} were unaffected by selection effects due to the detectability of the GW signals we simulate. The same applies to the geocentric time of arrival (t_{geo}), which shows a flat distribution over the span of an entire sidereal day, indicating that the marginal distribution of the arrival time of the events is uniform throughout the day. However, we know that the arrival time directly influences the type of events we observe, as it is highly correlated with α.

The plot in Fig. 2.18 illustrates the relationship between right ascension (α) and geocentric time of arrival (t_{geo}) for detected GW events. The colour scale indicates the bin count of the events, with yellow regions representing higher-density regions. The plot shows a clear periodic pattern in the distribution of events, reflecting the Earth's rotation, which affects the sensitivity of GW detectors to different parts of the sky over the course of a sidereal day (approximately 86,164 seconds). This periodicity in the high density regions with

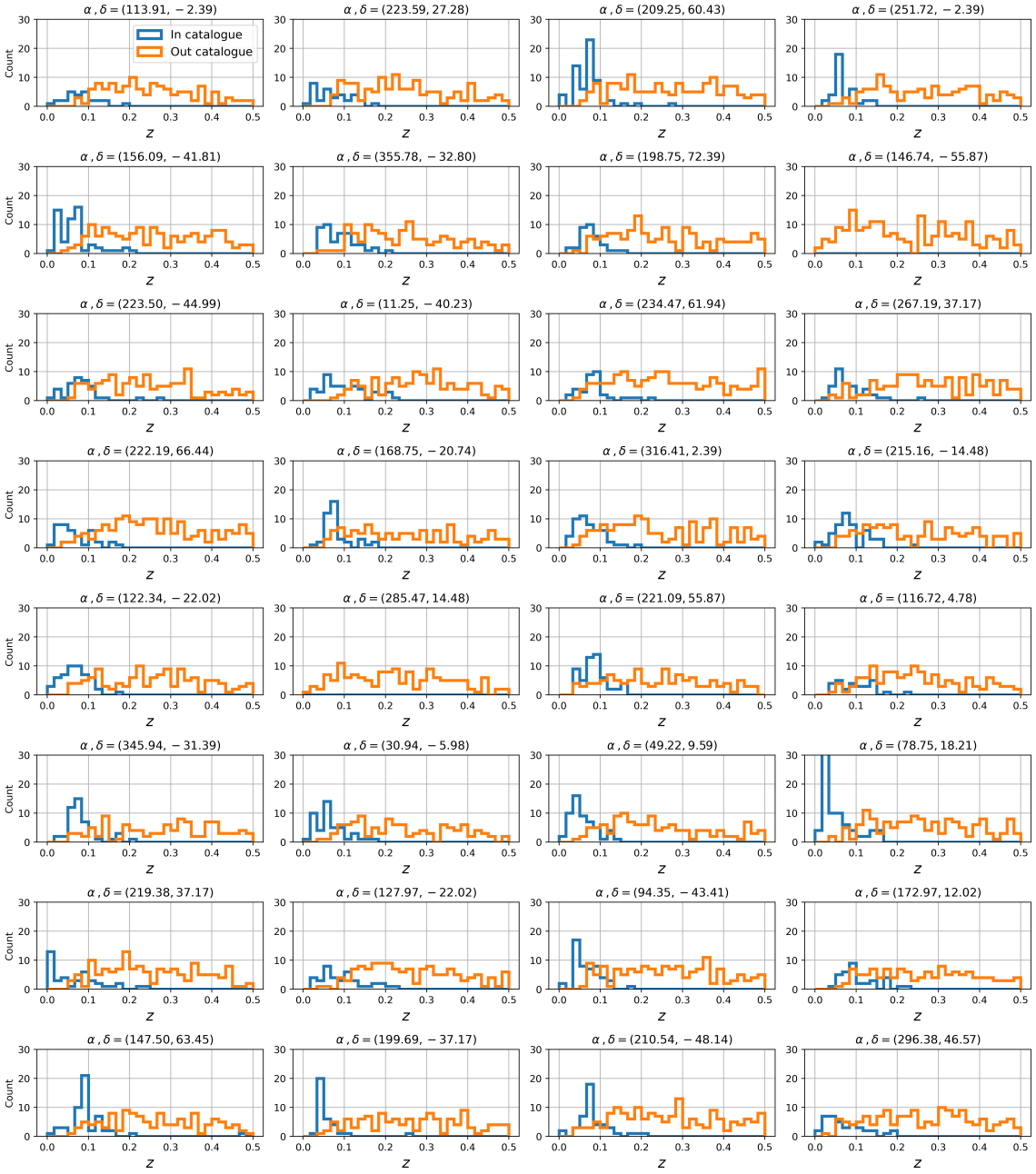

Figure 2.17: Redshift variation in 32 random pixels from a HealPy pixelated map, with NSIDE = 32, of the synthetically generated events from Fig. 2.16. Each subplot shows two histogram distributions: blue for in-catalogue redshifts and orange for out-of-catalogue redshifts. A cut-off at 0.5 was set for better visualization of the variation of the in-catalogue distribution of redshifts.

respect to right ascension (α) and geocentric time of arrival (t_{geo}) can be attributed to the antenna response patterns of the GW detectors (HLV). The antenna response of GW detectors varies depending on the direction from which the GWs are coming, as depicted in Fig. 2.7. This means that detectors are more sensitive to certain parts of the sky at specific times due to their orientation and the rotation of the Earth. Consequently, the detectors' sensitivity to different sky locations changes as the Earth rotates, leading to the observed periodic structure.

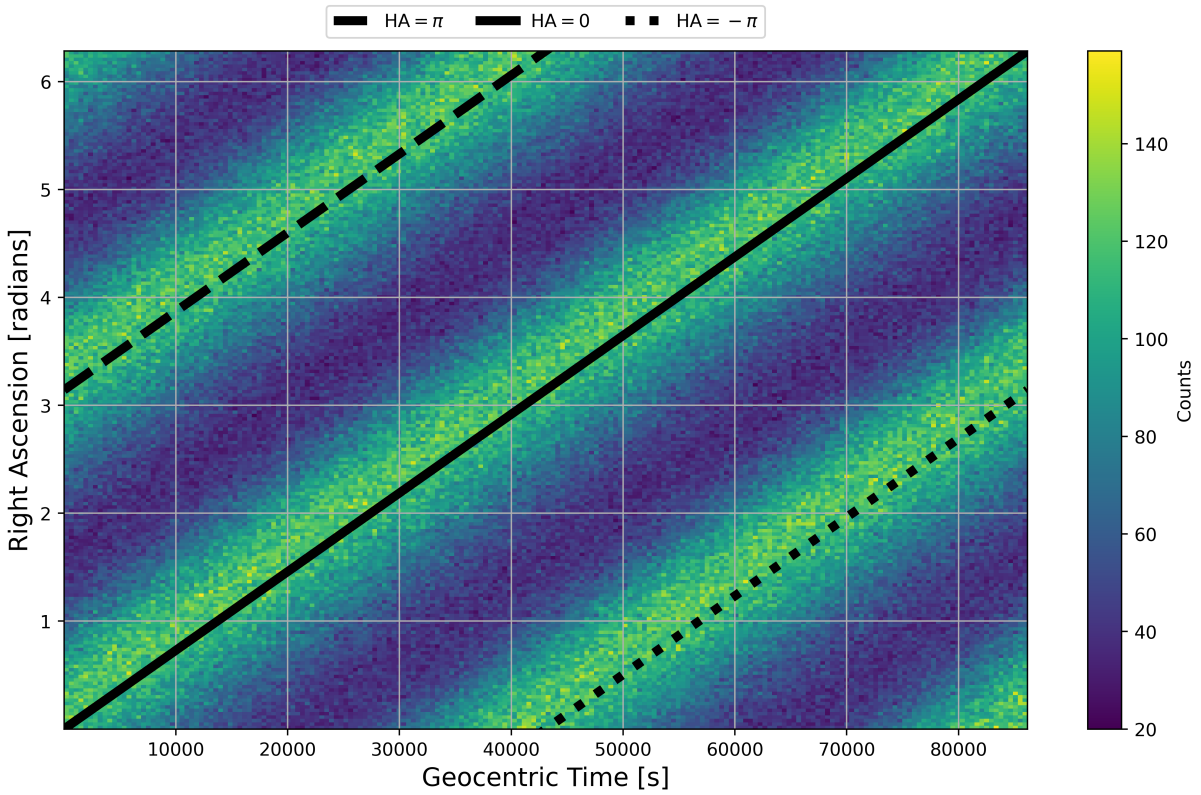

Figure 2.18: Scatter plot showing samples of right ascension (in radians) and geocentric time of arrival (in seconds). A Gaussian KDE was used to evaluate the density of each scatter point, with regions of higher-density indicated by lighter colors. Constant hour angles are also plotted at values π, 0, and $-\pi$.

We describe the Hour Angle (HA) of an event as:

$$\mathrm{HA} = \alpha - \frac{t_{\mathrm{geo}}}{239.34}\left(\frac{\pi}{180}\right), \tag{2.64}$$

where α is the right ascension in radians, t_{geo} is the geocentric time of arrival in seconds, and 239.34 is the factor that converts the length of one sidereal day (86,164 seconds) into degrees. As the Earth rotates, the sensitivity of the detectors varies with the changing hour angle, leading to regions where the probability of detecting an event is higher or lower. The straight diagonal lines in the plot represent constant hour angles, corresponding to values $-\pi$, 0, and π in Eq. (2.64). These lines appear as specific right ascensions cross the local meridian (culminate) at these Local Sidereal Time (LST) values, creating regions of higher and lower detection probabilities. When a specific region of the sky aligns with the detectors' optimal sensitivity, the probability of detecting events from that region increases, resulting in higher densities. As the Earth rotates, the same right ascension regions move into and out of the detectors' view, forming the observed diagonal bands of higher and lower density regions. This results in the periodic pattern seen in the plot, where specific regions of the sky have varying probabilities of event detection due to the changing sensitivity of the detectors over the sky.

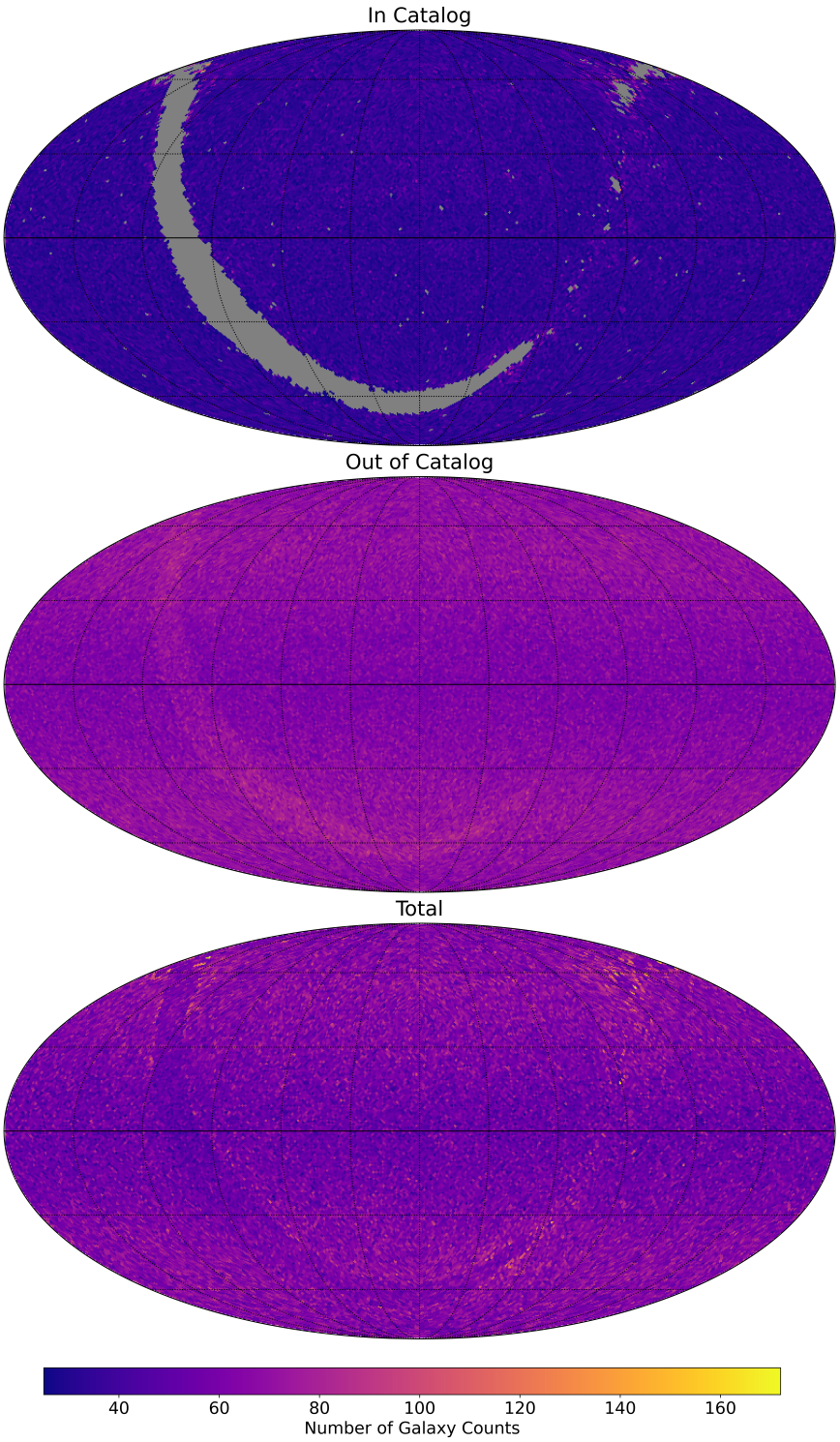

Figure 2.19: Mollweide projections showing the galaxy count distributions for three different categories for the synthetic GW events from Fig. 2.16, for an NSIDE = 64. The top panel displays the counts of galaxies within the catalogue (pixels removed for zero counts for better readability), the middle panel shows the counts of galaxies outside the catalogue, and the bottom panel presents the total counts combining both in and out-of-catalogue galaxies. The color bar beneath the projections indicates the number of galaxy counts, with the scale spanning from 40 to 160.

Table 2.2: Percentages of the data sets for in and out-of-catalogue for each run and detector setup

Run	Detectors	in-catalogue (%)	out-of-catalogue (%)
O1	H1, L1	42	58
O2	H1, L1	31	69
O2	H1, L1, V1	28	72
O3	H1, L1, V1	18	82
O3	H1, L1	19	81
O3	H1, V1	23	77
O3	L1, V1	23	77

In Fig. 2.19, the sky location bin counts of synthetic GW events are plotted for in-catalogue, out-of-catalogue, and total combined events. The top plot illustrates events originating within the GLADE+ galaxy catalogue, while the middle plot shows events sourced externally. Interestingly, the bottom plot, which combines in-catalogue and out-of-catalogue events, reveals an under density of counts along the Galactic plane of the Milky Way. This is unusual, as a uniform distribution was expected when combining both in-catalogue and out-of-catalogue events. This effect was observed in data sets generated for the O1 and O2 runs, where we saw a more pronounced concentration of catalogue-hosted GW events, showing an excess of events in the plane of the galactic band. The reasons for these under densities in the galactic plane are still unknown and may be related to differences in detection sensitivity, data integration methods, or the intrinsic distribution of the events. No clear explanation has been found for this effect, and future work will investigate this feature more thoroughly.

Overall, the data distribution for the O3a and O3b runs, with the H1, L1, and V1 detectors active, has been generated. We then proceeded to generate datasets reflecting each observing run and each detector setup. Each of these different datasets will be used to train a separate NF. To assess how much of the EM information from the catalogue each dataset will be using, we can look at the different contributions, in percentages, between in-catalogue and out-of-catalogue for each dataset.

The ratios of in-catalogue and out-of-catalogue events are shown in Tab.2.2, with the O1 HL dataset showing 42% in-catalogue, while the O3 HLV dataset shows an in-catalogue contribution of 18%. This is a good sanity check, as we expect that with better sensitivities, the number of detectable events would increase with luminosity distance, thus placing

more events in the out-of-catalogue category. This indicates that to introduce more EM information from galaxy catalogues, more in-depth catalogues are necessary, spanning deeper ranges of redshifts, and thus increasing the expected number of GW events sourced from the catalogue.

2.7.1 Batching the Data Generation with N_{select}

Despite major improvements in efficiency and speed, further enhancements in data generation are possible. Currently, generating 3 million samples takes approximately 21 hours. To further enhance efficiency and minimise the time required to generate synthetic GW events, while still covering the entire prior space defined in Table 2.1, we introduce a parameter N_{select}.

The parameter N_{select} is designed to increase the number of samples drawn from each distribution, thereby enhancing detection efficiency. If there are N different samples of H_0, we sample N_{select} times from the redshift and mass distributions, and other priors, for each H_0. This means each value of the Hubble constant is repeated N_{select} times. After each iteration of the data generation process, we detect N_{det} events, thereby decreasing the number of events we need to synthetically generate after each iteration by $N - N_{\text{det}}$. To further increase efficiency, we can increase N_{select} after each iteration as follows:

$$N_{\text{select}}(t = i+1) = N_{\text{select}}(t = 0) \times \left\lfloor \frac{N}{N - N_{\text{det}}(t = i)} \right\rfloor, \tag{2.65}$$

where $N_{\text{select}}(t = i+1)$ and $N_{\text{select}}(t = 0)$ are the values of the N_{select} parameter at the $i+1$ iteration and the zeroth iteration, respectively, N is the total number of events we wish to generate, and $N_{\text{det}}(t = i)$ is the cumulative number of events detected at iteration i. The floor function $\lfloor \cdot \rfloor$ ensures that the fraction gets rounded to the nearest integer, as $N_{\text{select}}(t = i+1)$ must always be an integer.

Starting at iteration $i = 0$, $N_{\text{det}}(t = 0) = 0$ since no events have been detected yet. For example, if we are interested in detecting 1,000 events, and at the nth iteration $N_{\text{det}}(t = n) = 900$, then the new updated $N_{\text{select}}(t = n) = 50$ for an initial setup of $N_{\text{select}}(t = 0) = 5$. This allows the data generation code to be more efficient when generating events from

priors that are detectable in parameter space regions where detectability is very unlikely, such as low values of H_0. While there might be more sophisticated methods to update the N_{select} parameter after each iteration, the method described in Eq. (2.65) efficiently updates N_{select}, minimising memory usage and processing time.

The speedup from using this process is about 3 times faster, allowing us to generate events from an O3 run with the HLV setup, generating 3 million samples in about 9 hours, when choosing a $N_{\text{select}}(t=0)=2$. Of course, we also ensure that out of the N_{select} GW samples we obtain for each H_0, we only retain one from each batch, even if there are more than one detected events for a single value of H_0.

2.8 Conclusion

In this chapter, we introduced `CosmoFlow`, a data generation tool enhanced with ML capabilities, designed to simulate GW events from an EM catalogue. The tool includes all relevant GW event parameters, as well as information from the GLADE+ catalogue.

We began by outlining the Bayesian framework within which `CosmoFlow` operates and the primary purpose of this analysis: to estimate the posterior distribution of cosmological and population parameters using GW event observations. With a clear understanding of the posterior distributions outlined in Eq. (2.12), our goal was to evaluate this equation by expediting the analysis using a *conditional* NF, conditional on the cosmological and population parameters. Before training NFs, we generated training data reflecting the probability distribution outlined in the numerator of Eq. (2.12), i.e., $p(\boldsymbol{\theta}|D,\Omega,I)$, or the probability of the GW parameters conditioned on the event being detected and the cosmological and population parameters. For the 1D case, the conditional cosmological parameters will only consider H_0.

We then outlined the process for generating these datasets in a flowchart (Fig. 2.3), allowing for EM data to be introduced from galaxy catalogues such as GLADE+ through the evaluation of the apparent magnitude of each synthetic host galaxy we generated from priors and comparing it with the *magnitude threshold* map of GLADE+. This map acts as a cutoff threshold for the actual EM catalogue, differentiating what would be considered in-catalogue and out-of-catalogue from synthetically generated GW events. An MLP was used to vectorize the SNR calculation of each simulated event, expediting the

data generation process by about 4 orders of magnitude, thus bypassing the bottleneck of evaluating the SNR for each event individually. The MLP was trained to evaluate the SNR multiplied by the luminosity distance for each detector for specific observing runs, using time-averaged PSDs for each run. We then generated 3 million data points for each detector setup and observing run, where each dataset will be used to train a NF, consistent with the different types of events observed and categorised in the GWTC catalogues [99]. The overall distributions presented in Fig. 2.16 are consistent with the expected distributions of the events, divided into in-catalogue and out-of-catalogue events. Each dataset showed different percentages of in and out, as tabulated in Tab. 2.2.

After the successful generation of a comprehensive dataset for each observation run and detector setup, the next step involves training the NF. This involves training an NF model for each distinct setup to encapsulate the probability distribution of $p(\boldsymbol{\theta}|H_0,\boldsymbol{D},\boldsymbol{I})$, where $\boldsymbol{\theta}$ are the GW parameters and $\boldsymbol{D}$ is the detectability status. It is essential to generate data that accurately reflects the specific conditions under which events are detected. This alignment between data generation and event conditions ensures the accuracy and reliability of our results. By properly accounting for the joint probability distribution under various scenarios, we can minimise potential biases in our analysis.

Chapter 3

CosmoFlow: Normalising Flow

In Chapter 2, we introduced `CosmoFlow`, which aims to estimate the posterior distribution over cosmological and population parameters using observed GW events. This Bayesian framework is detailed in Sec. 2.2, where we outlined the mathematical foundations, culminating in the posterior distribution equation for cosmological parameters conditioned on observed GW data, as follows:

$$p(\Omega \mid \mathbf{h}, \mathbf{D}, I) \propto p(\Omega \mid I) \prod_i \left\langle \frac{p(\theta_{i,\mathrm{GW}} \mid \Omega, D_i, I)}{p(D_i \mid \theta_{i,\mathrm{GW}}, I) p(\theta_{i,\mathrm{GW}} \mid \Omega_0, I)} \right\rangle_{\theta_{i,\mathrm{GW}} \sim p(\theta_{i,\mathrm{GW}} \mid h_i, I)} . \tag{3.1}$$

The goal of the `CosmoFlow` analysis is to estimate Eq. (3.1). To expedite this process, we employ an approach driven by machine learning, specifically using a conditional normalising flow (NF) to evaluate the numerator in Eq. (3.1), $p(\theta_{\mathrm{GW}} \mid \Omega, D, I)$.

3.1 Introduction

As described in Chapter 1, a NF is a type of probabilistic model that transforms a simple, tractable distribution (such as a Gaussian) into a more complex target distribution using a series of invertible and differentiable transformations, conditioned on some input data [69, 71]. This allows for flexible modeling of complex distributions while maintaining the ability to compute exact likelihoods and perform efficient sampling. Starting from a standard Gaussian distribution, we apply a series of transformations using trainable parameters ϕ to scale and shift the distribution, ultimately transforming it into the input data, generated in Chapter 2. This process is achieved by using the change-of-variable formula described in Eq. (1.58), which allows the Jacobian of the transformation to be

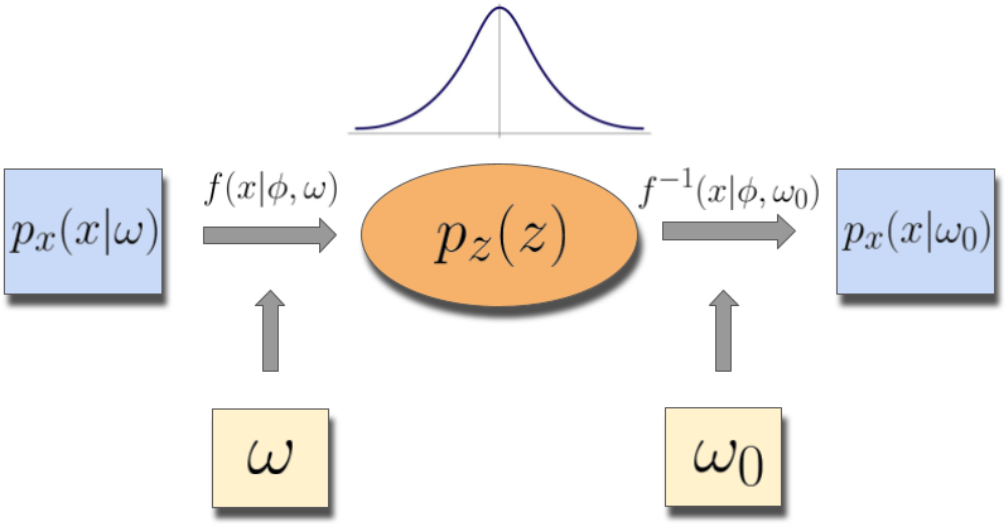

Figure 3.1: This diagram illustrates the transformation of the posterior distribution $p_x(x \mid \boldsymbol{\omega})$ using a latent variable z and a transformation function f parameterized by $\boldsymbol{\phi}$. The posterior distribution $p_x(x \mid \boldsymbol{\omega})$ is given on the left, and the function $f(x \mid \boldsymbol{\omega}, \boldsymbol{\phi})$ is applied to map the distribution $p(x \mid \boldsymbol{\omega})$ to a latent probability space represented by $p_z(z)$, shown in the center (Gaussian). The inverse function $f^{-1}(x \mid \boldsymbol{\omega}_0, \boldsymbol{\phi})$ maps the latent variable z back to the posterior distribution $p_x(x \mid \boldsymbol{\omega}_0)$ for a new set of conditional statements $\boldsymbol{\omega}_0$. The transformed posterior distribution $p_x(x \mid \boldsymbol{\omega}_0)$ is depicted on the right.

modeled with these trainable parameters. We define the change-of-variable equation as

$$\log(p_{\mathbf{x}}(\mathbf{x} \mid \boldsymbol{\omega})) = \log(p_{\mathbf{z}}(f^{-1}(\mathbf{x} \mid \boldsymbol{\phi}, \boldsymbol{\omega}))) + \log\left(\det\left|\frac{df^{-1}(\mathbf{x} \mid \boldsymbol{\phi}, \boldsymbol{\omega})}{d\mathbf{x}}\right|\right), \tag{3.2}$$

where $\mathbf{x}$ is the input data, which in the `CosmoFlow` analysis consists of 14 input parameters corresponding to the GW parameters, described in Tab. 2.1. The phase parameter is excluded, as we are interested in parameters affected by selection effects, and since phase does not influence the SNR of an event, it is excluded. The conditional parameters are the Hubble constant values and the detectability status, hence $\boldsymbol{\omega} \equiv [H_0, D]$. During data generation, we ensured that the synthetic GW events are detectable above a specific SNR threshold; this makes the detectability status, D, a placeholder indicating that the events the flow is training on are all detectable, but it has no practical effect on the conditional statement.

Figure 3.1 illustrates the process of evaluating the left-hand side of Eq. (3.2) for specific conditional input values. Here, the input data (sampled from p_x) is input into the flow, with conditional statement $\boldsymbol{\omega}$, sampled from a specific prior, used together to optimise the parameter $\boldsymbol{\phi}$ of the function f, which transforms p_x into the latent space p_z. After

training the NF, the parameters ϕ are optimised while keeping the transformation function invertible and tractable, allowing for evaluation of f^{-1}. With the optimised parameters ϕ, we can now evaluate the desired probability distribution $p_x(x \mid \omega_0)$ for any specific conditional input parameters, ω_0.

In this chapter, we will explore how the conditional NF is trained with the GW input parameters, with conditional inputs as H_0. We will quantify the learning process by evaluating the loss function of the training routine, then we will perform performance tests on the NF, which include the resampling test, the Probability-Probability test, and the final bias test. We will examine why these tests are essential for assessing the trained NF's capacity to evaluate the correct distribution.

3.2 Training the Normalising Flow

The training of the normalising flow (NF) is facilitated by the Python package `glasflow` [135], which acts as a wrapper for the well-established `nflows` [136] library, which specializes in NFs. With `glasflow`, users have the ability to easily initialise flow models of different types, including the RealNVP (Non Volume Preserving) flow model and the spline-based flow known as CouplingNSF (Neural Spline Flow). For clarity and conciseness, we will focus on results from the flow model using the O3 data set with the HLV (Hanford-Livingston-Virgo) detector setup; the flows trained for the other data sets areshown in the Appendix A.

In Table 3.1, we present the hyper-parameters employed for training the NF, as:

- The **Batch Size** specifies the subset of the dataset used at each iteration to estimate the loss value as defined in Eq. (1.63). A suitable batch size ensures efficient training and stable gradient estimates, which are important for the model convergence.
- The **Number of Epochs** indicates the total number of complete passes through the entire training dataset.
- The **Shuffle Option** determines if the data is randomized before each epoch. Shuffling helps in preventing the model from learning the order of the data, which can improve generalization.
- The **Activation Function** is a non-linear function applied to the output of each layer to introduce non-linearity and allow the network to learn complex patterns.

- The **Dropout Probability** indicates the proportion of network connections (weights) omitted during training for regularization. This technique helps to minimise overfitting by preventing the network from becoming too reliant on any particular set of connections.
- The **Learning Rate** sets the magnitude of updates to the weights and biases (size of the step) during training. A well-chosen learning rate ensures that the model converges efficiently and avoids oscillations or divergence.
- The **Learning Rate Scheduler** is a mechanism that adjusts the learning rate over time. Gradually decreasing the learning rate allows for finer adjustments to the weights during later stages of training, improving the accuracy of loss value estimates and aiding in generalization.
- The **Split Parameter** determines how much of the data is used for training and how much is used for validation. This helps in monitoring the model's performance on unseen data, checking if the model is generalizing well over the data or not.
- The **Optimiser** is the algorithm used for updating the network's weights. Common optimisers include SGD [137] and Adam [63]. The choice of optimiser affects the speed and stability of the training process.
- The **Linear Transform** is applied before each coupling transform for data manipulation.
- The **Neurons, Layers, and Blocks per Transform** define the architecture's depth and complexity. N_{neurons} refers to the number of neurons per layer, N_{layer} indicates the number of layers, and N_{blocks} specifies the number of residual blocks per transform.
- The **Input Dimensions** specify the size of the target data, which directly affects the model's architecture and the complexity it needs to handle.
- The **Conditional Input Dimensions** specify the size of the conditional dataset.
- The **Flow Type** determines the NF method used, which in `glasflow` could either be `RealNVP` or `CouplingNSF`.
- The **XYZ Parameter** controls spatial coordinate scaling, with True for Cartesian coordinates and False for Polar coordinates. This parameter transforms the input data, specifically the right ascension, declination, and luminosity distance, i.e., $\alpha, \delta, D_{\text{L}}$, into x, y, z spatial Cartesian coordinates if enabled.
- The **Scaler** defines the input data scaling method, which is important for normalising the data to improve model performance and training stability.
- The **Mask** is used in constructing the flow, determining which parts of the data are transformed [136]. This helps in focusing the model's learning on relevant features.
- The **Number of Bins** determines the spline interpolation granularity when using the flow type `CouplingNSF`. A higher number of bins allows for more detailed transformations but may increase computational complexity.

Hyperparameter	**Value**
Batch Size	50,000
Number of Epochs	1,000
Shuffle Option	True
Activation Function	ReLU
Dropout Probability	0.0
Learning Rate	0.0005
Learning Rate Scheduler	CosineAnnealingLR [138]
Data Split Ratio	80% (Training)
Optimizer	Adam [63]
Linear Transform	LU Decomposition
Neurons	128
Layers	5
Blocks per Transform	3
Input Dimensions	14
Conditional Inputs	1
Flow Type	CouplingNSF [77]
XYZ Parameter	False
Scaler	Standard
Mask	None
Number of Bins	4

Table 3.1: Hyperparameters for training the normalizing flow.

For additional details on the implementation and other initial setups for the flow using `glasflow`, including configuration options, modular design, and usage guidelines, please refer to Ref. [135]; this provides comprehensive documentation on the functionality of `glasflow`, including its compatibility with PyTorch, the implementation of NFs, and specific details on the `CouplingNSF` architecture

From here we initiated a CouplingNSF model with the configuration parameters shown in Tab. 3.1. Once configured, the model can be moved to a computational device, such as a Compute Unified Device Architecture (CUDA)-enabled GPUs, to leverage accelerated computing resources, thereby enhancing performance for model training and inference. This step allows handling larger datasets or for achieving faster computation times in practical applications [139].

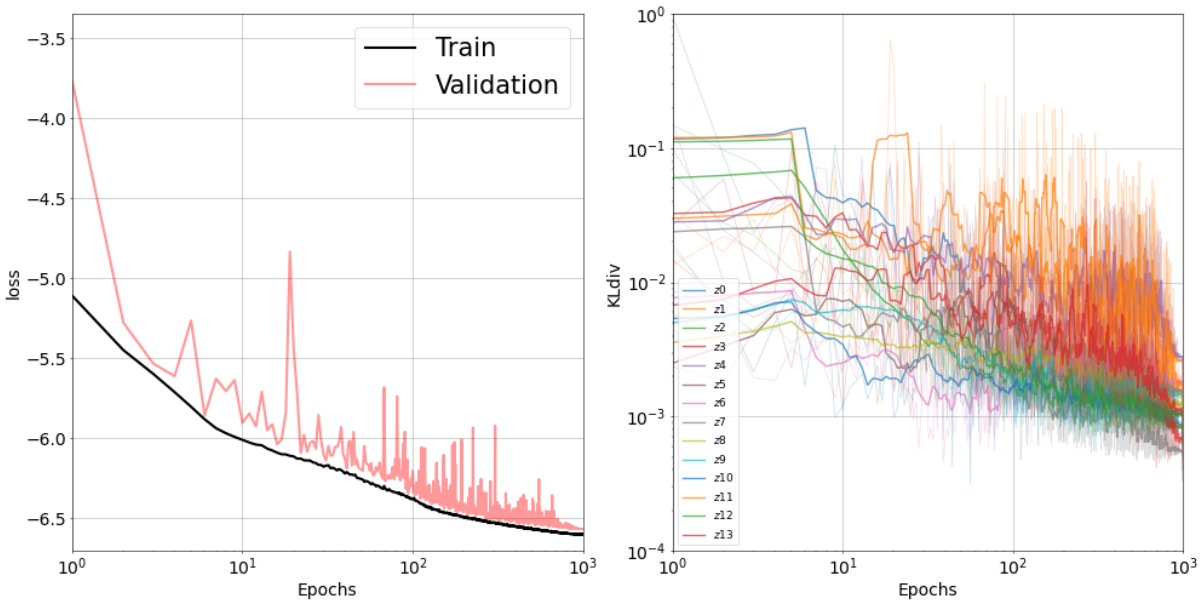

Figure 3.2: (Left): Loss versus epochs for both the training and validation data. (Right): The Kullback-Leibler (KL) divergence between a 1D Gaussian distribution and each dimension of the latent space is plotted against epochs. The curves have been smoothed using a running mean filter for improved visualisation, with the original curves shown in the background in a transparent style. The KL divergence values are measured in nats.

After initialising the parameters, the subsequent phase involves training the NF. In ML, a widely recognized method to show how a model performs during training is to observe the values of the loss function, expressed in Eq. (2.11), at each epoch. Ideally, one aims for a general trend of decreasing values as the model progresses through the predetermined number of iterations, although fluctuations and occasional increases are common, as evidenced by the loss plot in Fig. 2.11. The flow model was trained for 1000 epochs using the `glasflow` package. The training process, accelerated by an NVIDIA RTX G-Force 2080 GPU, took approximately four hours. Given the specific nature of the NF model, which aims to transform a simple latent distribution into a more complex target distribution, an additional metric can be employed to assess the training performance. The primary objective of NFs is to scale and translate the input dataset to approximate a normal distribution. At various stages of the training process, we can draw samples from the latent space to evaluate the model's performance. By performing a kernel density estiamtion (KDE) on these samples and expecting them to approximate a normal distribution, we can assess the closeness of our sampled distribution to an ideal normal distribution. This comparison can be quantitatively measured using the KL divergence, as defined by the equation

$$D_{\mathrm{KL}}(P||Q) = \sum_i P(x_i) \log \frac{P(x_i)}{Q(x_i)}, \tag{3.3}$$

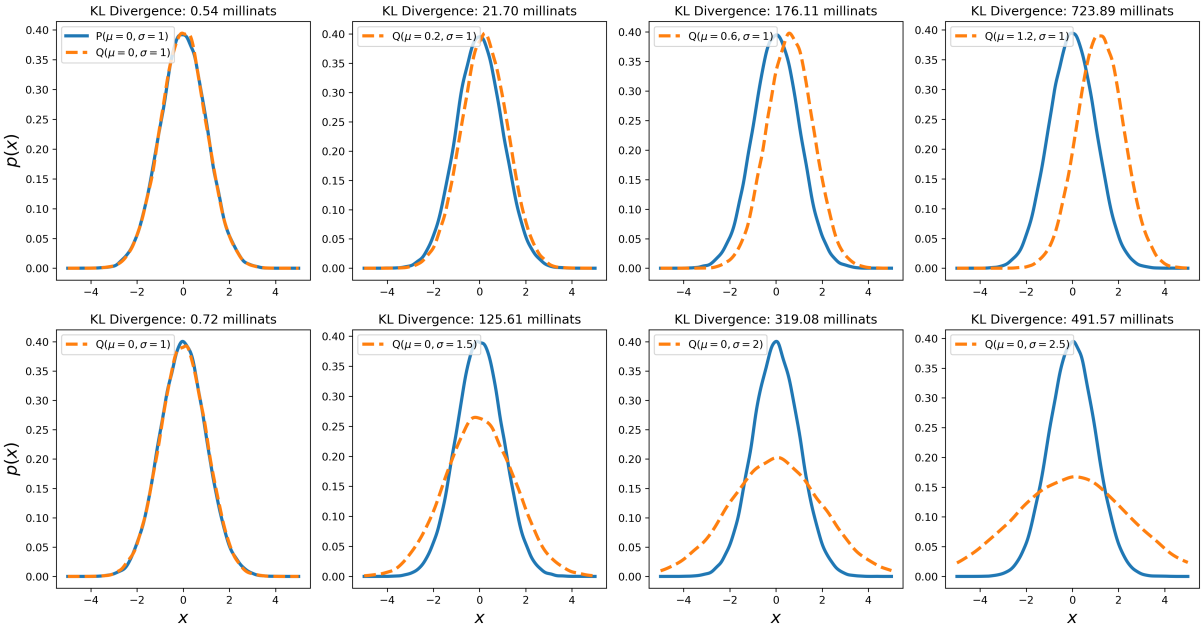

Figure 3.3: The Kullback-Leibler (KL) divergence between two Gaussian distributions: as the mean of the orange dashed distribution (Q) diverges from the solid blue distribution (P), the relative separation increases. The KL divergence, expressed in millinats, quantifies the growing difference between the two distributions. The second row represents the same idea, but with the mean value fixed and varying the standard deviation of the Gaussian for Q. Each subplot shows the PDFs of the two distributions at varying levels of separation.

where P is the true distribution and Q is the approximated distribution modeled by the flow and x_i is the ith sample from the P distribution. In Fig. 3.2, the plot on the right shows the evaluated KL divergence at each epoch of training for each dimension of the latent space, z_0 up to z_{13} (corresponding to 14 dimensions of the GW parameter space). While the latent space has the same dimensions as the input data (14 GW parameters), these dimensions are not strictly related one-to-one due to correlations in the data. The monotonic decrease in the KL divergence for each dimension suggests that the latent space distributions are progressively approaching closer and closer to a Gaussian distribution as the NF training proceeds.

To provide a reference for the magnitude of KL divergence values, a simple illustrative example is presented in Fig. 3.3. Two Gaussian distributions are compared across four different stages. The first distribution, P, remains fixed, centered at 0 with a standard deviation of 1. In the first row, the second distribution, Q, shifts its mean across four instances with values $[0.0, 0.2, 0.6, 1.2]$ while maintaining a standard deviation of 1. In the second row, Q varies its standard deviation with values $[1, 1.5, 2, 2.5]$, keeping the mean fixed at 0. For each distribution, 50,000 samples were generated, and a Gaussian KDE was used to estimate the PDFs. It is important to note that the use of KDE introduces a slight amount of noise into the estimated distributions, adding an additional layer of variability

to the setup. This noise simulates the types of distributions that might be encountered in practical scenarios discussed later in this thesis. By examining the evolution of the Q distribution, we observe how the KL divergence values increase with the separation between the distributions. The resulting KL divergence values are $[0.54, 21.7, 176.11, 723.89]$ millinats for the changing mean example, and $[0.72, 125.61, 319.08, 491.57]$ millinats for the varying standard deviation case. The KL values were computed using Eq. (3.3).

Then, at each stage of training, the KL divergence between a normal Gaussian distribution (centered at zero with unit standard deviation) and the latent samples of the n-th dimension is compared by sampling 10,000 samples from both P and Q and evaluating Eq. (3.3). A KL divergence value of zero indicates that the sampled distribution exactly matches the standard normal distribution, while larger values indicate greater differences. However, when estimating it from samples through a KDE, there will be noise from sampling, making it impossible to achieve a perfect zero. The extent to which the KL divergence can approach zero depends on the number of samples used. Thus, a lower KL divergence value indicates a closer match to the standard normal distribution, highlighting the effectiveness of the training process. From Fig. 3.2, the KL values across all dimensions show a steady and noisy descent towards low values, achieving final KL values ranging from 4 down to 0.8 millinats. This approach validates the model's capability to accurately transform and fit the data to a normal distribution.

A comparison between the latent space and a normal multivariate distribution is illustrated in Fig. 3.4. In this figure, the black contours represent the distribution of samples in the latent space after training the NF, while the red contours depict samples from a multivariate normal distribution. Each plot uses 10,000 samples. The KL divergences for the 1D samples between the latent space and the normal distributions are also displayed, with values presented in units of millinats (where 1 millinat $= 10^{-3}$ nats, a measure of information divergence). A KL divergence value below 4 millinats indicates strong agreement between the latent and target distributions. This demonstrates that the training process was effective in mapping the target data into the normal distribution. This plot shows the latent space p_z at the final stage of training, indicating consistency with a multivariate normal distribution and confirming the success of the flow model's training. To simplify the evaluation, the KL divergence was computed on the individual one-dimensional data sets instead of the full 13 dimensions because evaluating the KL divergence with that many dimensions was challenging. Therefore, we resorted to this practical and straightforward approach.

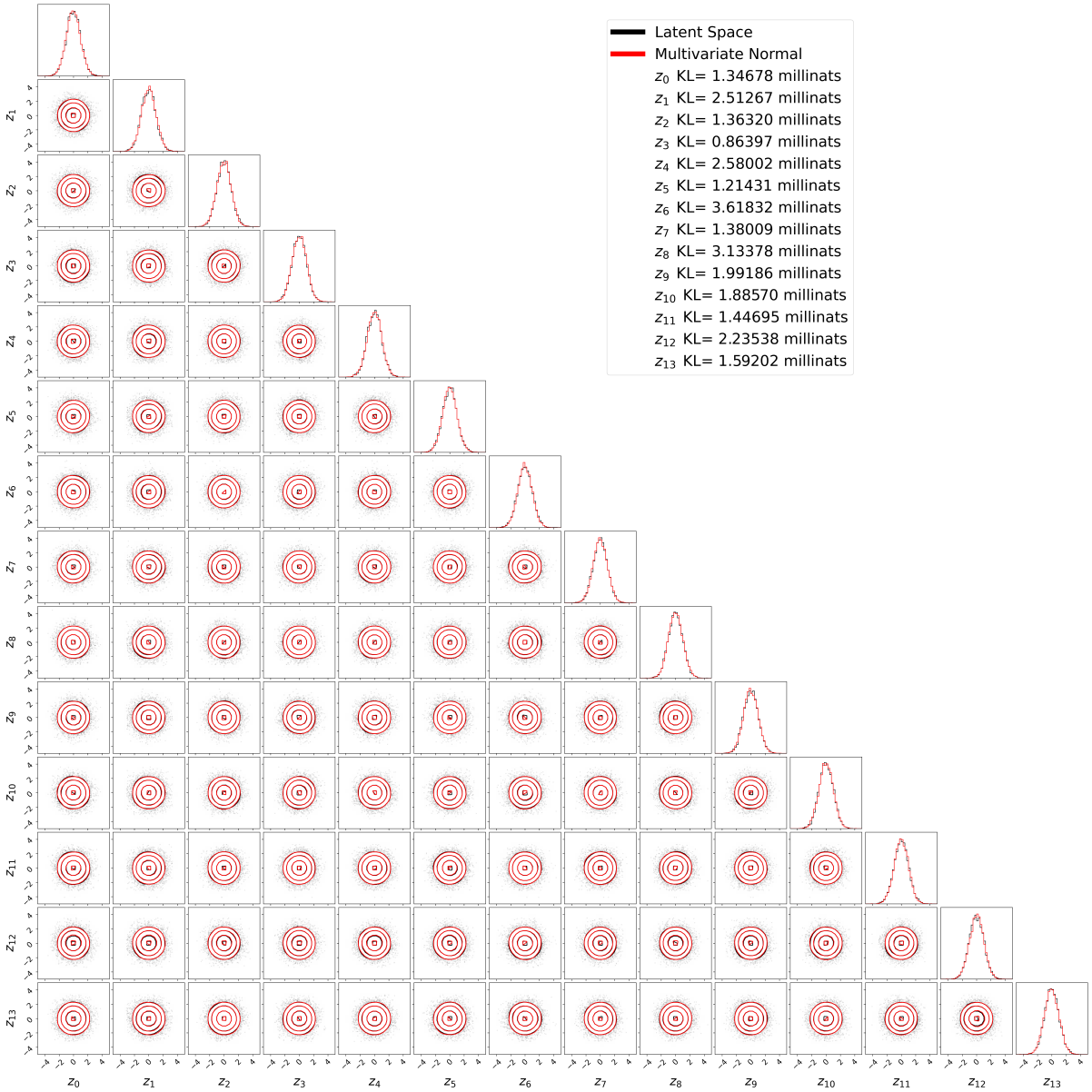

Figure 3.4: Corner plot with the latent samples distribution and overlaid the samples generated from the multivariate normal distribution. The KL divergences of each 1D distribution are shown in the legend. The contour levels are 68%, 95% and 99% of the total data enclosed

To better understand the information enclosed in Fig. 3.4, we present the covariance matrix of the latent-space samples in Fig. 3.5. This matrix helps us understand the correlation between each dimension. The covariance between two dimensions X_i and X_j, for n samples, is given by

$$\text{cov}(X_i, X_j) = \frac{1}{n-1} \sum_{k=1}^{n} (X_{ki} - \bar{X}_i)(X_{kj} - \bar{X}_j), \tag{3.4}$$

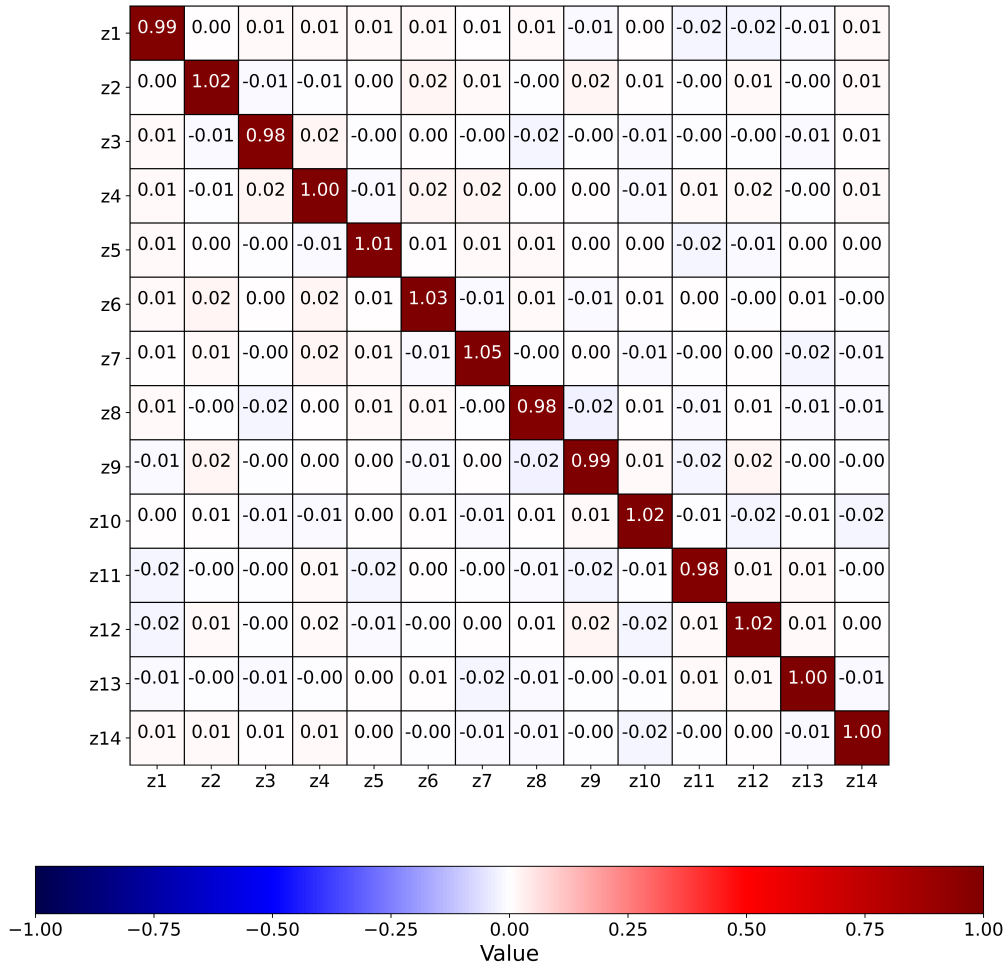

Figure 3.5: Covariance matrix for the latent space shown in Fig. 3.4. The color scale indicates the magnitude of the covariance, with values ranging from -1 to the maximum covariance. The diagonal elements represent the variance of each latent dimension.

where X_{ki} is the kth sample of the ith dimension, X_{kj} is the kth sample of the jth dimension, and $\bar{X}_i$ and $\bar{X}_j$ are the means of the ith and jth dimensions. Computing the covariance matrix between latent space dimensions shows very little covariance, as indicated by the fact that the diagonal entries are all one or close to one. This demonstrates that the latent distribution is both Gaussian (low KL) and uncorrelated (low covariance).

After training the NF to the point where the loss no longer decreases, we select and save the best NF state, defined by the lowest validation loss, for future use. Before using the NF for the intended calculations, we conduct preliminary tests to ensure it has been properly trained. These tests include the resampling test, Probability-Probability (PP) test, and bias test.

3.2.1 Resampling test

The resampling test is designed to evaluate the effectiveness of the mapping functions employed by the flow. In Fig. 3.4, the latent space showed high Gaussianity, indicating that the mapping function $f(x|\phi,\omega)$ has been correctly tuned. However, this does not provide a good indication of whether the inverse process, using f^{-1} as depicted in Eq. (3.2), is sufficiently accurate to map back to the initial distribution, p_x. To address this, we perform the resampling test. This involves generating samples from the latent space and then transforming these back into the original target data space to verify how closely the final distribution aligns with the training data. This process assesses the accuracy and reversibility of the flow's mapping functions. Significant discrepancies between the original and resampled distributions indicate that the inverse functions were not optimally calibrated or sufficiently trained.

To conduct this test, we uniformly sample from the conditional space, $\omega \equiv [H_0, D]$. As mentioned previously, the detectability status is just a placeholder in the conditional statement, indicating that all the GW parameters we sample from the flow should be detectable events. We then sample H_0 from the prior distribution used in the data generation ($U \sim [20,140]\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$) and input these into the conditional statement of the trained NF. Having satisfied the conditional part of the NF, we can then directly sample from the flow.

The sampling is performed straightforwardly by first sampling from a standard normal distribution. Each sample is then mapped from the latent space, p_z (which we have assumed to be perfectly Gaussian, even though it slightly deviates from it), to the target data space, p_x. The new samples in the target data space are now the resampled GW parameters from the flow. The results of the resampling test are shown in the corner plot in Fig. 3.6, where the target data (in black), the actual data used to train the flow, and the samples from the flow (in red) are overlaid. Qualitatively, the distributions match very well, except for the marginalized uniform distributions, specifically in a_1, ϕ_{JL}, ϕ_{12}, ψ, and t_{geo}. This discrepancy is likely due to the fact that these are uniform hard-bounded distributions.

The NF models usually struggle with uniform distributions, especially those with hard boundaries, because these distributions lack smoothness and have sharp edges, making it challenging for the smooth, invertible transformations in NF to capture them accurately. Additionally, the sharp density changes at the boundaries can cause instability during

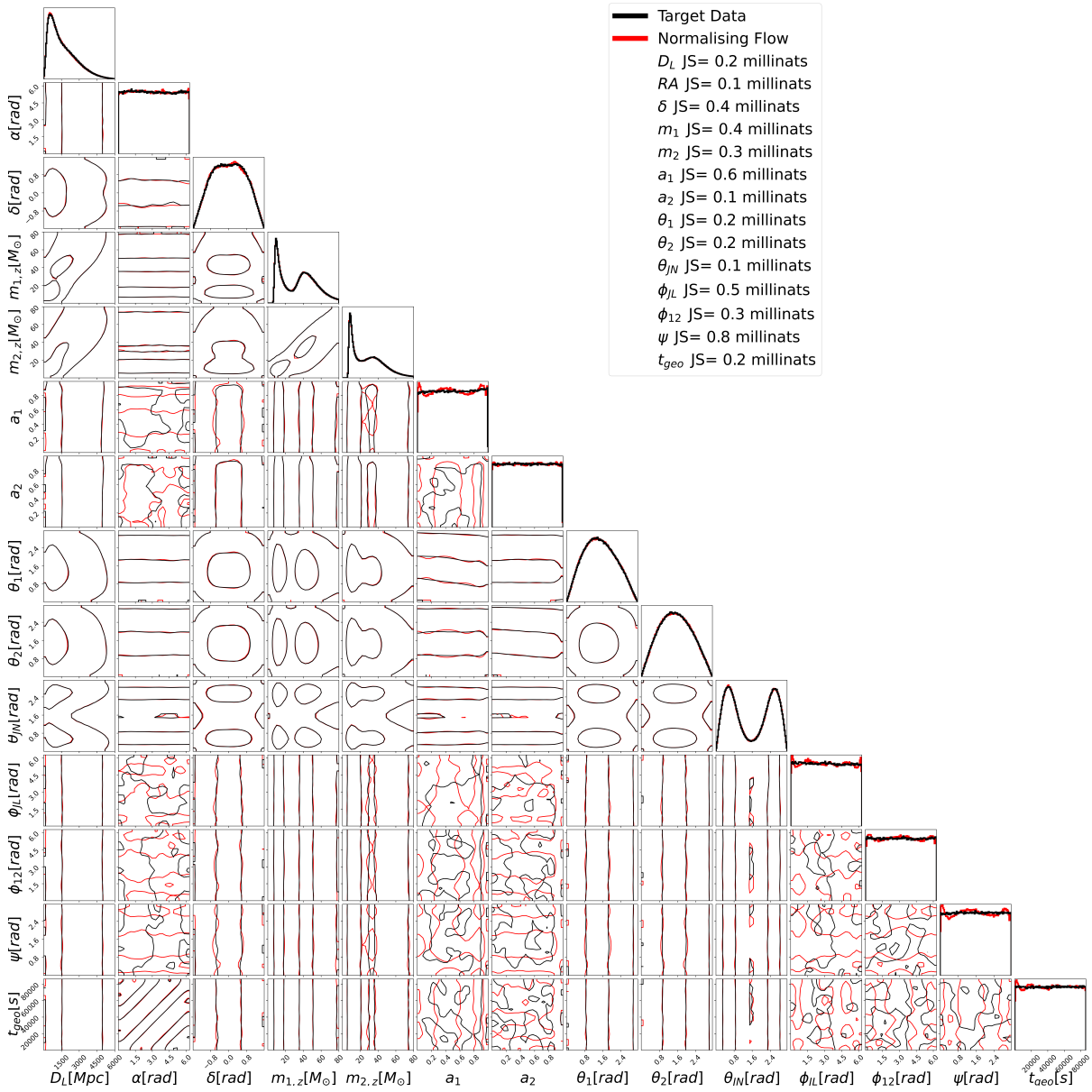

Figure 3.6: Corner plot illustrating the target data distribution, consisting of 3 million data points, overlaid with samples generated from the trained flow model, which includes 1 million data points. The contours represent the 50% and 99% confidence levels of the total data. The black lines denote the target data distribution, while the red lines indicate the distributions generated by the NF. The Jensen–Shannon (JS) divergences for each one-dimensional marginal distribution are also displayed, quantitatively indicating the similarity between the target data and the generated samples. The JS divergence values for each parameter are presented in the legend, showing variations across different parameters.

training and poor gradient estimates [140]. The initial reason for using neural spline flows was to mitigate issues with hard boundaries in distributions by increasing the number of spline bins, which helps to handle these boundaries more effectively [77]. However, it appears that this approach was not sufficient to fully address the problem.

One way to mitigate this effect would be to apply a transformation to the uniformly distributed GW parameters in the $p(\theta|H_0,D)$ distribution. This can be achieved using a logistic function, given by

$$\text{logistic}(x) = \frac{1}{1+e^{-x}}. \tag{3.5}$$

By passing the uniformly distributed samples through Eq. (3.5), the boundaries extend from finite to $[-\infty,\infty]$. This essentially transforms a uniform flat distribution into a Gaussian, allowing the NF model to better train and learn these parameter dimensions due to the new smooth parameter distributions. For the resampling test, one must remember to convert the transformed distribution back to its initial shape by applying the inverse of the sigmoid function, the *logit* function.

The JS divergences between the target data and the distributions given by the NF are also shown for each marginalized distribution in Fig. 3.6. In this case, the JS divergence was chosen for its symmetry, which is crucial for comparing two distributions without favoring one over the other. When comparing two distributions, if one is the target distribution, the KL divergence is often sufficient to measure how far off the approximating distribution is from the target. However, for the resampling test, we need to quantitatively determine the offset between the real data and the resampled data from the NF without ambiguity or directional bias. The JS divergence, being symmetric, ensures that the comparison treats both distributions equally, making it more suitable for this purpose. It is defined as

$$\text{JS}(P||Q) = \frac{1}{2}D_{\text{KL}}(P||M) + \frac{1}{2}D_{\text{KL}}(Q||M), \tag{3.6}$$

where $M = \frac{1}{2}(P+Q)$ is the average of the two distributions, P is the target distribution, and Q is the distribution given by the NF. This symmetry property ensures $\text{JS}(P||Q) = \text{JS}(Q||P)$, unlike the KL divergence, which is asymmetric and measures the difference in a specific direction.

3.2.2 Probability-Probability test

The Probability-Probability (PP) test uses cumulative distribution plots to identify biases in the resampling of data spaces, providing a measure of the normalizing flow model's consistency and reliability in reproducing parameter distributions.

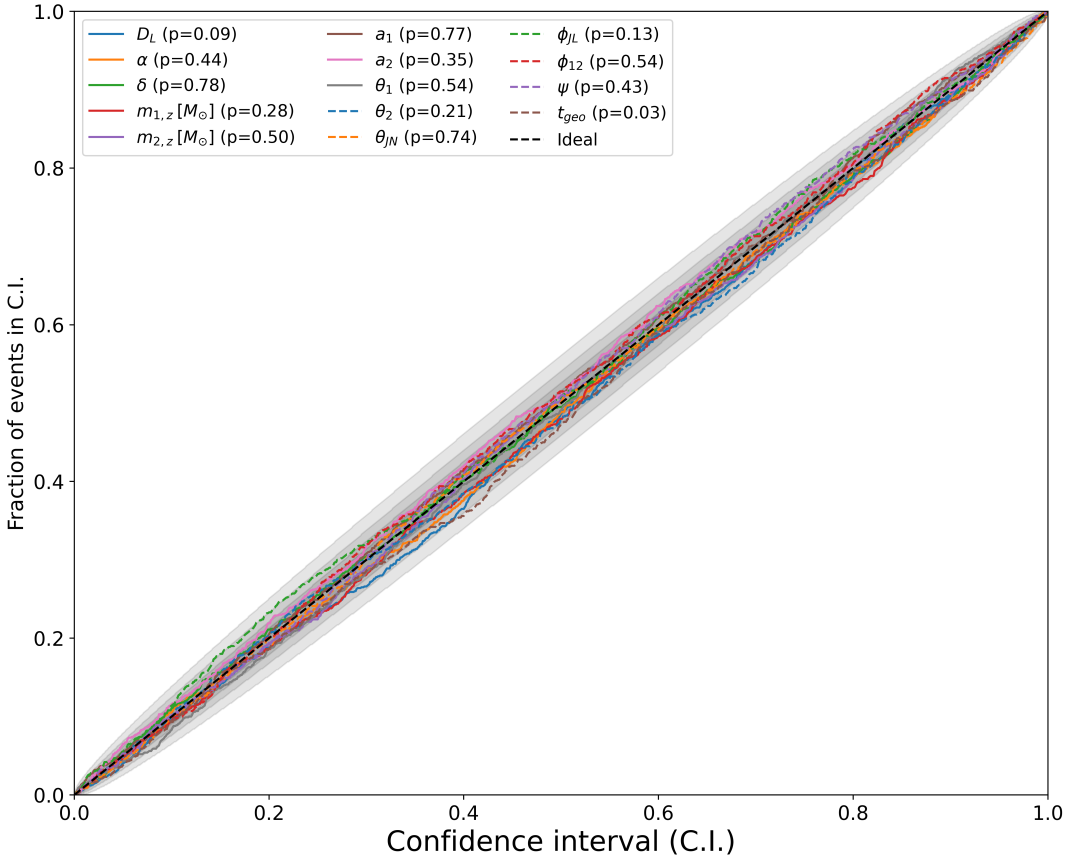

Figure 3.7: PP plot for various parameters of the latent space, comparing the fraction of events within a given confidence interval (C.I.) to the ideal 1:1 line. The parameters include distance D_{L}, angles α, δ, and θ, detector frame masses $m_{1,z}$ and $m_{2,z}$, spins a_1 and a_2, and additional parameters such as ϕ, ψ, and t_{geo}. The values in parentheses indicate the respective p-values. The y-axis represents the fraction of events within the specified confidence interval, while the x-axis represents the confidence interval. The shaded region represents the confidence intervals of 68%, 90%, 99% and 99.99%.

The process starts with uniformly sampling the conditional space (as the flow was trained) and then sampling the parameter space. Specifically, given data comprising 14 GW-event parameters and a cosmological conditional value, we sample the latent space while keeping the conditional values consistent with those used in training. This generates distributions for the parameters. For each sample in the training dataset, the true parameter values are known. By comparing the distribution generated by the flow model to these true values, PP plots are constructed.

In a PP plot, the x-axis represents the cumulative probability or confidence interval (C.I.), ranging from 0 to 1. The y-axis shows the fraction of events (or data points) for which the true parameter value falls within the corresponding confidence interval on the x-axis, also ranging from 0 to 1. The ideal scenario is represented by a 45-degree line ($y = x$) on the plot. This line indicates perfect calibration, meaning that the fraction of events within a given confidence interval matches the confidence interval itself. For instance, if the model is perfectly calibrated, 68% of the true parameter values should fall within the 68% confidence interval of their posterior distributions. If the plot lies above the ideal line, it indicates the model is overconfident, as the actual fraction of events falling within the given confidence intervals is higher than expected, suggesting the intervals are too narrow. Conversely, if the plot lies below the ideal line, it means the model is under confident, as the actual fraction of events falling within the given confidence intervals is lower than expected, indicating the intervals are too wide. We can also compute the p-value for each parameter, which is a measure that helps determine whether the results observed in data are real or occurred by chance, with a value ranging from 0 to 1. In the context of PP plots, a high p-value indicates that the model's predictions match the actual data well, while a low p-value suggests potential biases in the model, specifically for values below 0.05 [141]. For further understanding of PP plots, a recommended resource is available in [142].

In Fig. 3.7, the PP test outcomes for the training data parameters are displayed, indicating no inherent biases in data resampling through the flow. The PP plot was generated using the `bilby` Python package [100], with 1000 iterations and 50,000 samples per iteration. This extensive sampling ensures a thorough examination of the distribution for each parameter. The PP plot shown in Fig. 3.7 shows a maximum p-value of 0.78 for δ and a minimum p-value of 0.03 (just below the 0.05 statistical threshold) for the $t_{\rm geo}$ parameter. The combined p-value is 0.2516.

This test can also be performed over the conditional data space. This is done by evaluating the NF for an array of values in the conditional space, ω_0, for each individual synthetic GW event in the training dataset. Each GW synthetic event in the training dataset is assumed to be perfectly constrained (no uncertainty on the GW parameters), allowing us to use the posterior equation $p(H_0|\theta, D) \propto p(\theta|H_0, D)$. Since each event also has a true value of H_0, we can compare the number of events found within a confidence interval of the distribution evaluated, which in this case would be $p(H_0|\theta, D)$. An example of this process is shown in Fig. 3.8, where examples of $p(H_0|\theta, D)$ curves are shown for six synthetic GW events from the training data. This is achieved by computing the NF over fixed values of the GW parameters and varying H_0 in steps from $[20, 140]\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$.

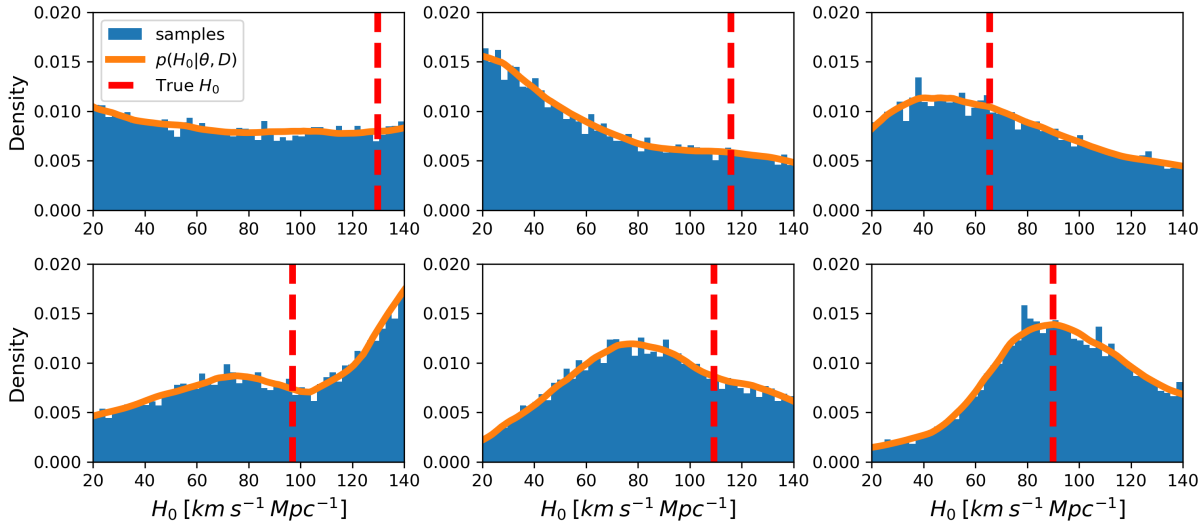

Figure 3.8: Subplot of different $p(H_0|\boldsymbol{\theta}, D)$ distributions in orange, with the appropriate samples from it, and the true value shown (red dashed line).

Now that we evaluate single event posteriors on H_0, we can compare the posterior distribution samples with the true values for each event and compute a PP plot, as shown in Fig. 3.9. Performing the same test with 1000 iterations with 50,000 samples each distribution, it is clear that there is no underlying bias, with a p-value of 0.736.

Although these tests are strong, showcasing the performance and the sampling power of the trained NF, a better approach would be to generate 1000 synthetic GW events from the data generation process with a fixed value of H_0 and then combine all the events together hierarchically. In this context, combining hierarchically refers to integrating the information from all events into a unified analysis using a statistical or probabilistic framework. This involves calculating a joint posterior distribution for H_0 by aggregating the likelihoods or posteriors from individual events, allowing the cumulative evidence from the entire dataset to constrain H_0 more effectively. The combined distribution can then be used to evaluate the number of samples within each confidence interval. We would perform this analysis over many values of H_0, each time generating 1000 events and combining them. The construction of the PP plot would lead to a more powerful result showcasing the unbiased nature of the trained NF.

We can also perform this test over one iteration, meaning fixing the value of H_0 and combining the event posteriors together to obtain an overall combined distribution and check if the true value lies within the distribution. This is the bias test mentioned in Section 3.2, which we will now discuss.

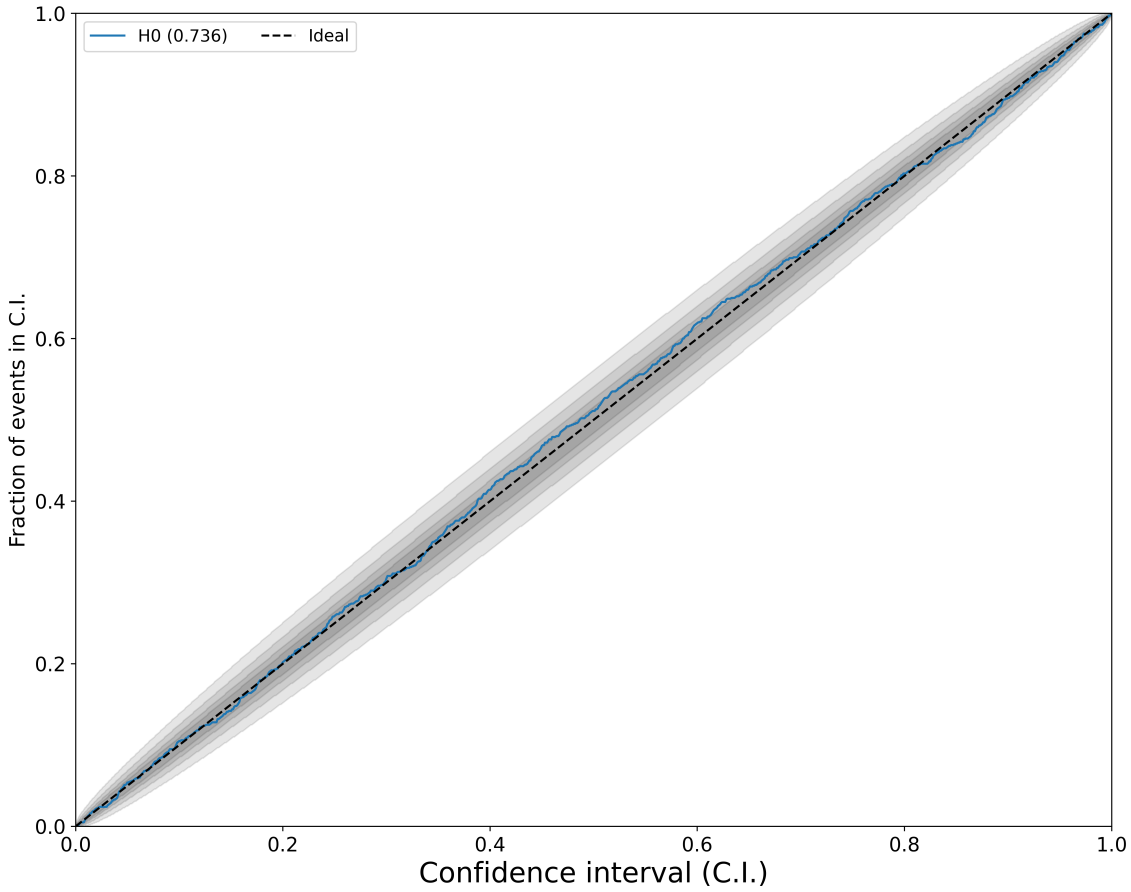

Figure 3.9: PP plot for H_0 sampled from $p(H_0|\boldsymbol{\theta},D)$ of the target space, comparing the fraction of events within a given confidence interval (C.I.) to the ideal 1:1 line. The value in parentheses indicates the respective p-value. The y-axis represents the fraction of events within the specified confidence interval, while the x-axis represents the confidence interval. The shaded region represents the confidence intervals of 68%, 90%, 99%, and 99.99%.

3.2.3 Bias test

In the bias test, we examine the inherent bias of the trained flow model. Using our data generation algorithm, we produce 1000 events with $H_0 = 70\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$. These events are treated as real with precisely constrained parameters, except for Gaussian noise in the SNR. We then use Eq. (3.1) to combine multiple events, omitting the denominator since it is constant across all events. This test can be performed for any value in the cosmological space, not just $H_0 = 70\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$.

By assuming there is no noise in the estimation of the synthetic GW parameters (other than the SNR), we simplify the probability term $p(\theta|h)$ to a delta function. This represents the best-case scenario for PE, thereby making the bias test more stringent. However, because the observed SNR is sampled from a non-central χ^2 distribution (with the network matched-filter SNR serving as the non-central parameter), there is still some uncertainty in detectability due to this sampling process. This introduces noise into the observed SNR, affecting detectability slightly. Therefore, the expectation value is calculated over the delta function, which allows us to ignore the denominator in Eq. (3.1), as it does not depend on the cosmological parameters being tested.

These events are passed to the flow model for probability evaluations. The results, shown in Fig. 3.10, present combined posterior analyses for the Hubble constant, H_0, across different totals of events: 10, 50, 100, 250, 350, 500, 750, and 1000. Each subsequent combined posterior includes all previous events, meaning that the combined posterior for 50 events also contains the 10 events from the previous plot, and so on. In the figure, the red curves represent the individual posteriors from each event, while the blue curve represents the combined hierarchical inference of H_0 by combining the single posterior distributions together. The true value of H_0 falls within the 2σ confidence interval of each combined posterior, except 1000 events combined. This might indicate a systematic bias of the NF at this level of precision.

We also present the mean H_0 value for each distribution, with error bars indicating the 95% highest density interval (HDI), as shown in the subplot below in Fig. 3.10. Additionally, we plot the theoretical curve showing how the standard deviation decreases by $\sqrt{N_{\text{events}}}$, in red. To plot these curves, we used the 95% HDI result of the first 10 combined events, took a mean of the uncertainties ranging above and below the mean value, and used that as the mean uncertainty starting at 10 combined events. We then divided the result by $\sqrt{N_{\text{events}}}$. From the plot, we see that the error bars are within the theoretical standard deviation, but there seems to be some level of systematic bias in the NF. To improve these results, increasing the amount of training data and dedicating additional time to training the model, potentially with higher complexity networks, may enhance the inference power of the NF, reducing bias when combining multiple events.

It's important to emphasize that this analysis assumes an ideal scenario where each GW event is perfectly measured, with no uncertainty, meaning they are perfectly localised in the GW parameter space. Under these conditions, combining 1000 events gives us an error of about 5 units for H_0, resulting in an accuracy of $\frac{5}{70} \approx 0.071$ (or approximately 7.1%). This means that even with 1000 perfect events, the best accuracy we could achieve for H_0

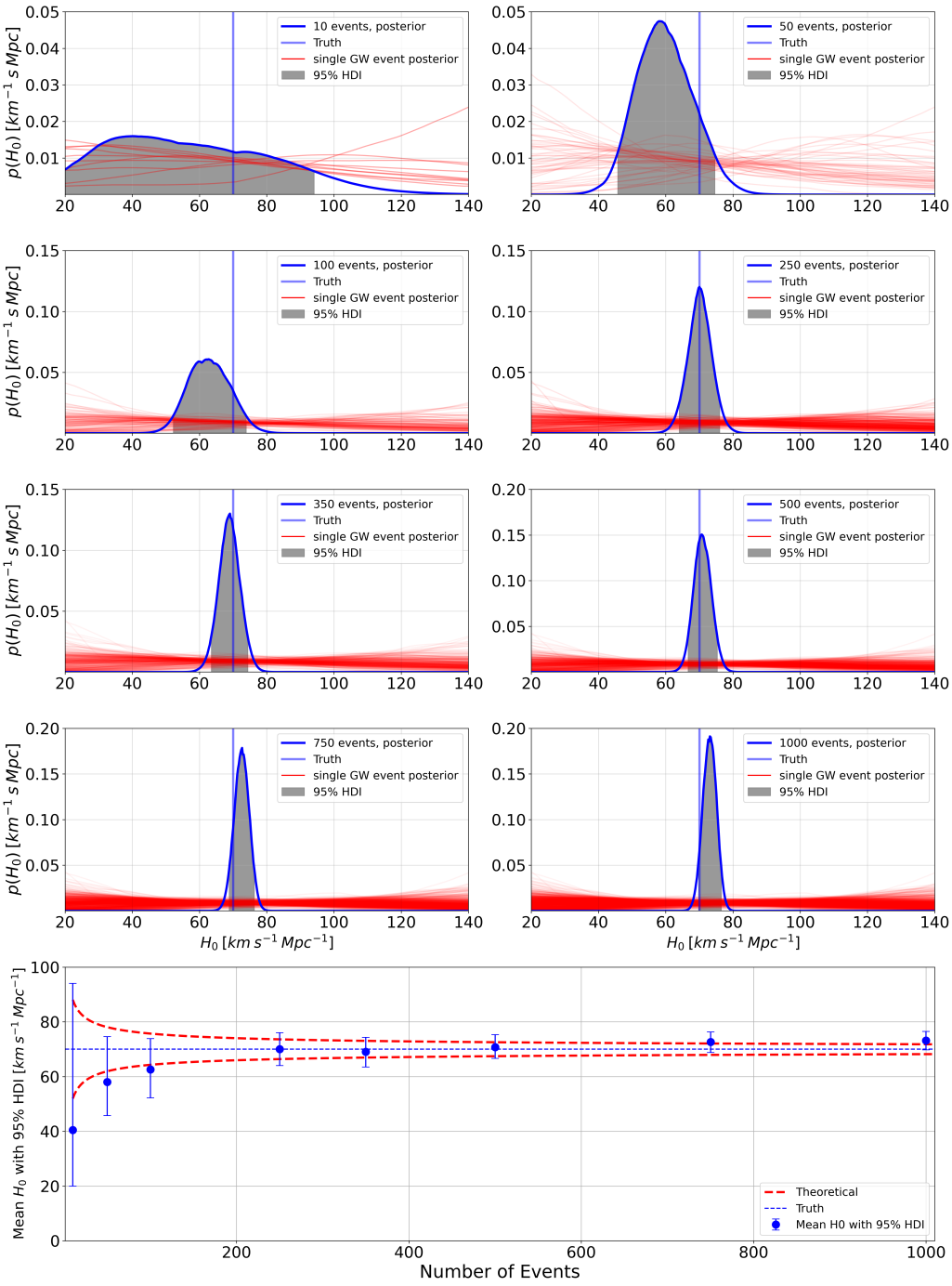

Figure 3.10: Subplots showing the combined posterior distributions for 10 to 1000 events using testing data. The true value used to generate the testing data was set at $H_0 = 70\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$. The red curves represent the posterior distributions of H_0 for each individual synthetic GW event, while the shaded grey region indicates the 95% highest density interval (HDI). The bottom subplot illustrates the mean estimates and 2σ error bars of each combined posterior as a function of the number of events. The theoretical curve, $2\sigma/\sqrt{N_{\mathrm{events}}}$, is also plotted as a dashed red line. THe blue line is the total posterior distribution over H_0.

is around 7.1%. Since this scenario represents a best-case upper limit (where all events

are perfectly constrained, which is unrealistic), the actual number of events required to constrain H_0 to below 1% would be much higher. A lower-boundary estimate suggests that at least 10,000 GW BBH events would be needed to reach sub-percent precision, uisng only BBH GW events.

3.3 Conclusion

In this chapter, we explored the sections of the `CosmoFlow` analysis of training a NF and assessing its validity through a series of tests.

After generating the dataset, a NF model is initialised for training, incorporating 14 GW parameters: luminosity distance (D_{L}), primary mass (m_1), secondary mass (m_2), detector frame masses, right ascension (α), declination (δ), primary spin magnitude (a_1), secondary spin magnitude (a_2), inclination (θ_{JN}), spin tilt angles (θ_1 and θ_2), spin phase angles (ϕ_{JL} and ϕ_{12}), polarization angle (ψ), and coalescence time (t_{geo}). Additionally, a conditional variable is included, given by the H_0 values attributed to these events. The training of the flow model, executed over 1000 epochs using the `glasflow` package and facilitated by an NVIDIA RTX G-Force 2080 GPU, is done over an estimated duration of four hours. The model then undergoes a series of validation tests, specifically the resampling test, the PP plot test and the bias test.

Initially, a resampling test was conducted to verify that sampling from the latent space, conditioned on the same dataset used for data generation, accurately reflects the target data distribution. The results of this test are shown in Fig. 3.6. The next test involves using PP plots to assess potential biases by comparing the distribution conditioned on specific H_0 values against the corresponding true values. The outcomes, presented in Fig. 3.7, reveal no inherent bias. We can also perform a PP plot for the conditional space. In this analysis, we use the single event GW parameter set from the target data and evaluate for different values of H_0. This approach allows us to estimate the posterior distribution $p(H_0|\theta, D)$ and obtain posteriors over H_0 for each individual event. Since each event has a different true value of the H_0 parameter, we can compare the posterior distribution of each event with the true values, enabling the construction of a PP plot for H_0. The results in Fig. 3.9 show an unbiased PP plot, with a p-value of 0.736.

The final test focuses on bias detection within the model. This involves generating datasets with a fixed H_0 value by creating 1000 synthetic GW events and applying Eq. (2.12), with the exception of the SNR, which is adjusted for Gaussian noise per detector. This approach allows for the combination of posteriors from multiple simulated events, as demonstrated in Fig. 3.10. The results from the bias test indicate the presence of a slight systematic bias, particularly when combining 1000 GW event posteriors over H_0. This bias can be mitigated by increasing the amount of training data and extending the number of training epochs for the NF.

Having rigorously trained the model with synthetic data generated from Chapter. 2, we proceed to apply real GW posterior samples to compute the H_0 posterior distribution for each event, considering various detector setups and PSDs.

Chapter 4

CosmoFlow: Results from GW data

Declaration: The results presented in this Chapter are derived from my first author paper [54], with additional findings included. We provide one-dimensional posteriors for both BBHs and NSBHs, and explain the inconsistencies observed with `gwcosmo`*. One significant difference, while still consistent with [54], is that at the time of producing these results, only 5 dimensions of the gravitational wave data were used, instead of the full 15 dimensional space. This limitation has since been corrected, but the results presented here are based on a 5 dimensional flow with a 1 dimensional conditional space.*

4.1 Introduction

The primary objective of the `CosmoFlow` analysis is to estimate cosmological parameters using information extracted from GW detections using NFs. In Chapter 2, we discussed the preliminary stage of this analysis, which involves training an NF model on synthetic data that represents GW events derived from specific cosmological and population priors. This approach enabled us to create a dataset with no uncertainty on the parameter values for GWs, conditional on many different realisations of the Hubble constant.

After generating the training data, a *conditional* NF was trained to evaluate the mapping function from the data space to a Gaussian latent space, as shown in Chapter 3. In alignment with the `gwcosmo` analysis in [16], only five GW parameters were used: luminosity distance, primary and secondary masses, and sky locations (right ascension and declination). The conditional data was the cosmological parameter, specifically the Hubble constant. Training the NF allows us not only to map the target data to a Gaussian latent

space and back, but also to evaluate $p(\theta|\Omega,D)$ in the numerator of Eq.(3.1), for specific values of the cosmological parameters, Ω. To remind the reader, in Sec. 2.2, we outlined the Bayesian framework used to evaluate the posterior distribution of cosmological parameters using GW data. The derived equation was

$$p(H_0|\mathbf{h},\mathbf{D},I) \propto p(H_0|I)\prod_i \left\langle \frac{p(\theta_i|D_i,H_0,I)}{p(D_i|\theta_i,I)p(\theta_i|\Omega_0,I)} \right\rangle_{\theta_i\sim p(\theta_i|h_i,I)}, \tag{4.1}$$

where h_i, D_i, θ_i are the strain data, detectability status, and the posterior samples of the ith event, respectively, and Ω_0 is the cosmological fixed values for the priors used in the PE analysis of the GW events. The trained NF provided us with a tool that enabled us to compute the numerator in Eq.(4.1), thus allowing us to perform hierarchical inference on the cosmological parameters using GW posterior samples as input. In this chapter, we will present the results obtained from using O1, O2 and O3 GW posterior samples events from BBHs to infer the Hubble constant, H_0.

Since this is a novel approach for inferring H_0 and uses ML, benchmarking is necessary to evaluate the consistency and validity of the results. Therefore, we will compare our posterior results with those obtained from `gwcosmo` during the publication of the collaborative work on constraining cosmological parameters using GW data [16]. The `gwcosmo` analysis has been tested in the past, including a Mock Data Analysis (MDA) [51] for well-localised BNSs, which showed no systematic bias. However, it is important to note that the results from the `gwcosmo` analysis are not the true posterior distributions. Matching these results only provides indirect evidence of a correct analysis for `CosmoFlow`. While `gwcosmo` has undergone some benchmarking, it has not been extensively validated to the point where its posteriors can serve as a definitive benchmark. Therefore, even if our results match those of `gwcosmo`, it only serves as indirect evidence of correctness. To date, there has not been a BBH Mock Data Challenge (MDC) with a known truth against which `gwcosmo` has been tested. A more rigorous test of our analysis (and `gwcosmo`) would involve using simulated data with known correct results. This would allow us to argue for the absence of bias and pass PP-plot tests. Although we approach this, with the bias test in Chapter 3, we fall short of actually simulating posterior distributions for BBH events. Before using `CosmoFlow` for cosmological analysis, it is essential to understand the nature of the data being analysed.

4.2 GWTC data

On September 14, 2015, at 09:50:45 UTC, both detectors of the LIGO observatory detected a transient GW event, designated GW150914, with a delay of only 7 milliseconds between them. This event marked the first direct observation of GWs, providing a groundbreaking confirmation of Einstein's theory of general relativity. The strain data, h, from this event was then used to estimate the dynamics and underlying physics of the black hole merger, allowing us to infer the system's properties. The detection showed a matched-filter SNR of 24. A high SNR, such as the value of 24 observed for GW150914, indicated a very strong signal standing out clearly above the noise, making it highly reliable. Additionally, the event had a false alarm rate (FAR) of less than one event per 203,000 years. The FAR quantifies the expected rate at which noise alone would produce a signal that mimics a true GW event. Mathematically, FAR can be expressed as:

$$\mathrm{FAR} = \frac{N_{\mathrm{false}}}{T} \tag{4.2}$$

where N_{false} is the number of false positives detected in a given time period T. In this context, a false positive is defined as a non-astrophysical candidate that equals or exceeds the detection statistic in question. A FAR this low as one over 203,000 implies an extremely rare probability of a false detection, translating to a statistical significance exceeding 5.1σ. This high level of statistical significance further confirms the authenticity of the detection, ensuring that the observed signal is indeed a genuine GW event rather than a random noise fluctuation [30].

Following the identification of a GW signal such as GW150914 via matched-filtering [30], the next phase involves deducing the source's physical attributes. This is known as parameter estimation (PE) and is done using sophisticated Bayesian inference methods. As outlined in Chapter 1 Sec. 1.4, Bayesian inference evaluates the probability of certain parameters given the observed data and prior knowledge of the parameters' probable values. Detection is determined by matched-filtering, which involves comparing the observed data with theoretical templates to find the best fit. Once a signal is detected, some limited information from the matched-filtering analysis, such as the time of arrival and the nature of the event (e.g., BBH, NSBH or BNS), is used to define the prior space for the Bayesian analysis. Bayesian inference is then employed to deduce the physical parameters that generated the signal. Within this framework, the `bilby` [143, 144, 145] Python package is commonly used for analysing posterior estimations within the LVK collaboration.

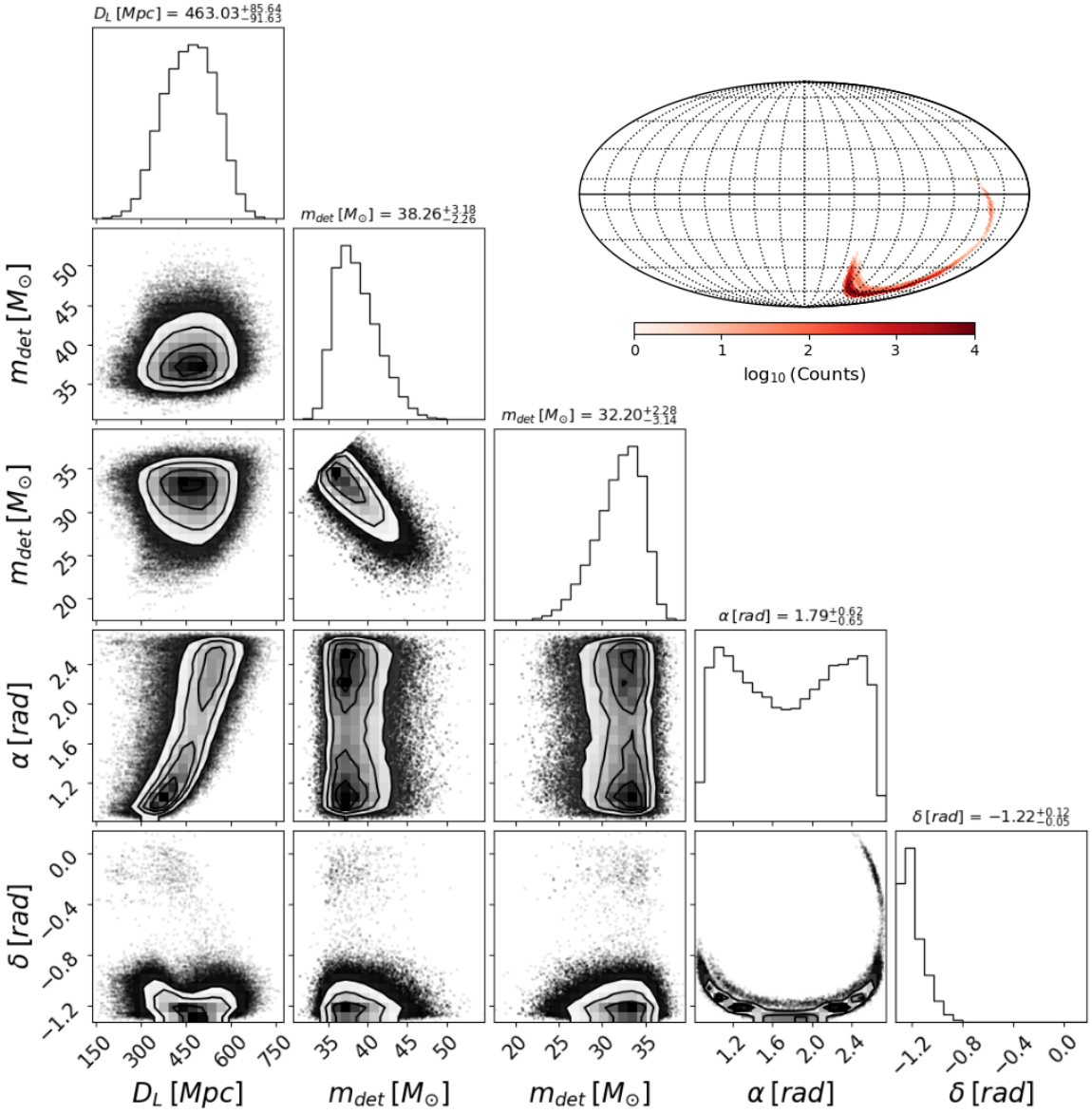

Figure 4.1: Corner plot of the sampled joint posterior distribution showing luminosity distance, detector frame masses, and sky location (right ascension and declination). The marginalized distributions' titles show the mean value with 16% and 84% quantiles, and the 2D histograms display the 11.8%, 39.3%, 67.5%, and 86.4% credible regions. The remaining 10 signal parameters have been marginalized over and are not shown here. (Top Right): Bin counts per pixel plot over a Mollweide projection of the sky location posterior samples of GW150914 [48, 86, 88]. The colorbar shows the $\log_{10}$ scale of the bin counts, using a resolution of $3.36\,\text{deg}^2$ per pixel

As discussed in Sec. 1.5, `bilby`, a Python-based package that serves as a high-level interface for GW parameter estimation (PE), is used to process data. It simplifies the workflow by providing a wrapper around various stochastic sampling algorithms to determine the posterior distributions of parameters that define the waveform template. Traditional methods like MCMC face challenges with increasing problem dimensionality. However, more advanced approaches, such as NS, implemented via the `dynesty` [146] code, offer

a different strategy for exploring the joint posterior distribution of the parameter space. Both MCMC and NS methods have their own advantages and limitations, particularly in high-dimensional settings, and remain widely used for GW PE [123, 146]. More recent machine learning enhanced methods, like Nested Sampling with Artificial Intelligence (`nessai`), employ NF to efficiently sample from the prior within iso-likelihood contours, thus expediting the process of sampling the joint posterior distribution [37].

An example result of the PE analysis of GW150914 is partially shown in Fig. 4.1. The corner plot displays the posterior distributions of five parameters that will be used as target inputs to the `CosmoFlow` analysis [1]. However, the full PE analysis considers all 15 parameters, as described in Tab. 2.1. The figure also includes a Mollweide projection of the sky location posterior samples, showing bin counts per pixel. This skymap illustrates the sky coverage of this GW event, which spans approximately 200 square degrees, and indicates how much of the galaxy catalogue GLADE+ is covered by the event. This plot highlights the complex nature of these parameter data spaces, which should be taken into consideration when using these samples to evaluate posterior distributions over H_0.

We gathered the posterior samples from 42 BBH events observed during the O1, O2, and O3 detector runs, cataloged in GWTC-2.1 and GWTC-3, for our `CosmoFlow` analysis. Events were selected with an SNR greater than 11, in line with [16]. It is important to note that the GW data collected from GWTC-2.1 and GWTC-3 includes events detected during the O1 through O3 runs, each with different detector setups. The detectors exhibited varying noise characteristics between runs, and even between events within the same run, due to ongoing improvements and maintenance. Furthermore, not all detectors were operational for every event; hence, each detection was made with a different network configuration, involving either two or three detectors. We do not consider any single-detector events. This variability in the detector network and noise characteristics is summarized in Tab. 4.1.

To ensure an unbiased analysis, we generated training data to reflect various combinations of detector configurations across different observing runs. Specifically, we considered the following combinations: O1 [H, L] (Flow 1), O2 [H, L] (Flow 2), O2 [H, L, V] (Flow 3), O3 [H, L] (Flow 5), O3 [H, V] (Flow 7), O3 [L, V] (Flow 6), and O3 [H, L, V] (Flow 4). Each entry in the brackets indicates whether a particular detector was operational (included)

1. As noted at the beginning of this chapter, the results presented in Chapter 4 for `CosmoFlow` are obtained using NF trained on 5 GW parameters. For comparison, results using the full 14D parameter space of the GW events, excluding the phase parameter, are provided in Appendix B, where posteriors obtained with the 5D and 14D conditional NFs are analysed side by side.

Observing Run	Event	Detectors	Network SNR	Flow
O1	GW150914_095045	H, L	24.4	Flow 1
	GW151226_033853	H, L	13.1	Flow 1
O2	GW170104_101158	H, L	13.0	Flow 2
	GW170608_020116	H, L	15.4	Flow 2
	GW170809_082821	H, L, V	12.4	Flow 3
	GW170814_103043	H, L, V	16.3	Flow 3
	GW170818_022509	H, L, V	11.3	Flow 3
	GW170823_131358	H, L	11.5	Flow 2
O3a-O3b	GW190408_181802	H, L, V	14.8	Flow 4
	GW190412_053044	H, L, V	19.7	Flow 4
	GW190503_185404	H, L, V	12.8	Flow 4
	GW190512_180714	H, L, V	12.4	Flow 4
	GW190513_205428	H, L, V	12.9	Flow 4
	GW190517_055101	H, L, V	11.3	Flow 4
	GW190519_153544	H, L, V	13.9	Flow 4
	GW190521_030229	H, L, V	14.4	Flow 4
	GW190521_074359	H, L	24.7	Flow 5
	GW190602_175927	H, L, V	12.6	Flow 4
	GW190630_185205	L, V	15.2	Flow 6
	GW190701_203306	H, L, V	11.9	Flow 4
	GW190706_222641	H, L	12.7	Flow 5
	GW190707_093326	H, L	13.2	Flow 5
	GW190708_232457	L, V	13.1	Flow 6
	GW190720_000836	H, L, V	11.6	Flow 4
	GW190727_060333	H, L, V	12.1	Flow 4
	GW190728_064510	H, L, V	13.4	Flow 4
	GW190828_063405	H, L, V	16.6	Flow 4
	GW190828_065509	H, L, V	11.1	Flow 4
	GW190910_112807	L, V	13.4	Flow 6
	GW190915_235702	H, L, V	13.1	Flow 4
	GW190924_021846	H, L, V	13.0	Flow 4
	GW191109_010717	H, L	15.8	Flow 5
	GW191129_134029	H, L	13.3	Flow 5
	GW191204_171526	H, L	17.1	Flow 5
	GW191216_213338	H, V	18.6	Flow 7
	GW191222_033537	H, L	12.0	Flow 5
	GW200112_155838	L, V	17.6	Flow 6
	GW200129_065458	H, L, V	26.5	Flow 4
	GW200202_154313	H, L, V	11.3	Flow 4
	GW200224_222234	H, L, V	19.2	Flow 4
	GW200225_060421	H, L	13.1	Flow 5
	GW200311_115853	H, L, V	17.6	Flow 4

Table 4.1: Summary of BBH GW events and corresponding detectors by observing run for a total of 42 detected GW events. Detection thresholds were determined based on a matched-filter SNR above 11 [16]. The table comprises five columns: the observing run, the event names, the detector configuration during the event detection, the network matched-filter SNR, and the flow used. The flows are defined as follows: Flow 1 - HL O1, Flow 2 - O2 HL, Flow 3 - O2 HLV, Flow 4 - O3 HLV, Flow 5 - O3 HL, Flow 6 - O3 LV, Flow 7 - O3 HV.

or not (excluded) during the detection of GW events. These combinations resulted in seven distinct datasets, and consequently, seven distinct NFs, each representing different configurations of operational detectors during the O1, O2, and O3 observing runs, as described in Tab. 4.1.

Initially, we considered generating a single dataset and using "one-hot encoding" to include information about the detector setup and the specific PSD from each detector run. One-hot encoding, a machine learning and data preprocessing technique, transforms categorical variables into a numerical format, enabling algorithms to interpret and process the data more effectively for predictions [147]. In our case, one-hot encoding would differentiate the data based on different detector configurations, assigning a binary value to each detector: 0 if it was off and 1 if it was operational. For example, for an HLV dataset, the extra features would be $[1,1,1]$, and for an HV dataset, it would be $[1,0,1]$. We also considered adding a fourth one-hot variable to represent the observing run (O1, O2, O3), or incorporating additional conditional input to represent the PSD of each run or event. Mathematically, this would modify the NF model to learn the distribution $p(\boldsymbol{\theta}|\boldsymbol{\Omega},D,\mathbf{C})$, where $\mathbf{C}$ represents the conditional one-hot encoded features, including detector configuration and PSD. The method involved generating entire datasets, each having the same number of data points, and then training the flow with these extra conditional features. For instance, with HLV binary values, we would have a total of 4 dimensions in the conditional space. In our initial test, we chose to fix the PSD specific to the detector run. However, this approach was unsuccessful due to the complexities involved in setting up the problem. Specifically, it was challenging to evaluate the probability $p(\boldsymbol{\theta}|\boldsymbol{\Omega},D,\mathbf{C})$ for varying values of H_0 while keeping $\mathbf{C}$ fixed for specific events. The results showed inconsistencies when producing the posterior distributions over H_0, including the presence of unphysical features. Future research will explore this area further to address these challenges. This approach would have allowed us to generate a single training dataset per observing run, and hence a single flow for each observing run, or even better, for all configurations, making the process more efficient. However, instead of using one-hot encoding, we ultimately decided to create separate datasets for each combination of detectors and observing runs. This approach resulted in more manageable and distinct datasets, avoiding potential biases and complexities that could arise from combining different configurations and PSDs into a single dataset, while also simplifying the training process for each NF, even though this required training seven separate NFs.

Therefore, we trained a separate conditional NF model for each detector configuration and observing run, rather than a single NF model with multiple conditional inputs. This approach allowed us to maintain the integrity and accuracy of our analyses while accommodating the varied detector configurations and noise characteristics across different

runs. Then seven NF models were trained, each with a six-dimensional dataset: five dimensions for the target data (luminosity distance, detector frame masses, right ascension, and declination) and one dimension for the conditional input (the Hubble constant). In the results presented in [54], each NF was trained with one million data points. However, as we will explain in the next section, more data was needed to better capture the catalogue features.

4.3 Results

4.3.1 Pre-analysis sanity check

Before using the trained NFs to estimate the single event posterior distributions, we should perform a sanity check on the datasets generated in Chapter 2. An effective way to analyse the data is by using the GW posterior sample distributions to sample data points from the training data. These samples are weighted by the probability distributions of the posterior, effectively cutting out a segment of the training data. We then examine the associated H_0 from each sampled data point. This process yields an H_0 distribution, which we can compare with the results from `gwcosmo`.

Initially, we attempted to approximate the posterior parameter space of the GW event by drawing simple bounding boxes. However, this approach proved inadequate for accurately capturing the complexities of the posterior distributions, particularly for parameters such as sky location and primary and secondary masses. These parameters are often highly correlated and exhibit non-trivial, irregular shapes that cannot be well-represented by simple bounding boxes, as shown in Fig. 4.1.

To address these limitations, a more effective approach is to use a non-conditional NF. This involves training the NF on the five-dimensional parameter space of the GW posterior samples, which includes luminosity distance, primary and secondary masses, and sky location. An intuitive way to understand this is to think of the posterior distributions as "cookie cutters", slicing through the generated training data to create shapes defined by equal-probability contours corresponding to $p(\boldsymbol{\theta}|h)$, the posterior distributions of the GW parameters. The NF models these complex shapes more accurately than simple bounding boxes, capturing their full structure and correlations.

With a NF trained on the posterior distribution data points we then follow specific steps to sample the training data using the GW posterior distribution. Firstly, we compute the log probability of each data point x_i in the training dataset by passing it through the NF. The log probability $\log p_\theta(x_i)$ is given by

$$\log p_\theta(x_i) = \log p_z(f_\theta(x_i, \phi)) + \log \left| \det \left(\frac{\partial f_\theta(x_i, \phi)}{\partial x_i} \right) \right|, \tag{4.3}$$

where p_θ is the probability distribution modeled by the NF approximating the posterior distribution of the GW parameters, $z_i = f_\theta(x_i, \phi)$ is the transformation defined by the NF with ϕ being the trainable parameters of the NF model, p_z is the probability density of the latent space (a standard normal), and $\left| \det \left(\frac{\partial f_\theta(x_i, \phi)}{\partial x_i} \right) \right|$ is the Jacobian determinant of the transformation. Next, by setting a specific threshold τ, we retain only those data points x_i whose log probability $\log p_\theta(x_i)$ is above the threshold:

$$\{x_i \mid \log p_\theta(x_i) \geq \tau\} \tag{4.4}$$

This process effectively segments the data set, keeping the data points with the highest relative log probability values. The threshold τ can be chosen to retain a desired proportion of the data, which in this case was set to $\tau = -3.0$. The retained data points can be visualised as being cut out by the posterior sample distributions of the event, similar to using "cookie cutters". This is because the NF has been trained to model the posterior distribution, and high log probability values indicate that the retained data points are more consistent with the posterior.

An important sanity check we wish to highlight is the results from GW170814. This particular GW BBH event is noteworthy due to its proximity, $\sim$ 540 Mpc [148], which allows significant overlap between the GLADE+ catalogue and the 3 dimensional space of the event. This overlap enables us to observe catalogue features in the results of `gwcosmo` [16]. Therefore, we present results for GW170814, although we have performed this sanity check for all 42 BBH events. The training process consists of 100 epochs with a simple network configuration: 12 neurons per layer, 3 layers, and 2 residual blocks, with a learning rate of 0.001, using a Tesla V100 GPU, with a total of 197,034 posterior samples from GW170814. The results of this method are illustrated in Fig. 4.2, where the corner plot shows a remarkable match between the posterior distribution of event GW170814 and the trained NF. In Fig. 4.3, the training and validation loss are plotted alongside the average KL divergence from comparing each marginalised latent space dimension for the NF with

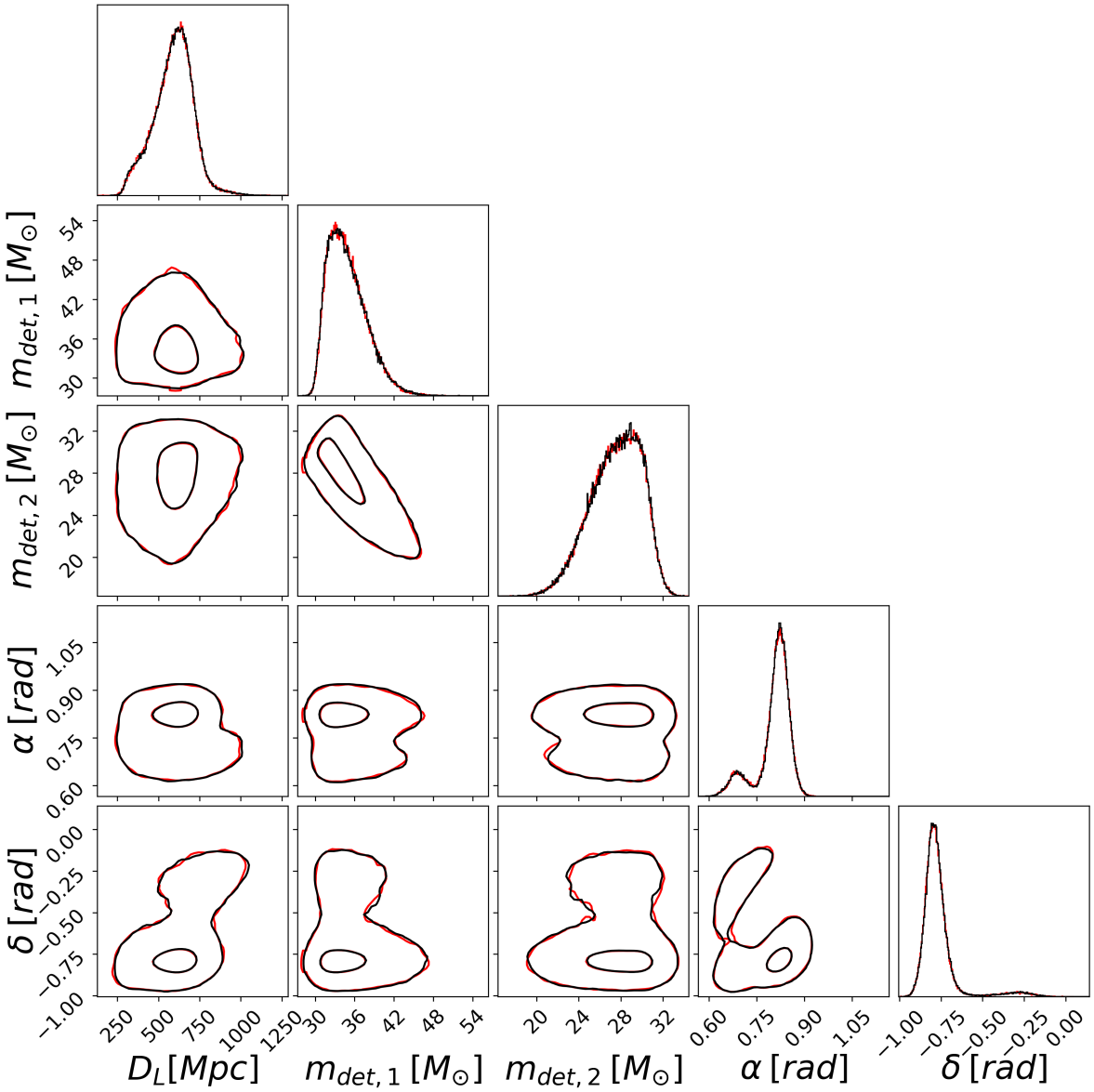

Figure 4.2: Results of the NF method for event GW170814, showing the posterior sample distribution (in black) overlaid with samples from the NF (in red). The contour levels are 50% and 90% of the total enclosed data.

a standard unit Gaussian, showing how quickly the model can achieve low loss values after just 100 epochs. Finally, we sample from the segmented training data to obtain a set of H_0 values. The resulting plot, shown in Fig. 4.4 provides insights into the H_0 distribution as inferred from the posterior sample distributions of the event.

In Fig. 4.4, we present the results of using the NF trained to model the GW parameter posterior distribution of the GW170814 event. The figure shows two histograms representing the contributions to H_0 from sources within the catalogue and those outside the catalogue. These distributions are mathematically described, using `gwcosmo` notation, as $p(H_0|\theta_{\rm GW}, D, G)$ and $p(H_0|\theta_{\rm GW}, D, \bar{G})$, where G and $\bar{G}$ represent the conditions of being

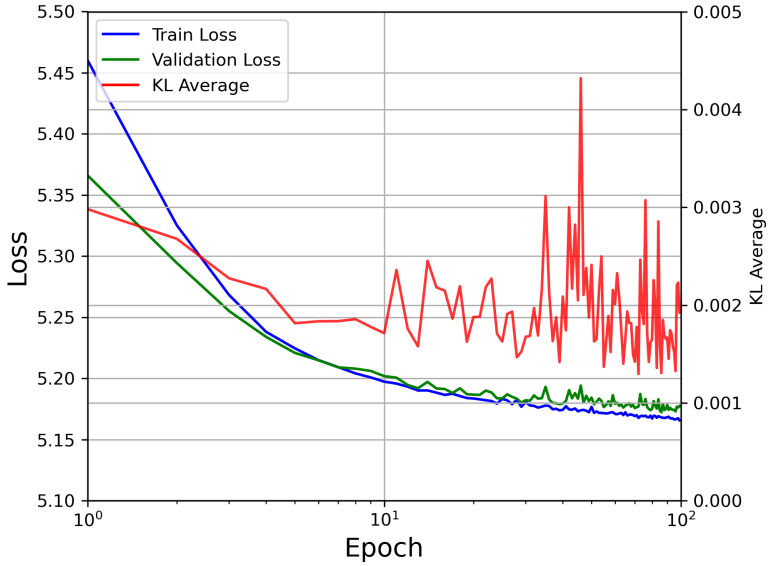

Figure 4.3: Results of the NF method for event GW170814, displaying the training and validation loss over epochs, along with the average KL divergence over all five latent space dimensions, each compared to a Gaussian distribution.

in the catalogue and not in the catalogue, respectively and $\boldsymbol{\theta}_{\mathrm{GW}}$ are the GW posterior samples of GW170814. Overlaid on the histograms are the `gwcosmo` results for this specific event [16], allowing us to compare the normalised H_0 distribution from the training dataset with the reference results. The term "empty catalogue"in the `gwcosmo` results refers to the scenario where no prior knowledge of the galaxy distribution is used.This results in a broader and less constrained posterior distribution for H_0 in the absence of additional EM information. In contrast, incorporating the galaxy catalogue leads to a sharper and more constrained posterior. The difference between the `gwcosmo` results with and without catalogue information shows the impact of including prior knowledge about galaxy distributions. When combining the data points from the training set, both from within and outside the catalogue (the green histogram), we see that the overall combined histogram matches well, within some uncertainty, to the `gwcosmo` posterior distribution with catalogue assumption. Specifically, if we had not included the in-catalogue information from GLADE+, the feature observed on the left side, around $H_0 = 35\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$, would not be captured by `CosmoFlow`. This test demonstrates the importance of including catalogue information for inferring H_0 and shows that the data generation process has successfully captured the over density of galaxies in the GLADE+ catalogue.

Although we managed to sample 3 million data points for the training routine for this specific detector configuration and observing run, after passing the data through the trained NF model, as described in Eq. (4.3), over the GW event, only 507 points were retained. This highlights the challenge of sampling the data space at a high enough density to cover all detected events. However, even with the relatively low number of samples, the contri-

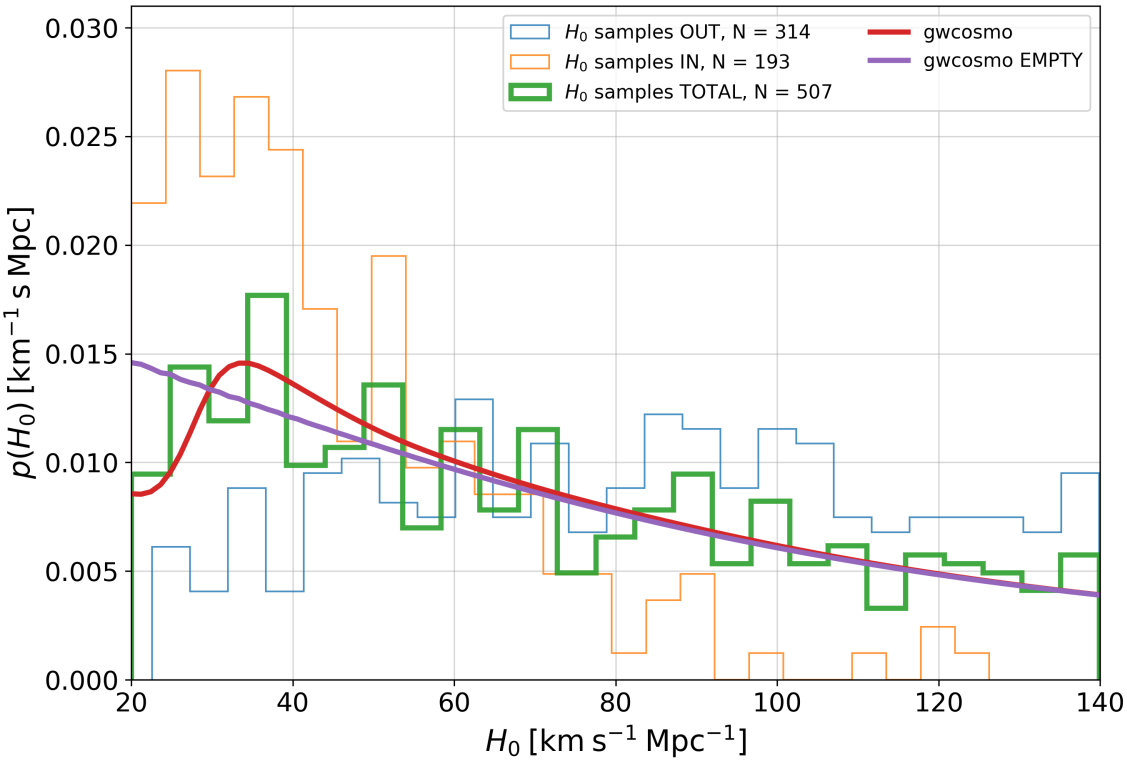

Figure 4.4: Histogram of the Hubble constant (H_0) samples for the GW170814 event, obtained using the cookie cutter method. The histogram is divided into contributions from out-of-catalogue events (blue), in-catalogue events (orange), and the total samples (green). Overlaid are the results from `gwcosmo` (red solid line) and `gwcosmo EMPTY` (no catalogue information) (purple line) for this specific event.

butions from within and outwith the catalogue are clearly visible, showing a peak in the total histogram that matches the results from `gwcosmo` when the GLADE+ catalogue is used. It is important to note that the features from the galaxy catalogue typically manifest in the lower part of the H_0 posterior distributions. This is because the galaxy catalogue provides additional information about the spatial distribution of galaxies, which helps to constrain the values of H_0 more effectively. When incorporating this prior knowledge, the posterior distribution tends to show more pronounced features at lower H_0 values, reflecting the regions where the galaxy catalogue GLADE+ information is most influential. This matching peak validates both the data generation process and the integration of EM information from the GLADE+ catalogue, showing the result is not solely driven by population assumptions. If the galaxy catalogue information from GLADE+ had not been included, as demonstrated by the `gwcosmo EMPTY` result in Fig.4.4, the distinct feature around $H_0 = 35\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$ would have been missed. This confirms that incorporating the catalogue information following the approach outlined in the data generation section in Chapter 2 effectively integrated EM data from GLADE+.

4.3.2 H_0 posterior distribution, BBH result

After visually validating the training data, we perform the calculation as described in Eq.(4.1) to evaluate the posterior distributions of the Hubble constant (H_0) for individual GW events detected during the O1, O2, and O3 observing runs, as listed in Table 4.1. We begin by presenting the results for the BBH events, followed by an analysis of the NSBH events. We do not perform any analysis regarding BNS events. This is because the `gwcosmo` analysis can evaluate the posterior for bright sirens relatively quickly, making it unnecessary to speed up this particular analysis using `CosmoFlow`. Bright sirens are typically well-localized in the sky and have EM counterparts, which facilitates a faster and more straightforward posterior evaluation with `gwcosmo`. Future work will look into having `CosmoFlow` perform the same bright siren analysis.

After generating datasets for each detector configuration and run, 7 distinct flow models were trained on these specific datasets, each with 3 million data points. The permutations and combinations of the detector setups for each event are detailed in Table 4.1. For each event, 1000 five-dimensional posterior samples from the GW event parameters were used. The resulting curves for each sample were then averaged together to produce a single posterior distribution for each event. Using each trained flow model, Eq. (4.1) was evaluated over a range of 500 H_0 values, evenly spaced, between 20 to 140 $\mathrm{km\,s^{-1}\,Mpc^{-1}}$, for each GW posterior sample. The plot in Fig. 4.5 compares our results using `CosmoFlow` with the `gwcosmo` results under both catalogue and empty catalogue assumptions. The red curves represent the single posterior sample evaluation of Eq. (4.1), while the blue curve shows the evaluation of Eq. (4.1) averaged over 1000 GW posterior samples. Several key observations can be made from this plot. Firstly, the variation between the red likelihood curves indicates the spread and uncertainty in the H_0 estimation for the event GW170814. Our `CosmoFlow` result aligns closely with the `gwcosmo` catalogue result, more so than with the empty catalogue result. To quantify the closeness between the various models, we will use the JS divergence.

In our comparison, the JS divergence between our `CosmoFlow` result and the `gwcosmo` catalogue result is 25 millinats, while the divergence between our result and the `gwcosmo` empty catalogue result is 35 millinats, suggesting that the `CosmoFlow` model has effectively learned the galaxy catalogue information. We then repeat this process for each event listed in Tab. 4.1. Figure 4.6 presents the results of the individual one-dimensional posterior distributions from `CosmoFlow` and `gwcosmo` for each event. Starting from GW150914 and

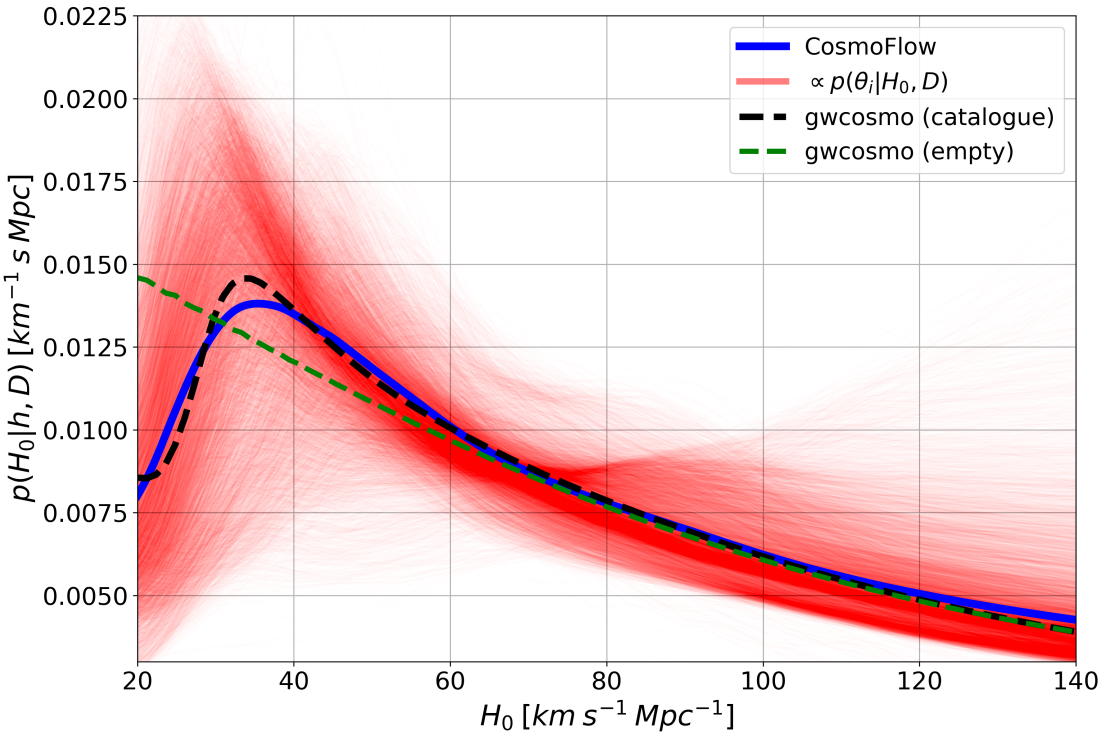

Figure 4.5: The plot compares the posterior distributions of the Hubble constant H_0 derived from different methodologies. The red curves represent normalised likelihoods of H_0 for each GW sample, proportional to $p(\theta_i|H_0, D)$. The blue line shows the `CosmoFlow` posterior, while the black and green dashed lines represent the `gwcosmo` posteriors with and without the GLADE+ catalogue, respectively. All analyses assume uniform priors on H_0. The JS divergences are 25 millinats (`CosmoFlow` vs. `gwcosmo` with catalog) and 34 millinats (`CosmoFlow` vs. `gwcosmo` without catalogue).

in chronological order, the posterior distributions over H_0 for both `CosmoFlow` and `gwcosmo` are shown in Fig. 4.6. The results show a good comparison between `CosmoFlow` and `gwcosmo` with EM catalogue information, with similarity ranging from from 0.03 millinats to 26.27 millinats with GW190706_222641.

The results also reveal some catalogue features, as explained in Sec. 4.3.1, although not perfectly captured by `CosmoFlow` compared to `gwcosmo`. Galaxy catalogue features seem to be captured by `CosmoFlow` for some events, such as GW150914_095045, GW170814_103043, GW190521_030229, but not so clearly in other events where catalogue features are more evident in `gwcosmo`, such as GW190412_053044. This discrepancy raises questions about the validity of the dataset generated using the in-catalogue procedure. This issue warrants specific investigation, as multiple factors could contribute to the discrepancy. Potential causes include imperfectly described GW-selection effects, as modeled by the MLP described in Chapter 2, or the EM selection effects, particularly in how the galaxy catalogue

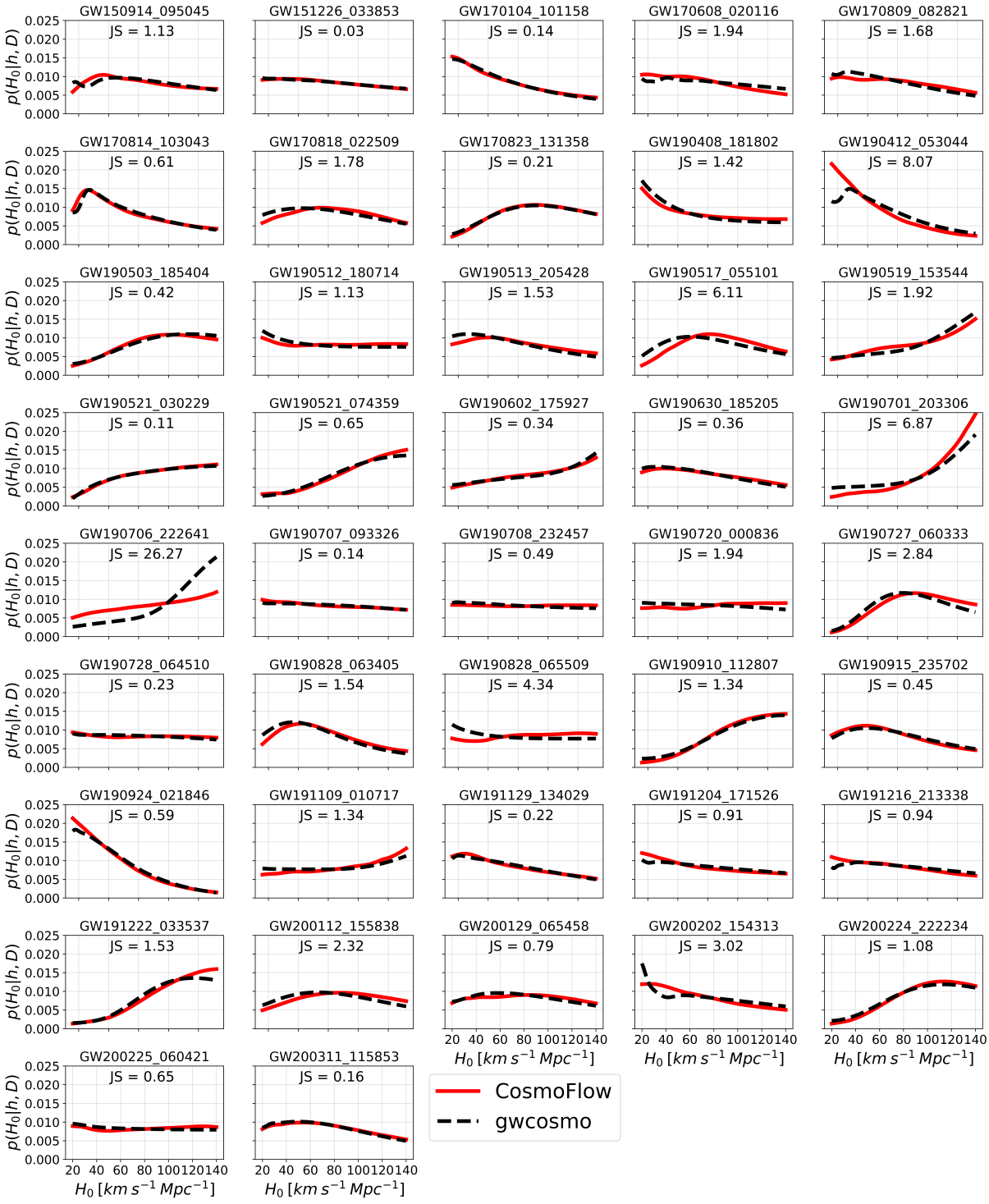

Figure 4.6: Comparison of Hubble constant H_0 posterior distributions for individual GW events from O1,O2 and O3 observing runs over the SNR threshold runs gravitational wave events using `CosmoFlow` (red solid lines) and `gwcosmo` (black dashed lines). The JS divergence is noted for each event, indicating the similarity between the two posteriors for each event.

is integrated within the `CosmoFlow` analysis. Nevertheless, the `CosmoFlow` analysis effectively captures the population-driven approach performed by `gwcosmo`, as outlined in [16]. A fully population-driven H_0 inference implies an estimation of H_0 based solely on the stat-

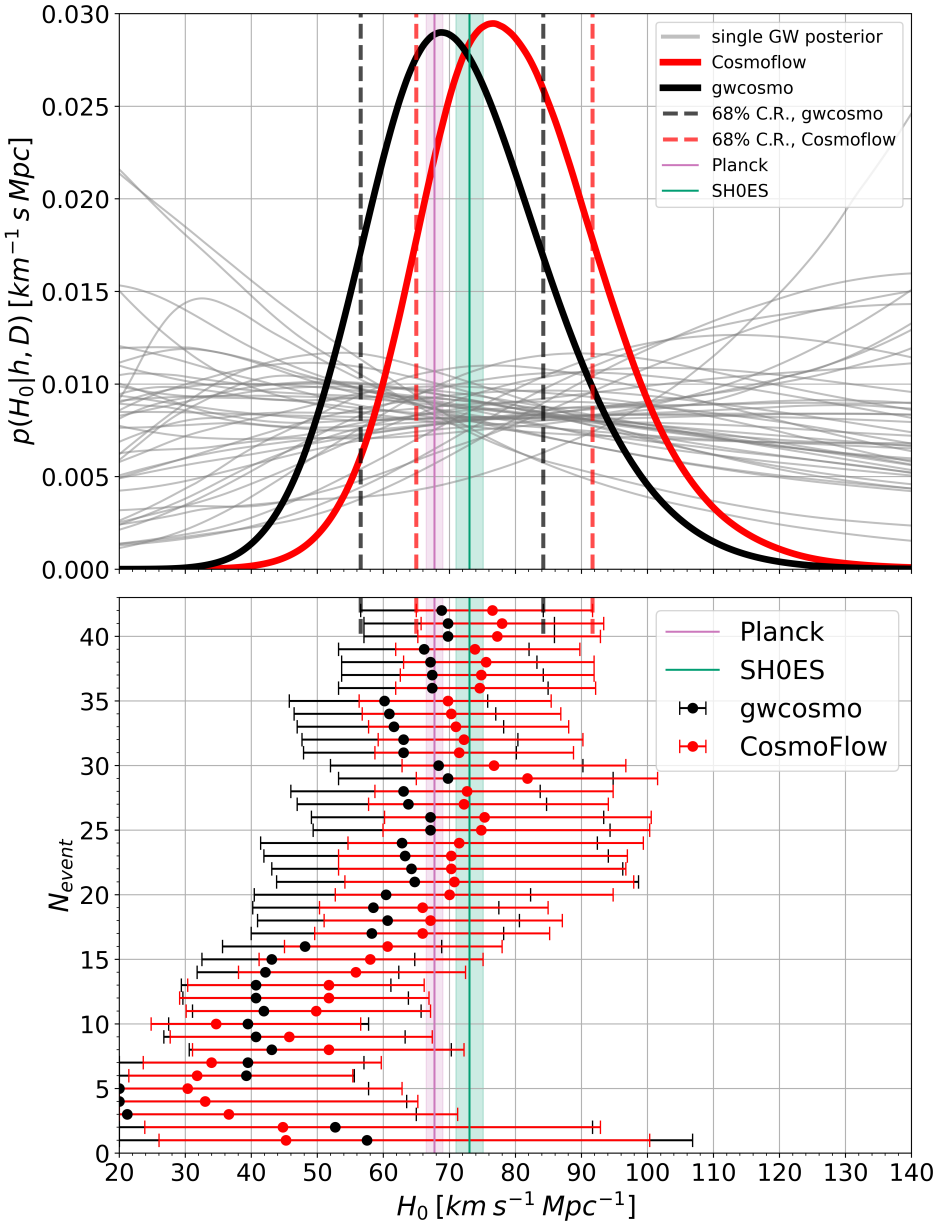

Figure 4.7: Comparison of Hubble constant H_0 estimations and posterior distributions using the `CosmoFlow` and `gwcosmo` methods. Top plot: Posterior distributions of H_0 for `CosmoFlow` (red solid line) and `gwcosmo` (black dashed line) overlaid with individual single GW event posteriors (gray lines). The 68% minimal credible region bounds are indicated by dashed vertical lines. Vertical lines also mark the Planck (pink) and SH0ES (green) measurements, with shaded regions representing their 2σ confidence intervals. Bottom plot: The cumulative combined measurement uncertainty on H_0 as a function of number of events in chronological order, with the center dot representing the mean of the combined distribution after combining N_{events} and the error bar is the 68% minimal confidence range. The Planck and SH0ES measurements are again shown as vertical lines with shaded regions for their 2σ confidence intervals.

istical properties and distributions of the GW events themselves, without incorporating external data such as EM information from galaxy catalogues. The population information

side of the analysis tends to agree with the `gwcosmo` population-driven analysis, specifically by matching the overall trends for each posterior distribution over H_0. This agreement is particularly evident in events with relatively low galaxy catalogue contributions, such as GW190910_112807 and GW200224_222234 [16].

We also consider the discrepancy between the results of `CosmoFlow` and `gwcosmo` for GW190706_222641. This could be due to many reasons, but upon close inspection, the posterior distribution for this event over the luminosity distance gives a measurement of 3630^{+2600}_{-2000} Mpc, placing this event at a very high distance where catalogue contributions are very low, with $p(G|z,H_0) = 0$ [16]. Therefore, the issue here could be associated with the selection effects implemented in the data generation of `CosmoFlow`. Specifically, the accuracy of the MLP in evaluating the matched-filtering SNR of the event, and since this event was predicted to have a network SNR of $13.4^{+0.2}_{-0.4}$, it lies within the uncertainty range of our capability to discern events above or below the SNR threshold of 11. However, other events, such as GW190720_000836, show less discrepancy between the two posteriors despite having a matched-filter network SNR of 11.6. Therefore, we do not have a definitive answer to why this discrepancy is present. Future work will investigate this aspect further to understand the nature of the mismatch between `CosmoFlow` and `gwcosmo`.

Since the individual posterior distributions of the GW events over H_0 in Fig. 4.6 are independent of each other, we can combine them into one full combined posterior distribution. This is shown in Eq. (4.1), where we take the product of each likelihood and multiply it by the prior once. However, since we enforced the prior to be uniform over the entire range of $[20, 140]$ km s^{-1} Mpc^{-1}, the prior becomes a proportionality constant.In Fig. 4.7, the result of combining the individual H_0 posteriors is shown. Our results demonstrate that the combined posterior obtained through `CosmoFlow` is largely consistent with the results from `gwcosmo`, although a small shift to higher H_0 values is observed. This shift suggests that while the overall agreement between the two methods is strong, there are both statistical and potentially systematic differences influencing the results. Additionally, with the current number of events, it is premature to determine whether the data favors the late-type or early-type measurements of H_0 as given by Planck [5] or SHOES [149]. The values obtained from the combined posterior distributions are as follows:

- `gwcosmo`: $H_0 = 68.72^{+15.64}_{-12.41}\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$
- `CosmoFlow`: $H_0 = 76.51^{+15.15}_{-11.54}\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$.

The uncertainties on these measurements were obtained by estimating the highest density interval, covering 1σ (68%) of the distribution. While both methods provide H_0 estimates within overlapping uncertainty ranges, the results from `CosmoFlow` consistently show a shift towards higher values. This discrepancy underscores the importance of investigating both statistical effects and potential systematic biases inherent in the `CosmoFlow` analysis.

4.3.3 NSBHs results

The methodology outlined for computing the posterior distribution over H_0 for BBH GW events can also be applied to NSBHs, with slight adjustments in the prior distributions, aligning with the methodology from [16]. During the O3 observing run, three events were detected with an estimated secondary mass within the typical neutron star mass range (approximately 1.1 to 2.5 solar masses). The specific events referred to are GW190814_211039 [50], GW200105_162426, and GW200115_042309 [150]. A table of the main characteristics of the NSBH events are listed in Tab. 4.2. To compute the posterior distribution over the NSBH events, we regenerated the training data to account for events that reflect the nature of NSBHs. Specifically, the prior on the secondary mass was set to be uniform between $[1,3]M_{\odot}$. This ensures that all simulated events during the data generation process reflect the characteristics of NSBHs, as only such events are considered during the signal detection stage of data generation. The optimal SNR approximation using the MLP was retrained to accommodate the new prior distribution, as the previous prior used for estimating the optimal SNR of BBHs had a lower boundary for the secondary mass set at $4\,M_{\odot}$. This made the MLP trained for BBHs unsuitable for the new configuration. We kept the same waveform approximant, IMRPhenomXPHM [132].

Table 4.2: Summary of 5 parameters of the NSBHs events detected during the O3 observing run with 90% credible regions. The table includes the primary and secondary masses, sky localization area, matched-filter detector network SNR, and luminosity distance (D_L) for each event [86, 150].

Event	$m_{1,\mathrm{det}}$ $(M_{\odot})$	$m_{2,\mathrm{det}}$ $(M_{\odot})$	$\delta\Omega$ (deg^2)	**SNR**	D_L (Mpc)
GW190814_211039	24^{+1}_{-1}	$2.7^{+0.1}_{-0.1}$	20	22.2	241^{+45}_{-41}
GW200105_162426	$8.9^{+1.5}_{-1.2}$	$1.9^{+0.2}_{-0.3}$	7000	13.9	280^{+110}_{-110}
GW200115_042309	$5.7^{+2.1}_{-1.8}$	$1.5^{+0.3}_{-0.7}$	700	11.3	300^{+200}_{-100}

For the GW190814 event, the sky map resolution of the EM catalogue GLADE+ was set to `NSIDE` $= 128$, corresponding to a pixel size of $0.2\,\mathrm{deg}^2$. This higher resolution was used to accommodate the more precise sky localization for this event, which was ten times smaller than other detected events. In comparison, for the BBH analysis, we used `NSIDE` $= 32$, corresponding to $3.36\,\mathrm{deg}^2$ per pixel. To align with the methodology from [16], GLADE+

was divided into 196,608 pixels, and the magnitude threshold map was generated accordingly using the K band (see Chapter 2 for more details). However, sampling over the entire sky proved inefficient due to the highly localized sky location of GW190814. This targeted data generation ensured the dataset was optimised for analysing this specific event. We restricted the sampling of sky coordinates to only cover the GW190814 event, thus making it a targeted data generation. Given that the total sky coverage is $41,253\ \mathrm{deg}^2$, restricting data generation to the $20\ \mathrm{deg}^2$ area of the event allows us to maximise the number of samples obtained in each run by 2300 times. By focusing the training of the NF on this smaller, highly localized sky region, we force the NF to concentrate its learning on the area of interest. This approach maximises the efficiency of data generation, allowing for a higher density of samples within the region where the posterior is expected to peak, and ensures that the flow is optimised for analysing the specific event, leveraging the precise sky localization. However, there are notable disadvantages to this approach. Training can only be performed after an event has been detected and its localization determined, which means the method is not applicable for general, pre-detection training. Additionally, the trained NF can only be used for this particular event, limiting its applicability to other detections. Instead for GW200105_162426 and GW200115_042309, NSIDE = 32 was used, and the entire sky was sampled during data generation. Therefore, two NF were trained, one specifically for GW190814 and one for the other two NSBH events.

The results of the posterior distributions for the three NSBH systems over H_0 are presented in Fig. 4.8. The individual event posterior results for GW200105 and GW200115 show good agreement with `gwcosmo`, exhibiting JS divergences of 0.5 millinats and 0.4 millinats, respectively. However, the result for GW190814 stands out, indicating a clear mismatch between the `gwcosmo` and `CosmoFlow` posterior distributions, both with catalogue and empty catalogue assumptions. Despite extensive investigation, no definitive reason for this discrepancy has been identified. This suggests a need to better understand how the data generation process should produce synthetic GW events from a given galaxy catalogue.

In the `CosmoFlow` result, a distinct catalogue feature is observed at $H_0 = 28\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$, which is absent in the `gwcosmo` result. This discrepancy warrants further investigation to uncover its underlying causes. Interestingly, results from Ref. [151], which used the Dark Energy Survey (DES) [9] catalogue, also show a similar feature for the same event. Their posterior distribution over H_0, depicted in Fig. 4.9, reveals a clear peak at $H_0 = 28\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$ that matches the `CosmoFlow` results, despite differences in other modes. While this observation initially seemed unexpected, given that the `CosmoFlow` analysis used the GLADE+ catalog, it is plausible since low H_0 spikes correspond to low redshift

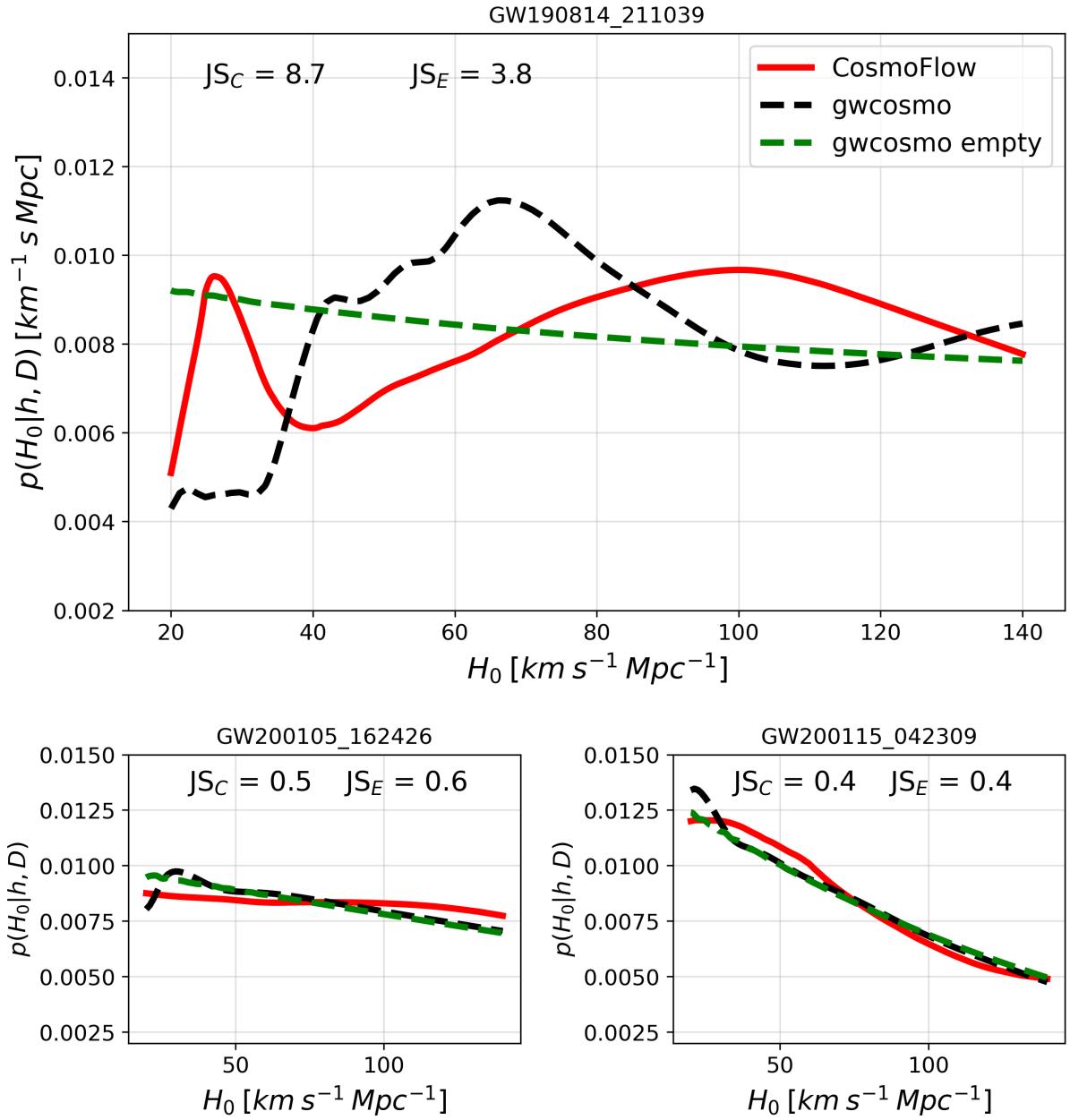

Figure 4.8: Posterior distributions of the Hubble constant, H_0, for three NSBH events. The top panel shows the posterior distribution for the event GW190814, while the bottom two panels show the posterior distributions for the events GW200105 (left) and GW200115 (right). The red solid lines represent the `CosmoFlow` posterior distributions, the black and green dashed lines represent the `gwcosmo` posterior distributions for catalogue and empty catalogue assumptions. The JS values between `CosmoFlow` and `gwcosmo` (JS$_C$ for catalogue distribution, JS$_E$ for empty catalogue distribution, both in units of millinats) are indicated in each panel.

galaxies, which are likely shared between both catalogues. Therefore, it is not surprising that such a feature could align in the posteriors obtained using the DES and GLADE+ catalogues. This intriguing result is noted for its interest, and further work will be directed at understanding the mismatch between our primary analysis and the `gwcosmo` results.

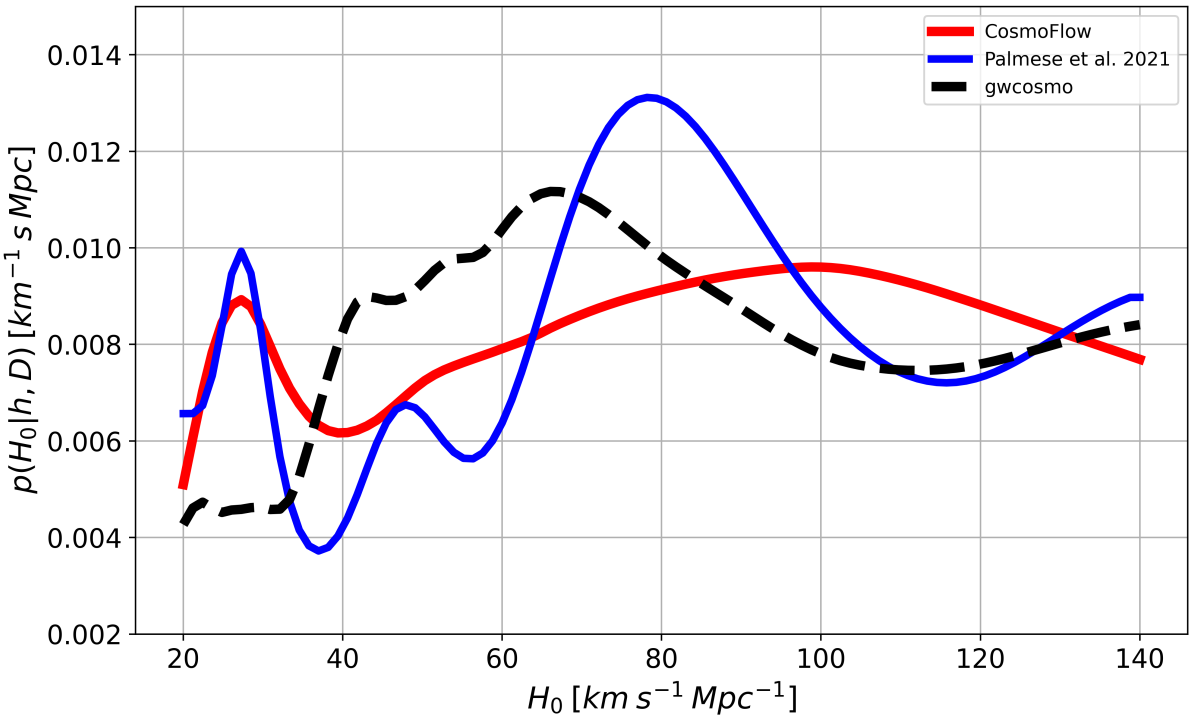

Figure 4.9: Plot showing three posterior distributions of H_0: the `CosmoFlow` result in red, the `gwcosmo` result in black dashed lines (both from Fig. 4.8), and the result from [151] in solid blue.

4.3.4 NSBH + BBH results

We are now able to combine all 45 GW events (BBHs and NSBHs) to compute the final combined posterior distribution over H_0. The final result is shown in Fig. 4.10. Clearly, there is a mismatch between the three posteriors, but there is evidence that this discrepancy is not due to `CosmoFlow` failing to capture catalogue features, such as modes in the lower value regions of H_0. If it had, there should have been a better match with the empty catalogue result of `gwcosmo`. Evidently, the mismatch could be more related to selection effects implemented within the data generation code, both from the EM and GW sides. This would explain the discrepancy with event GW190706_222641 in Fig.4.6, where this event is solely driven by the priors, as it is too distant to have any catalogue information present. What we noticed with this particular event is the limited data in regions consistent with the event posteriors, specifically in the dimensions of luminosity distance and detector-frame masses. When generating the training data, we conditioned it only on detectable events with a SNR of $\rho \geq 11$ to account for selection effects. This conditioning leads to a highly correlated prior distribution that is conditional on detection, which is not an analytic function and differs from the basic prior. As a result, the training data distribution is well-defined but constrained, causing fewer samples in specific parameter space areas where some GW event posteriors overlap, such as GW190706_222641. Con-

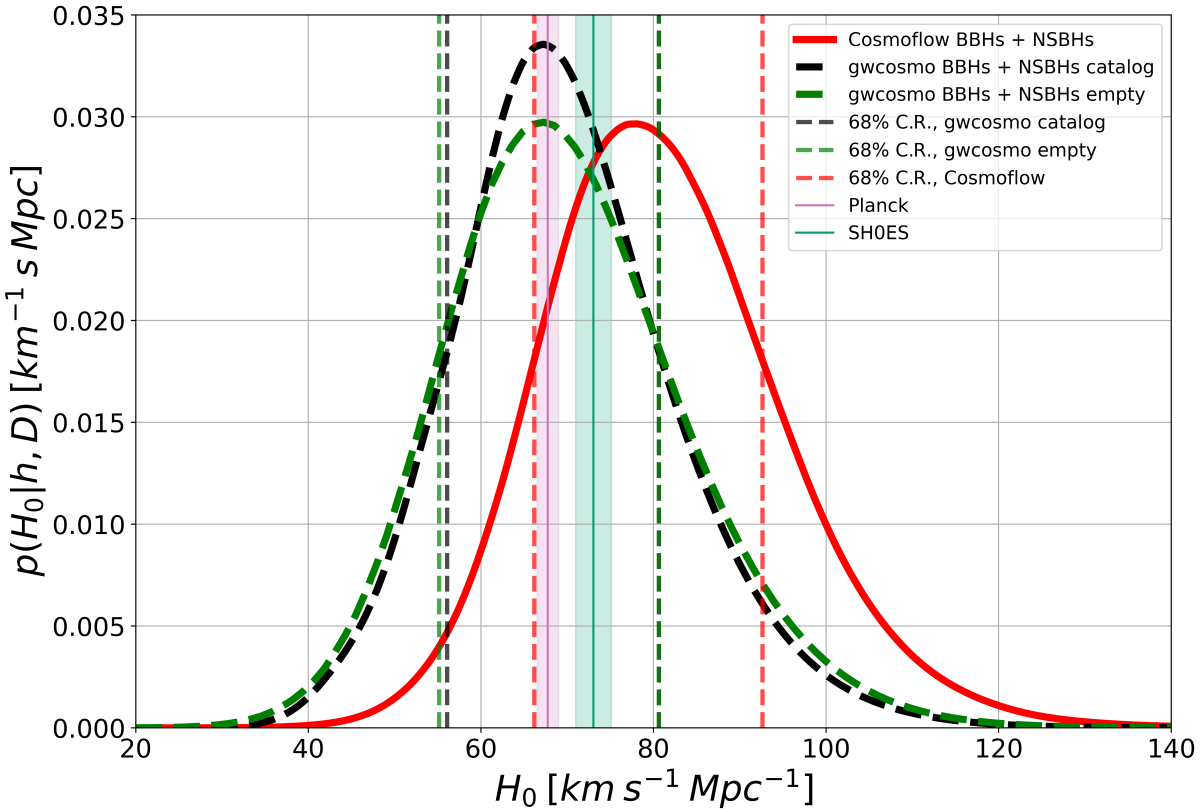

Figure 4.10: Posterior distributions of the Hubble constant H_0 derived from different datasets. The solid red curve represents the posterior from the `CosmoFlow` BBH + NSBH dataset. The dashed black line shows the posterior from the `gwcosmo` BBHs and NSBHs with galaxy catalogue psoterior distribution, the green dashed line shows the result for an empty catalogue approach for `gwcosmo`. The 68% minimal confidence regions are also shown for each respective distribution. For context, the Planck and SH0ES measurements of H_0 are shown with their respective vertical lines. The shaded regions around these lines represent their 2σ confidence intervals, providing a visual comparison of how the GW based measurements align with these established cosmological results.

sequently, the NF may not have learned sufficient detail in those areas. It is important to note that some differences between the results may be due to variations in the prior assumptions used by the two methods. Despite our efforts to keep these assumptions consistent, there can still be factors that influence the outcomes. One example is the $z_{\max}$ cutoff applied in Chapter 2 to speed up data generation, which may have contributed to the observed discrepancies.

In conclusion, by refining the SNR MLP approximator and re-evaluating our prior assumptions, we hope to resolve these issues and achieve a more robust and accurate estimation of H_0. Future work will focus on these improvements, aiming for better alignment between `CosmoFlow` and `gwcosmo` and advancing our understanding and analysis of cosmology with GW events.

4.4 Speed Up with Batching

The results of the posterior distributions over H_0 for each GW event shown in Fig. 4.6 were generated by evaluating the NF model for each value of H_0 sequentially, with a fixed set of GW posterior samples, N_{θ}. While this method effectively computes the posterior distribution, it requires looping over the cosmological parameters for each event, and then combining the resulting posterior distributions from each event to obtain the overall combine posterior distribution. However, this sequential approach does not take advantage of batching techniques that could significantly speed up the process.

The posterior distribution of the Hubble constant H_0, given the data, can be expressed as:

$$p(H_0|\mathbf{h},\mathbf{D},I) \propto p(H_0|I)\prod_i \left\langle \frac{p(\theta_i|D_i,H_0,I)}{p(D_i|\theta_i,I)p(\theta_i|\Omega_0,I)} \right\rangle_{\theta_i\sim p(\theta_i|h_i,I)}. \tag{4.5}$$

In this equation (this is the same as in Eq. (2.12)), the numerator inside the brackets is computed using the NF model and averaged over the posterior samples θ_i. Evaluating individual H_0 values sequentially for each event is inefficient, especially with a large number of events and samples, as batching is applied only over the GW parameters and not over H_0. To address this, we can batch all inputs: the GW posterior samples (N_{θ}), the cosmological parameter values (N_{samples}), and the number of events (N_{events}). By structuring these inputs into a single array with dimensions $[N_{\text{events}} \times N_{\theta} \times N_{\text{samples}}]$, we can evaluate Eq. (4.5) in parallel across all events, rather than sequentially. After the NF model computes the probabilities, the log probabilities of each event's posterior are summed to produce the final result.

We tested the batching method using different sets of N_{θ} posterior samples from 21 events labeled with O3 and H1 L1 V1, using a single flow (Flow 4 from Table 4.1). The left plot in Fig. 4.11 shows the clear advantage of batching over the sequential approach. The computation time for the minimal batching method remains relatively constant and slow, while the full batching method scales efficiently, keeping computation time below 10 seconds. The key advantage of batching is that it fully uses the available memory of the GPU, maximising the efficiency of computing Eq. (4.5). With minimal batching, each event's N_{θ} parameters are processed sequentially over all H_0 values, using minimal GPU memory, resulting in computation times around 100 seconds, as shown by the red points in Fig. 4.11. In contrast, when inputs are batched, the computation time ranges from 0.2 seconds for $N_{\theta} = 10$ to a maximum of 10 seconds for $N_{\theta} = 2000$. Using an RTX 2080 GPU with 8 GB of memory, we observed that as the memory usage approaches its limit, data must be processed in multiple smaller batches, leading to a linear increase in

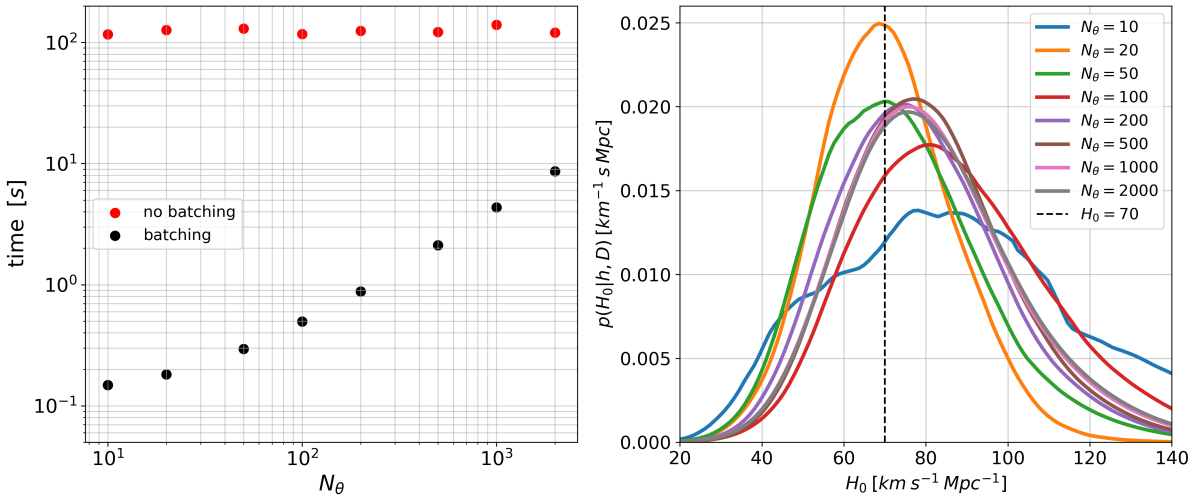

Figure 4.11: (Left) Computation times as a function of the number of posterior samples N_θ. Red points represent computation times minimal batching (sequential over H_0), while black points represent times using full batching. (Right) Posterior distributions for different values of N_θ. The vertical dashed line represents $H_0 = 70$ km s^{-1} Mpc^{-1}. The legend indicates the number of posterior samples used.

computation time. Interestingly, this linear trend appears at lower memory consumption levels than expected (i.e. lower values of N_θ), suggesting further investigation is needed to optimise computational speeds. Future work will address these issues to fully exploit GPUs capabilities, enhancing the efficiency of this analysis.

The right plot of Fig. 4.11 shows the impact of not using enough GW posterior samples, with values below $N_\theta = 500$ leading to noisy results. Based on our observations, we set a threshold of $N_\theta = 1000$ as the minimum number of posterior samples to ensure stable results, as little variation was observed between 500, 1000, and 2000 samples. For a minimum of $N_\theta = 1000$ posterior samples per event, the evaluation time was approximately 4.4 seconds when combining 21 events, with 1000 GW posterior samples evaluated over 100 H_0 values. This corresponds to about 2 milliseconds for one evaluation of H_0 for a single event. This computation time scales linearly, suggesting that at a scale of 1000 events, each using 1000 GW posterior samples, the final combined posterior distribution could be evaluated in approximately $2\text{ms} \times N_{\text{events}} \times N_{\text{samples}} \approx 200$ seconds, which is ≈ 3.3 minutes. However, a significant challenge with this method arises from the memory usage of the NF model when evaluated using a single GPU. Batching a large number of events and cosmological parameter samples can lead to substantial memory consumption, potentially filling the available GPU memory. Therefore, it is important to consider the memory limitations of the GPU when evaluating the NF model to avoid exceeding its capacity. As the number of events or posterior samples increases, memory usage also rises, potentially leading to memory overflow if not properly managed. This would negate

the advantages of the batching method, as full memory consumption would require the batched data to be processed iteratively, looping over smaller subsets to handle the entire large dataset. Balancing the computational efficiency gained through batching with the memory constraints of the GPU is essential for optimising the evaluation process.

Overall, this analysis shows the batching method not only provides a significant speedup but also handles larger datasets more effectively, making it a more suitable choice for computationally intensive tasks such as cosmological PE. Future work will focus on further optimising this approach by enabling the use of multiple GPUs in parallel. This will allow different NF models to be evaluated simultaneously across various GPUs, thereby distributing the computational load and avoiding the risk of overflowing the memory capacity of any single GPU.

4.5 Conclusion

In this chapter, the results of the `CosmoFlow` pipeline have been presented and thoroughly compared against the findings from [16], which used the `gwcosmo` method.

We began by describing the GW data used in the analysis, specifically focusing on the posterior samples from the selected events, which include parameters such as luminosity distance, primary and secondary masses, and sky location. The events selected for this analysis are listed in Table 4.1. These events were chosen based on their matched-filter SNR values across the detector network, specifically selecting those with a detector network matched-filtering SNR above a certain threshold, $\rho_{\text{th}} = 11$, as outlined in [16]. Consequently, only 42 BBH events from the GWTC catalogues [99] met this criterion. For each event, the active detectors and corresponding observational run were identified, prompting us to generate a dataset and train the model for each combination of detector setup and observing run, resulting in seven distinct NF models. With these seven conditional NFs trained, each corresponding to a specific observing run and detector setup, we computed the posterior distribution for each event over H_0. The individual posterior distribution results are presented in Fig. 4.6, where the results from [16], obtained using `gwcosmo`, are also plotted for comparison. The comparison shows that in many cases, the agreement between `CosmoFlow` and `gwcosmo` was strong, with a minimum JS divergence of 0.03 millinats. However, some cases, such as GW190706_22264, exhibited less agreement, with a JS divergence of 26.27 millinats. This discrepancy may be attributed to several factors, including undersampling of the parameter space in our generated datasets, which

could lead to underperformance of the NF. Additionally, potential inaccuracies in accounting for GW selection effects or the SNR approximation using the MLP could also contribute to the mismatch. Future work will investigate these factors more thoroughly. After obtaining the posterior distributions over H_0 for each event, we combined the results to derive an overall estimate of $H_0 = 76.51^{+15.15}_{-11.54}\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$ using the `CosmoFlow` analysis, which included only BBH GW events.

We extended the analysis to include NSBHs by adjusting the priors in the data generation process to reflect the physical prior for neutron star masses in the detector frame. Specifically, a uniform prior between $[1,3]\,M_\odot$ was applied, in line with the `gwcosmo` analysis from [16]. A particularly notable case in this extended analysis was GW190814, where the GW parameter space was highly constrained by the posterior distribution, especially in terms of sky location. The posterior distribution for this event indicated a well-defined sky location, covering at most 18 deg^2. To enhance the efficiency of sampling data points within this tightly constrained sky region, a targeted data generation strategy was employed specifically for GW190814. Despite this higher density of training data, a noticeable mismatch between `CosmoFlow` and `gwcosmo` was observed, with a JS divergence of 8.7 millinats. Finally, we explored how the inference process can be accelerated by batching the posterior sample data from the GWTC catalogues to vectorize the calculation of Eq. (4.5). By batching the GW posterior samples with the cosmological parameter values, we achieved a significant speedup, reducing the computation time to approximately 4 ms per 1000 GW posterior samples for one cosmological parameter value and one GW event. This optimisation highlights the potential for further enhancing the efficiency of the `CosmoFlow` pipeline, especially for large-scale analyses.

Chapter 5

Joint cosmological and population parameter estimation

5.1 Introduction

In Chapter 4, we presented the one-dimensional PE for the Hubble constant (H_0) using GW posterior samples from the third GWTC [99]. During this analysis, we held the population and merger-rate parameters constant primarily to perform a direct comparison with results from `gwcosmo` [16].

However, `CosmoFlow` is also capable of inferring and marginalising over uncertainties for any parameter associated with the distribution models used during the data-generation process in Chapter 2. Given this capability, we now aim to expand our analysis to include not only H_0 but also various population parameters. These include parameters from the Madau and Dickinson model for the merger-rate, as described in Chapter 2, Sec. 2.4.1, such as the power law index at low redshifts (γ), the power law index at high redshifts (k), and the peak redshift (z_p), as well as those related to the black hole mass PLP distribution, described in Chapter 2 Sec. 2.5.1, including the power-law slopes (α, β), the maximum mass ($M_{\max}$), the minimum mass ($M_{\min}$), the mean of the Gaussian component (μ_g), the standard deviation of the Gaussian component (σ_g), the fraction of black holes in the Gaussian component (λ), and the smoothing factor parameter (δ_m). The reason for specifically varying these parameters at this stage is to directly compare our results with the latest outcomes from `gwcosmo` [52], which is also capable of performing PE on these parameters.

In this chapter, we will first discuss the changes made to the data-generation process to account for the variation of these additional parameters, focusing specifically on the distributions $p(z)$ and $p(m_1,m_2)$, since these are the ones being varied over the hyperparameters. Then, we will explain how we train the NF models with 14 target inputs and 12 conditional inputs for each flow, one for each training data set as described in Tab. 4.1. Finally, we will compute the joint posterior distribution using NS, specifically employing `nessai` [38], which is a machine-learning driven approach to performing NS.

5.2 Data-generation with multiple parameter variation

The data-generation process for multiple parameters retains the same structure as the one-dimensional case shown in Fig. 2.3. The key differences involve the new hyperparameters of the distributions, specifically $p(z)$ described in Eq. (2.29), and the $p(m_1)$ and $p(m_2|m_1)$ distributions described in Eq. (2.55) and Eq. (2.58), respectively. When generating synthetic GW events, we ensure that each hyperparameter is reused if it does not yield a detectable event. This approach ensures that the chosen hyperparameter priors are under our control and conform to the distributions we dictate, leading to a known coverage of the parameter space independent of detectability, which in this case, we choose a uniform distribution. As discussed in Chapter 2, this method maintains a consistent prior coverage and avoiding biases that could arise from only using hyperparameters that produce detectable events. While it is technically acceptable to avoid reusing these hyperparameters, since the cosmological priors are on the right-hand side of the NF probability function described in the numerator of Eq. (4.5), doing so could bias the distribution towards hyperparameter values yielding more detectable events. This bias could, in turn, negatively impact the performance of the NF in regions with lower detectability.

the table in Tab. 5.1 lists the parameters varied during the multi-parameter data-generation process, along with their descriptions and priors. The prior values are based on the latest analysis performed with `gwcosmo` [52], which in turn relies on the LVK cosmology analysis [16], where wide uniform priors were used for both the cosmological and BBH population parameters. Also, a constant prior for the local merger-rate, $p(R_0) \propto 1/R_0$, was used in the `gwcosmo` analysis (see Chapter 1 Sec. 1.6.2). Although we haven't explicitly discussed merger-rate priors in previous sections, it is important to note that by not including this prior explicitly in our analysis, we are effectively assuming the same prior as used by `gwcosmo`. We also provide an example of how the PLP model for $p(m_1)$ and $p(m_2)$ is affected by variations in the population parameters, as illustrated in Fig. 5.1. Each subplot

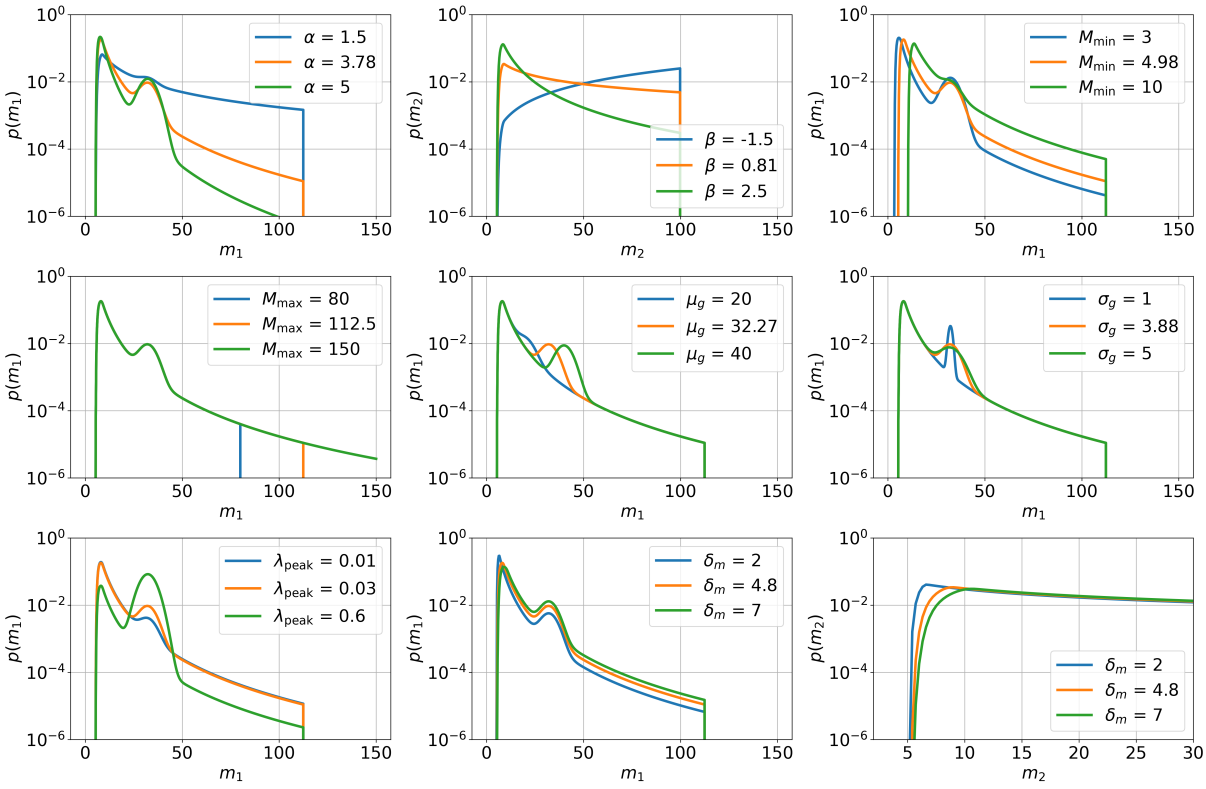

Figure 5.1: Impact of varying population parameters on the PLP model. Each subplot shows the effect of varying a single parameter on the probability density functions of the primary mass (m_1) and secondary mass (m_2) distributions for BBH mergers. The parameters varied include: (Top Row) α (power-law slope, fixed at 3.78), β (mass ratio slope, fixed at 0.81), $M_{\rm min}$ (minimum mass, fixed at 4.98 $M_\odot$); (Middle Row) $M_{\rm max}$ (maximum mass, fixed at 112.5 $M_\odot$), μ_g (mean of the Gaussian peak, fixed at 32.27 $M_\odot$), σ_g (width of the Gaussian peak, fixed at 3.88 $M_\odot$); (Bottom Row) $\lambda_{\rm peak}$ (λ_g) (fraction of the Gaussian peak, fixed at 0.03), and δ_m (smoothing parameter at the lower mass boundary, fixed at 4.8 $M_\odot$). The labels within each plot indicate the specific values of the varied parameter, demonstrating how changes in these parameters influence the overall mass distributions $p(m_1)$ and $p(m_2)$. For $p(m_2)$, $m_1 = 100\,M_\odot$.

shows different instances of the distributions with all parameters held constant except for one, highlighting its specific influence. It is evident that varying the population parameters simultaneously results in complex distributions, particularly when each GW event is subject to detection thresholds. Certain parameters, such as μ_g and $M_{\rm max}$, have a more pronounced effect on detectability.

After generating the synthetic data with different sets of cosmological and population parameters for each sample, we can then explore the data to check for any inconsistencies or potential biases introduced by the choice of priors. We begin by examining the component masses, m_1 and m_2 for the detector-frame. The sampling of each individual mass component was achieved by varying the population parameters within their respective priors, as outlined in Tab. 5.1. In Fig. 5.2, plot shows the distributions of the detector-

Table 5.1: Summary of hyper-parameter distributions with the corresponding priors [52].

Cosmology Parameters		
Parameter	Description	Prior
H_0	Hubble constant	$U(20.0, 180)\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$
PLP, $p(m_1, m_2)$		
Parameter	Description	Prior
$M_{\mathrm{BH\ min}}$	Minimum mass of the PL component of the black hole mass distribution	$U(2.0M_\odot, 10.0M_\odot)$
$M_{\mathrm{BH\ max}}$	Maximum mass of the PL component of the black hole mass distribution	$U(50.0M_\odot, 200.0M_\odot)$
α	Spectral index for the PL of the primary mass distribution	$U(1.5, 12.0)$
μ_g	Mean of the Gaussian component in the primary mass distribution	$U(20.0M_\odot, 50.0M_\odot)$
σ_g	Width of the Gaussian component in the primary mass distribution	$U(0.4M_\odot, 10.0M_\odot)$
λ_g	Probability of sampling from Gaussian component	$U(0.0, 1.0)$
δ_m	Range of mass tapering on the lower end of the mass distribution	$U(0.0M_\odot, 10.0M_\odot)$
β	Spectral index for the PL of the secondary mass distribution	$U(-4.0, 12.0)$
merger-rate Shape Parameters, $p(z)$		
Parameter	Description	Prior
γ	Power-law index describing the merger-rate at low redshift	$U(0, 12.0)$
κ	Power-law index describing the merger-rate at high redshift	$U(0, 6.0)$
z_p	The redshift where the slope of the merger-rate changes	$U(0, 4.0)$

frame masses for m_1 and m_2, conditioned on detectability. Figure 5.2 shows that most of the points in the distribution tend to cluster near the $m_1 = m_2$ line or within the broader Gaussian peak. The clustering along the $m_1 = m_2$ line occurs because scenarios where $m_1 \approx m_2$ generally yield higher matched-filter SNR, making these events more detectable. On the other hand, the broad and densely sampled Gaussian region is a result of marginalising over the parameters λ_g, σ_g, and μ_g, which describe the Gaussian component of the mass distribution model. Marginalising over these parameters broadens the Gaussian component and allows for a wide range of events to be sampled from this distribution. Specifically, λ_g, which ranges from 0 to 1, controls the relative contribution of the Gaussian component within the overall PLP distribution. Higher values of λ_g increase the chances of sampling masses from this Gaussian component, while lower values reduce its

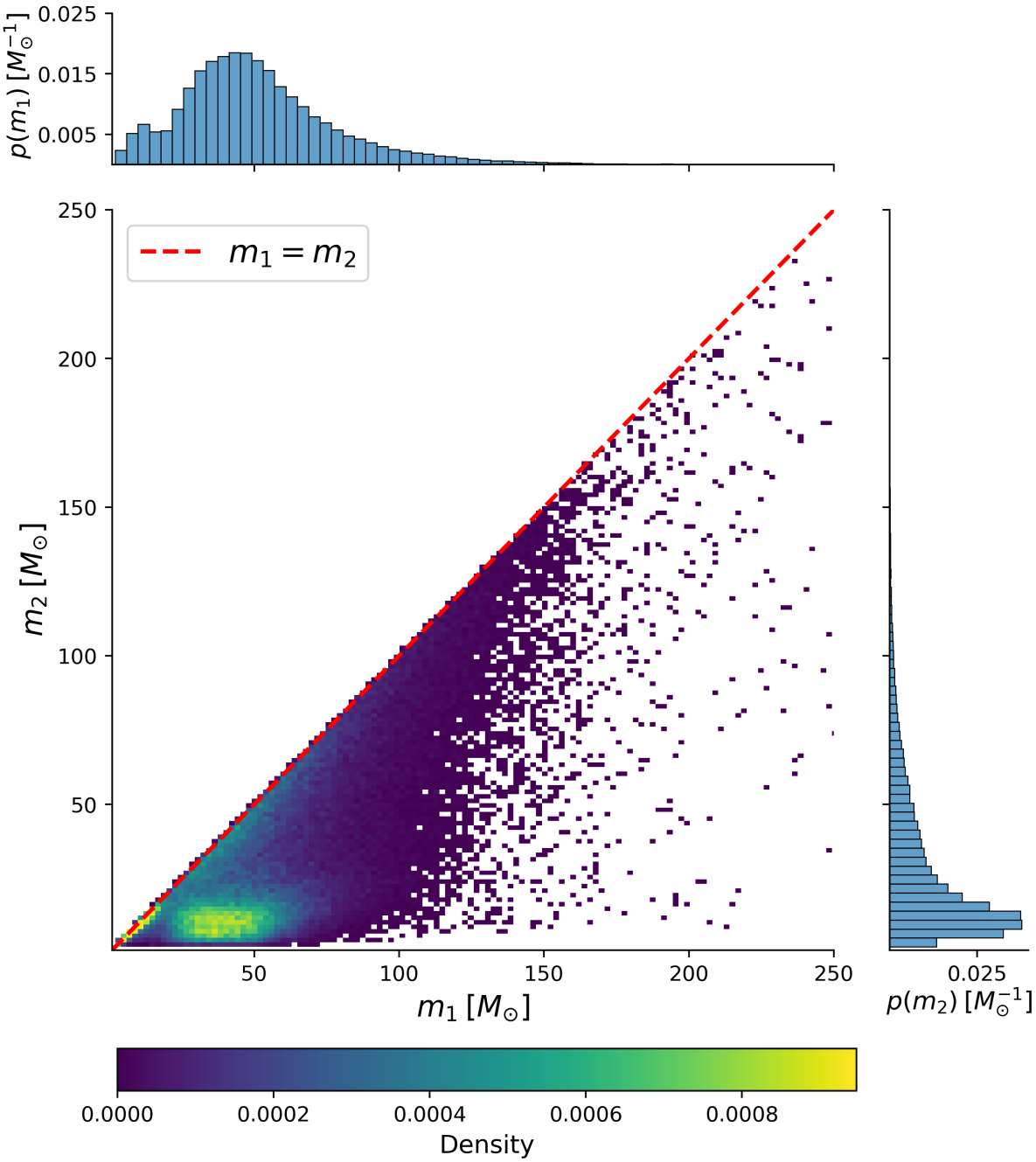

Figure 5.2: Distribution of primary and secondary black hole masses in the detector-frame, highlighting the relationship between m_1 and m_2 conditioned on detectability. The main panel presents a 2D histogram of m_1 versus m_2, with a color scale representing the density of events. For clarity, both axes are truncated at $250 : M_\odot$. The red dashed line indicates where $m_1 = m_2$, serving as a reference to identify mass pairs with equal components. The marginal distributions along the top and right axes depict the one-dimensional distributions of m_1 and m_2, respectively. The density map was obtained by computing a 2D KDE.

influence. The top and right panels show the marginalised distributions for m_1 and m_2, respectively. The marginal distribution of m_1 exhibits a pronounced peak, reflecting the

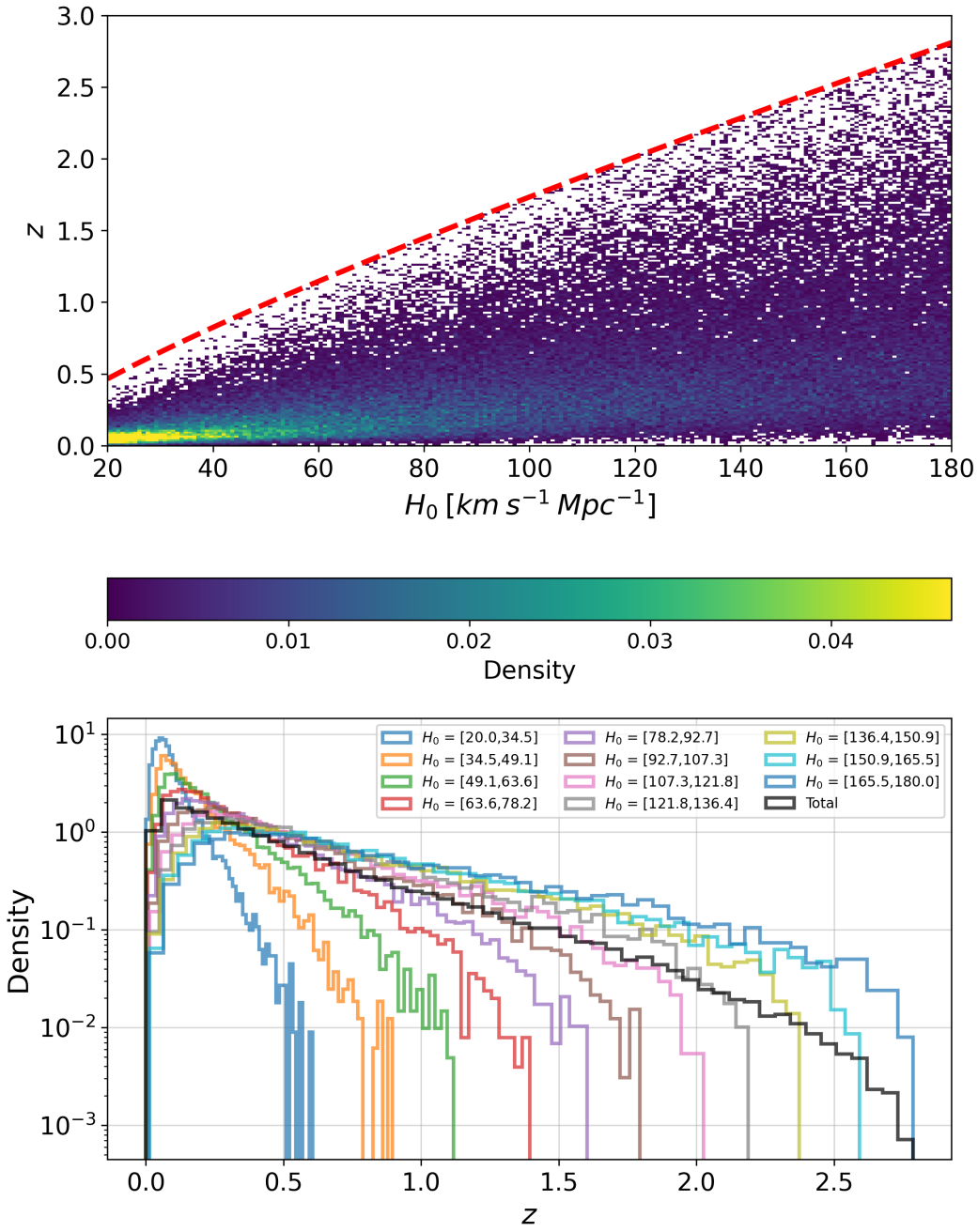

Figure 5.3: Top panel: 2D density distribution (obtained by computing a 2D KDE) of H_0 versus z with a density color scale. The red dashed line represents the maximum redshift $z_{\max}(H_0|\rho_{\rm th} = 11)$ based on the detectability threshold. Bottom panel: the marginal distribution of z is shown for different H_0 bins, with the total marginalised distribution in black.

combined effects of detectability and the contribution from the Gaussian component of the mass model. This peak is more pronounced due to the fact that higher primary masses are more likely to be detected, especially when the Gaussian component is emphasized by higher values of λ_g.

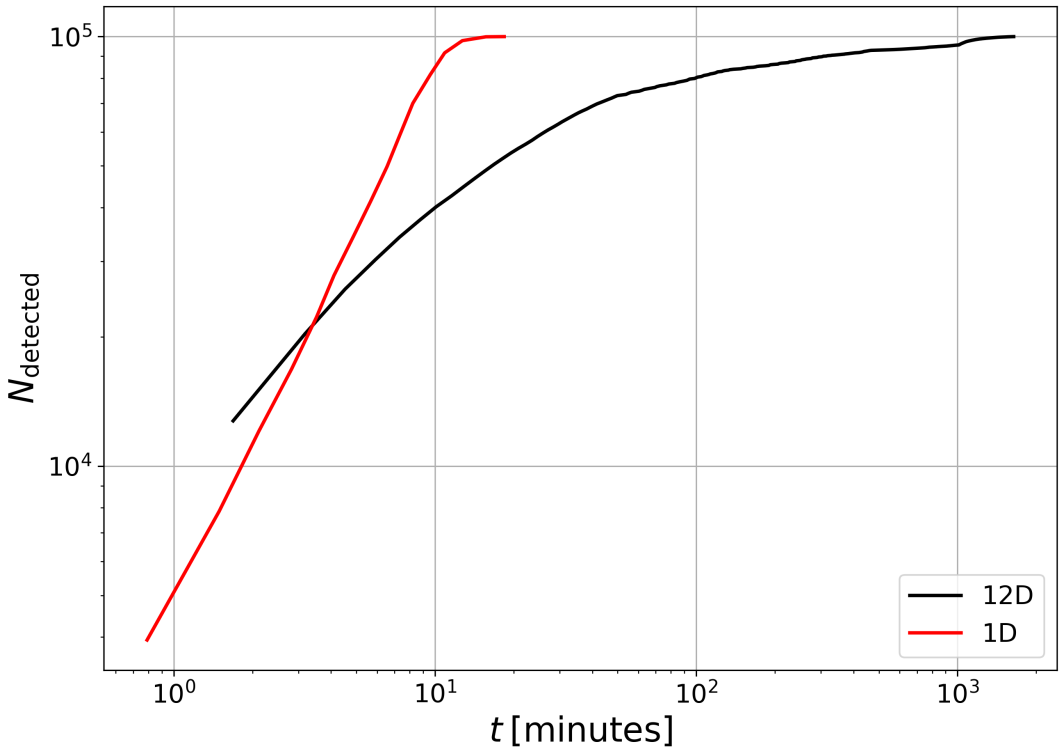

Figure 5.4: The cumulative number of detected events as a function of elapsed time for the multi-dimensional data-generation (12D, in black) and one-dimensional data-generation (1D, in red) process for 100,000 detected synthetic GW events.

In addition to the mass distribution, we also examine the relationship between redshift and the Hubble constant. In Fig. 5.3, the top panel displays a 2D density distribution of the redshift (z) versus H_0 from the generated training data. The density distribution shows that higher values of H_0 correspond to higher redshift values, with the density decreasing as both H_0 and z increase. The redshift samples were thresholded by the maximum redshift values obtained from Sec. 2.6.2, where a redshift GW horizon was set for all detectable events above a detector network matched-filter SNR above 11, as a function of H_0. However, for this case, the z_{max} (see Sec. 2.6.2) value was increased by 20% to address the observed cutoff in the redshifts at higher H_0 values. This adjustment was made to avoid artificial cutoffs in the generated synthetic GW events across all H_0 values. The goal is to allow the detectability criterion to naturally suppress the distribution at high redshifts, without artificially excluding any potentially detectable events. Using the same value as in Chapter 2 resulted in some cutoff in the data, prompting the increase of z_{max} by 20%. The cutoff was due to marginalisation over the mass hyperparameters, specifically λ_g and M_{max}. The first parameter, as observed in Fig. 5.2, allows for m_1 to be sampled from the Gaussian component, increasing the chance of sampling an event at higher distances due to the higher mass. The second parameter, M_{max}, sets the upper limit on the maximum mass that can be sampled from the joint distribution $p(m_1, m_2)$. This

adjustment allows for sampling higher masses, similar to the purpose of the λ_g parameter. To confirm the validity of the cutoff, the second plot in Fig. 5.3 presents the marginalised density distribution of z across different H_0 bin intervals. The distribution tails within each H_0 segment approach negligible values, indicating that the imposed cutoff at $z_{\max}$ is reasonable and does not exclude significant portions of the observable population.

Unfortunately, the data-generation process for the multi-parameter case is significantly more time-consuming than for the single-parameter (1D) case, taking approximately 75 times longer to complete. This increased time is due to the variation of all the hyperparameters, which often results in the generation of events that lie in very undetectable regions of the parameter space. Consequently, more time is required to find configurations of the sampled $\boldsymbol{\theta}_{\mathrm{GW}}$ parameters that yield detectable events. In Fig. 5.4, the number of detected events in the multi-hyperparameter data-generation process is plotted against the cumulative time taken. It is evident how slow the process is compared to the 1D case, even after applying various efficiency methods to the data-generation (e.g., SNR approximator in Sec. 2.6.1, maximum redshift distribution in Sec2.6.2, simultaneous sampling from CDFs Sec. 2.5.2). The process of detecting 100,000 events took approximately 1,500 minutes ($\approx$ 25 hours) for the 12D case, instead for the 1D it took approximately 18 minutes. This extended duration is primarily due to the data-generation process, which attempts to use fixed samples of hyperparameters that are situated in regions of the parameter space where GW events are very difficult to detect. This inefficiency becomes especially evident after the first 75% of the events are detected. The initial 75,000 events were detected in about 1 hour, but the remaining 25,000 events took approximately 24 hours. The reason for this significant slowdown is that the process continues to reuse fixed samples of hyperparameters from these hard-to-detect regions, which makes the detection process increasingly time-consuming. If we were to discard these fixed samples of hyperparameters instead of attempting to make them detectable, the process could be considerably faster. However, this approach carries the untested risk of generating hyperparameter distributions that are conditional on detection, deviating from the initial uniform prior distribution that was originally set. This deviation could introduce bias into the results. The trade-off between speeding up the process and maintaining the integrity of the original, unbiased parameter distribution should be carefully weighed before making any changes to the current approach.

5.2.1 Threshold on minimum detectable Chirp Mass

New methods to enhance efficiency are being investigated, one of which involves introducing a threshold in the luminosity distance and detector-frame chirp mass parameter space. We begin by defining the GW strain $h(t)$ for a binary inspiral in the leading-order approximation, commonly referred to as the Newtonian limit or quadrupole approximation [13], as:

$$h(t) \propto \frac{\mathscr{M}_z^{5/3}}{D_{\rm L}} \left(G\pi f(t)\right)^{2/3} \cos\left(2\pi f(t)t + \phi_0\right), \tag{5.1}$$

where $\mathscr{M}_z = \mathscr{M}(1+z)$ is the redshifted chirp mass, $D_{\rm L}$ is the luminosity distance to the source, $f(t)$ is the GW frequency, and ϕ_0 is the phase of the wave at time $t = 0$. By substituting Eq.(5.1) into the matched-filter SNR equation, described in Eq.(1.28), follows:

$$\rho \propto \frac{\mathscr{M}_z^{5/6}}{D_{\rm L}}. \tag{5.2}$$

The variation in luminosity distance imposes a threshold on the maximum redshifted chirp mass that detectors can observe, assuming a fixed SNR threshold, $\rho_{\rm th}$. Therefore, to focus on detectable events, a bootstrap analysis can be performed to identify the minimum detector-frame chirp mass associated with the maximum observable distance for a given $\rho_{\rm th}$.

One way to find the function $\mathscr{M}_{z,\rm min}(\rho_{\rm th}, D_L)$ is to perform a bootstrap analysis, similar to the one described in Sec. 2.6.2, where we found the maximum redshift at which events can be detected by marginalising over all other parameters and selecting the maximum distances times SNR. The matched-filtering SNR is influenced by factors beyond just detector-frame chirp mass and luminosity distance, making it difficult to identify a clear minimum. To account for complex dependencies and better understand detectability limits, a statistical approach like bootstrap analysis was used and applied to the $\mathscr{M}z$ and DL spaces. In this case, we sampled m_1 and m_2 uniformly in the detector-frame within the range $U[2, 350]\, M_\odot$, while fixing the distance at 100 specific steps between 10 Mpc and 12,000 Mpc. For each step, we sampled 200,000 sets of GW parameters (enough to find an SNR above a threshold of 11). We evaluated the minimum $\mathscr{M}_z$ for all the $\rho > \rho_{\rm th}$, at each luminosity distance step. From this analysis, we can perform a best-fit evaluation. The results are illustrated in Fig. 5.5, where the best fit is shown for the data set generated during an O3-type run with only HV detectors operational. This particular data set was chosen because it is the most time-consuming to generate, as the absence of the L detectors significantly lowers the probability of sampling detectable GW events. The figure presents a 2D density map of the parameter space defined by $\mathscr{M}_z$ and $D_{\rm L}$, conditioned on detectability. The best-fit function for the minimum chirp mass is overlaid on this map

and is expressed mathematically as:

$$\mathscr{M}_{z,\min} = (aD_L)^{6/5}, \tag{5.3}$$

where $a = (0.0055 \pm 0.0002)\, M_{\odot}^{5/6}\,\mathrm{Mpc}^{-1}$. The choice to keep the proportionality constant within the power of $6/5$ is to preserve the functional form of Eq.(5.2). We also note that the value of the fit parameter a in Eq. (5.3) depends on the threshold $\rho_{\mathrm{th}} = 11$. For a different SNR threshold, the fit value would change accordingly. Also, the presence of some events below the minimum redshifted chirp mass is due to the way the data-generation process handles SNRs. Initially, we generate optimal SNRs using the MLP approximator. The optimal SNR represents the maximum possible SNR for a given set of parameters, assuming perfect alignment and no noise interference. However, in reality, GW detectors measure the matched-filter SNR, which includes the effects of noise. To simulate this, we add noise to the optimal SNR values during data-generation. This noise can cause some events, which originally had an optimal SNR below the detection threshold, to cross the threshold due to favorable noise fluctuations. As a result, these events are detected even though their corresponding redshifted chirp masses fall below the fit predicted by the optimal SNR alone. In future work, we plan to refine the best-fit analysis for $\mathscr{M}_{z,\min}$ by explicitly accounting for the impact of noisy fluctuations on the matched-filter SNR and implement this within the analysis.

In Fig. 5.5, A clear region in the bottom right is not being sampled because, at increasing distances, a minimum detector-frame chirp mass is required to obtain a matched-filter SNR above ρ_{th}. These regions of the parameter space yield an SNR lower than the SNR threshold, therefore can be limited by sampling redshifted chirp masses above a certain value, $\mathscr{M}_{z,\min}(D_L)$, as plotted in Fig. 5.5. This minimum redshifted chirp mass would act as a minimum mass for simulating GW events from the data-generation code. Currently, we do not yet have a definitive method for implementing this threshold, but future work will focus on developing such an approach. Incorporating this threshold could significantly accelerate the data-generation process by reducing the parameter space that needs to be explored. Preliminary estimates, based on overlaying the minimum redshifted chirp mass function onto the prior distribution, suggest that the lower boundary of the prior space's maximum cutoff could exclude approximately 94% of the parameter space. This reduction implies that the speed of the data-generation process could potentially be increased by a factor of 10.

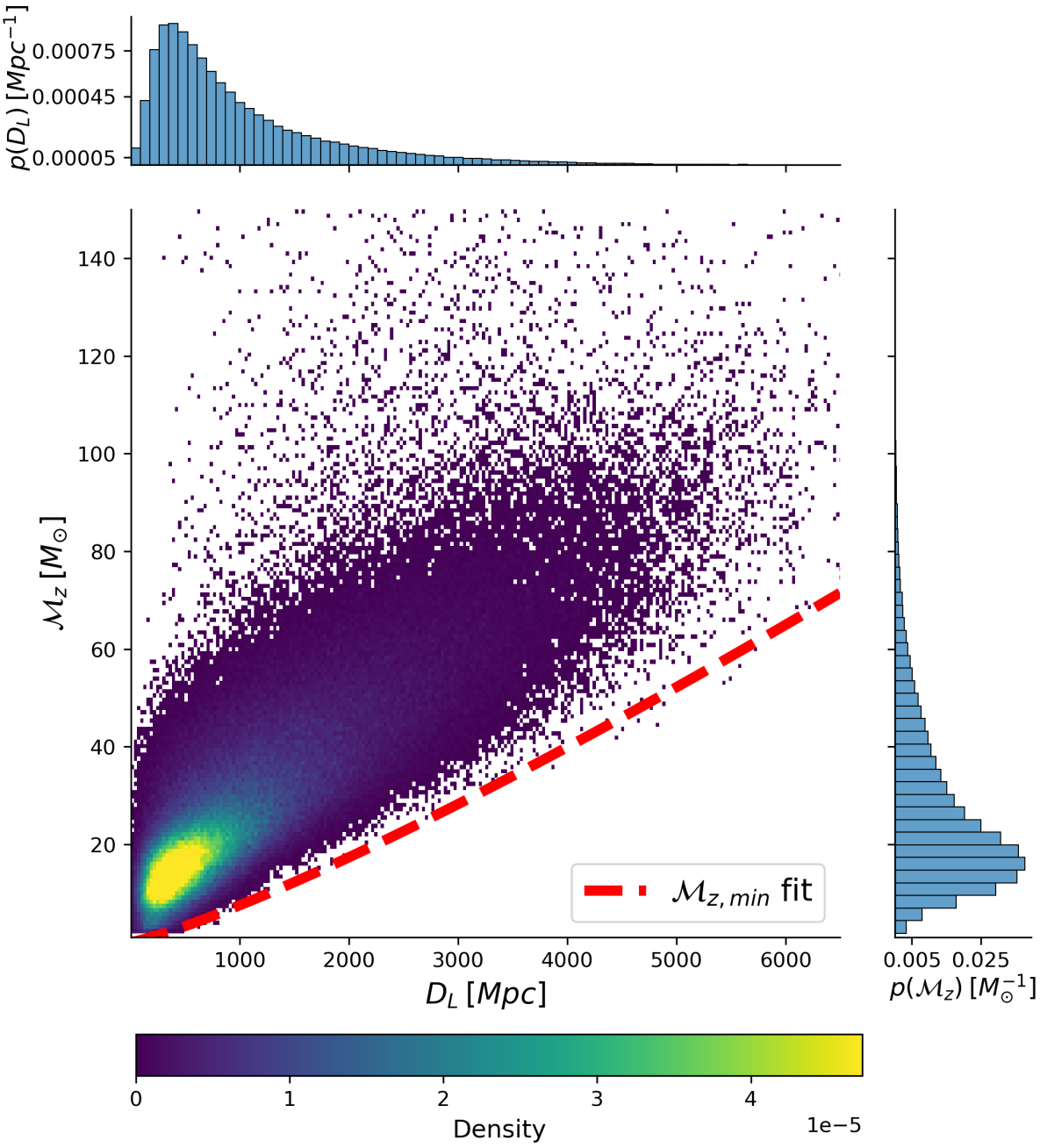

Figure 5.5: 2D density map of the detector-frame chirp mass $\mathscr{M}_{z,\mathrm{detector}}$ against the luminosity distance D_L for O3 synthetic data set with HV detectors operational. The red dashed line indicates the minimum detector-frame chirp mass described in Eq.(5.2), above which detectable events have 1/200,000 chance of being detectable, assuming a fixed SNR threshold $\rho_\mathrm{th} = 11$. For visual purposes, the y axis was limited to $150\,M_\odot$ and the x axis at 6500 Mpc. The density map was obtained by computing a 2D KDE.

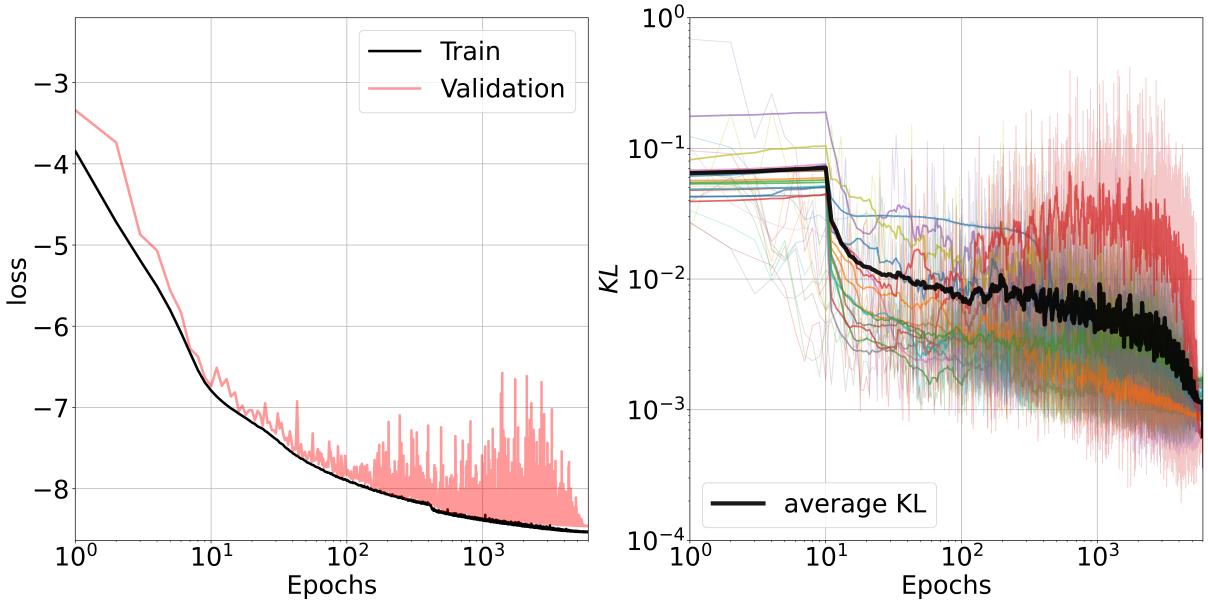

Figure 5.6: (Left): The training and validation losses are plotted across the range of epochs. (Right): The KL divergence values are shown for the 14D latent space, comparing the marginalised latent distributions with a unit Gaussian $\mathcal{N}(\mu = 0, \sigma = 1)$. A running mean filter over 20 epochs is applied to smooth the results, highlighting trends more clearly. The average KL divergence is depicted by the black line, providing a summary of the overall performance.

5.3 Training the multidimensional conditional flow

As described in Chapter 3, we will employ the same procedure to instantiate the spline conditional NF, now extended to incorporate higher-dimensional conditional data. During the process of obtaining the results for Chapter 4, it became evident that, within the Bayesian framework structured in Chapter 2, there is no reason to omit other GW parameters. In the previous analysis, we only considered luminosity distance, component masses in the detector-frame, and sky location, since these are the same parameters used by `gwcosmo`. Therefore, in this analysis, we have decided to include all GW parameters, with the exception of phase, as it does not affect the matched-filter SNR. In this section, we briefly outline the training process of the conditional NF and then conduct the bias test, as introduced in Chapter 3, since this test remains the most rigorous evaluation method for a conditional NF.

We trained the conditional NF model with the parameters listed in Tab. 5.1, with two modifications: the number of layers was increased from 5 to 7, and the conditional data space now comprises 12 dimensions, corresponding to the hyperparemters in Tab. 5.1. The NF was trained over 6000 epochs, taking approximately 15 hours on an RTX 2080 GPU. In Fig. 5.6, the loss curves for the training and validation datasets are plotted, along with the

KL divergence values between the marginalised latent space single dimensions and a unit Gaussian distribution. A generalisation gap is evident in the final stages of the training routine, indicating a disparity between the model's accuracy on the training dataset and its performance on a separate validation dataset [152]. This gap can be mitigated by applying regularization techniques during training, such as L1 or L2 regularization and implementing a dropout mechanism [152]. However, this issue was not observed in the other flows trained for each data set.

To evaluate the performance of the new NF model trained with a 12D conditional space, a bias test similar to the one described in Sec 3.2.3 can be conducted. This test involves generating multiple noise-free GW events using the same hyperparameters and combining the posterior distributions from each event to obtain a final posterior distribution for each hyperparameter. However, a more efficient approach is required for performing hierarchical Bayesian inference using testing data, as evaluating the 12D space over finite steps in each direction would be overwhelmingly resource-intensive. Therefore, we propose using nested sampling (NS) [36], which allows for efficient exploration of the joint posterior distribution space. Specifically, we will use the `nessai` (Nested Sampling with Artificial Intelligence) sampler [37, 38], which is particularly well-suited for exploring the posterior distribution of a 12D joint posterior.

5.3.1 Bias Test with `nessai`

Table 5.2: Injected values used for the bias test with `CosmoFlow`.

Parameter	Injected Value
H_0	70.0 km s^{-1} Mpc^{-1}
γ	4.56
k	3.1
z_p	2.4
β	0.1
α	2.4
$M_{\max}$	112.0 $M_\odot$
$M_{\min}$	4.5 $M_\odot$
μ_g	35.0 $M_\odot$
σ_g	3.0 $M_\odot$
λ_g	0.1
δ_m	2.0 $M_\odot$

5.3. Training the multidimensional conditional flow

The `nessai` algorithm is a Bayesian inference tool designed to enhance NS through the use of NFs [37, 38]. By integrating machine learning, specifically NFs, it significantly improves the efficiency of the sampling process compared to traditional NS methods, as detailed in Sec. 1.5. To demonstrate its capabilities, we will apply the `nessai` algorithm to perform a bias test (see Chapter 3) over a 12D parameter space. This analysis will yield a combined joint posterior distribution for the cosmological and population parameters specified in Tab. 5.1.

For this test, we used the set of cosmological and population parameters described in Tab. 5.2. Based on these fixed cosmological and population parameters, 500 simulated GW events were generated using the configuration of HLV detectors from the O3 run. Typically, for each event, we would generate many thousands of posterior samples, which would describe our uncertainty in the parameters estimated from that particular event. These posterior samples represent the distribution of possible parameter values given the data, reflecting the uncertainty due to noise and other factors. However, for this test, we assume no uncertainty in the individual event parameters, and each event is represented by only a single posterior sample corresponding to the true value of the event. This simplification is unusual because it eliminates the uncertainty in the parameter estimation for each event. Even with a finite number of exact measurements (one per event), we still cannot expect to perfectly measure the hyperparameters. Though this approach is somewhat extreme, it reflects the best the `CosmoFlow` pipeline can constrain the parameters, assuming that the data-generation accurately represents how these events are sourced in the universe and that the parameters of the GW can be perfectly constrained.

To perform the hierarchical Bayesian inference using the injected values, we begin by evaluating the posterior distribution of the cosmological and population parameters as follows:

$$\log p(\vec{\Omega}|\mathbf{h},\mathbf{D},I) \propto \log p(\vec{\Omega}|I) + \sum_{i}^{N} \log \left\langle \frac{p(\theta_i|D_i,\vec{\Omega},I)}{p(D_i|\theta_i,I)p(\theta_i|\Omega_0,I)} \right\rangle_{\theta_i \sim p(\theta_i|h_i,I)} . \tag{5.4}$$

Here, $\vec{\Omega}$ represents the set of hyperparameters, which includes the cosmological and population parameters described in Tab. 5.1 (this is the same as in Eq. (2.12)) . We apply Eq. 5.4 to each event, sequentially summing the results across all events. The numerator in Eq. 5.4 is specifically evaluated using the NF model trained on the set of generated events, as detailed in Chapter 3, instead, the probabilities in the denominator, as well as the averaging brackets, can be omitted since each event is represented by a single posterior sample. In this process, we leverage the advantage of batching the data, comprising 500 GW synthetic events, each with a single posterior sample, into the NF for

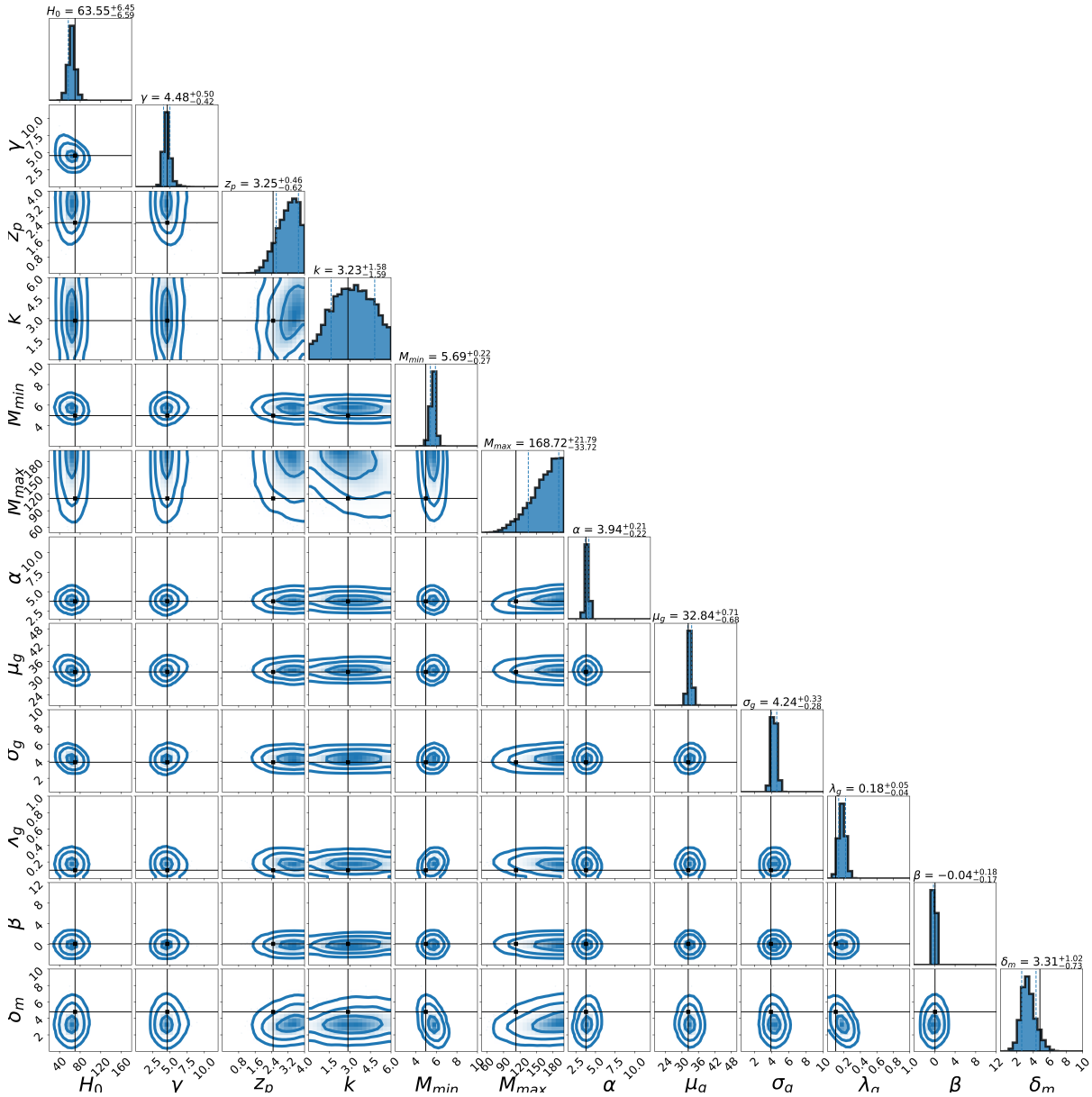

Figure 5.7: Posteriors on 12 cosmological and population parameters obtained by combining the simulated GW posteriors from 500 synthetic BBH events. The true values, listed in Tab. 5.2, are marked by black crosshairs. The figure is a corner plot, where the off-diagonal panels show joint 2D marginalised posteriors, and the diagonal panels present 1D marginalised posteriors. In the diagonal panels, the titles display the median values along with the associated 0.16 and 0.84 quantiles. For clarity, units have been omitted from the plot but are provided in Tab. 5.2. The contours represent the 50%, 90%, and 99% credible intervals, respectively.

a more efficient evaluation of the posterior distribution in Eq. 5.4. At each stage of the `nessai` sampler, we evaluate the posterior using Eq. 5.4, invoking the appropriate NF for each event. Since the 500 synthetic events were generated using an O3 run and HLV detector configuration, we use a single flow model, specifically Flow 4, for this evaluation, as described in Tab. 4.1. The results of the bias test are shown in Fig. 5.7. The `nessai` sampler completed 200,555 likelihood evaluations, yielding 6,245 posterior samples in 1

hour, 38 minutes, and 22 seconds, using 1,000 live points. The stopping criterion was set to $dZ = 0.1$, which is based on the change in the estimated evidence. Further details on the parameter estimation process using `nessai`, as shown in Fig. 5.7, can be found in Appendix C.

This threshold ensures that the sampling process halts once further iterations are unlikely to significantly improve the evidence estimate, balancing computational efficiency with accuracy. Overall, the test results indicate minimal bias across most dimensions; however, there is noticeable bias in the M_{min} and M_{max} dimensions, where the true values lie on the 99% contour. This suggests that while the `CosmoFlow` analysis is largely unbiased with respect to its own data-generation algorithm, some parameters may still exhibit minor bias. This bias, particularly in the M_{min} and M_{max} dimensions, warrants further investigation and could potentially be reduced with additional training and more extensive training data.

The covariance between two variables, X and Y, is formally defined as:

$$\text{cov}(X,Y) = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])], \tag{5.5}$$

where $\mathbb{E}[X]$ and $\mathbb{E}[Y]$ represent the expected values of X and Y, respectively. This formulation is equivalent to Eq. (3.4), expressed in terms of expectation values for the variables X and Y. Using this definition, the covariance matrix provides a comprehensive summary of the linear relationships between pairs of parameters in the joint posterior distribution. It captures how changes in one parameter might correspond to changes in another. Positive covariance values indicate that as one parameter increases, the other tends to increase as well, while negative values reflect an inverse relationship. The magnitude of the covariance signifies the strength of this linear association, with larger absolute values indicating stronger correlations [153].

We present the covariance matrix of the joint posterior distribution in Fig. 5.8, corresponding to the results shown in Fig. 5.7. This matrix offers valuable insights into the dependencies between parameters, enhancing our understanding of their interplay in the model. The diagonal elements of the matrix represent the variances of each parameter, indicating the degree of uncertainty associated with each. Notably, the variance of M_{max} is particularly high, suggesting significant uncertainty in the estimation of this parameter. This high variance reflects a wider spread in the possible values of M_{max}, indicating that the model has less confidence in pinpointing its exact value. For H_0, the variance is approximately 48.1195, implying a standard deviation of about $\sqrt{48.1195} \approx 6.94$. Assuming

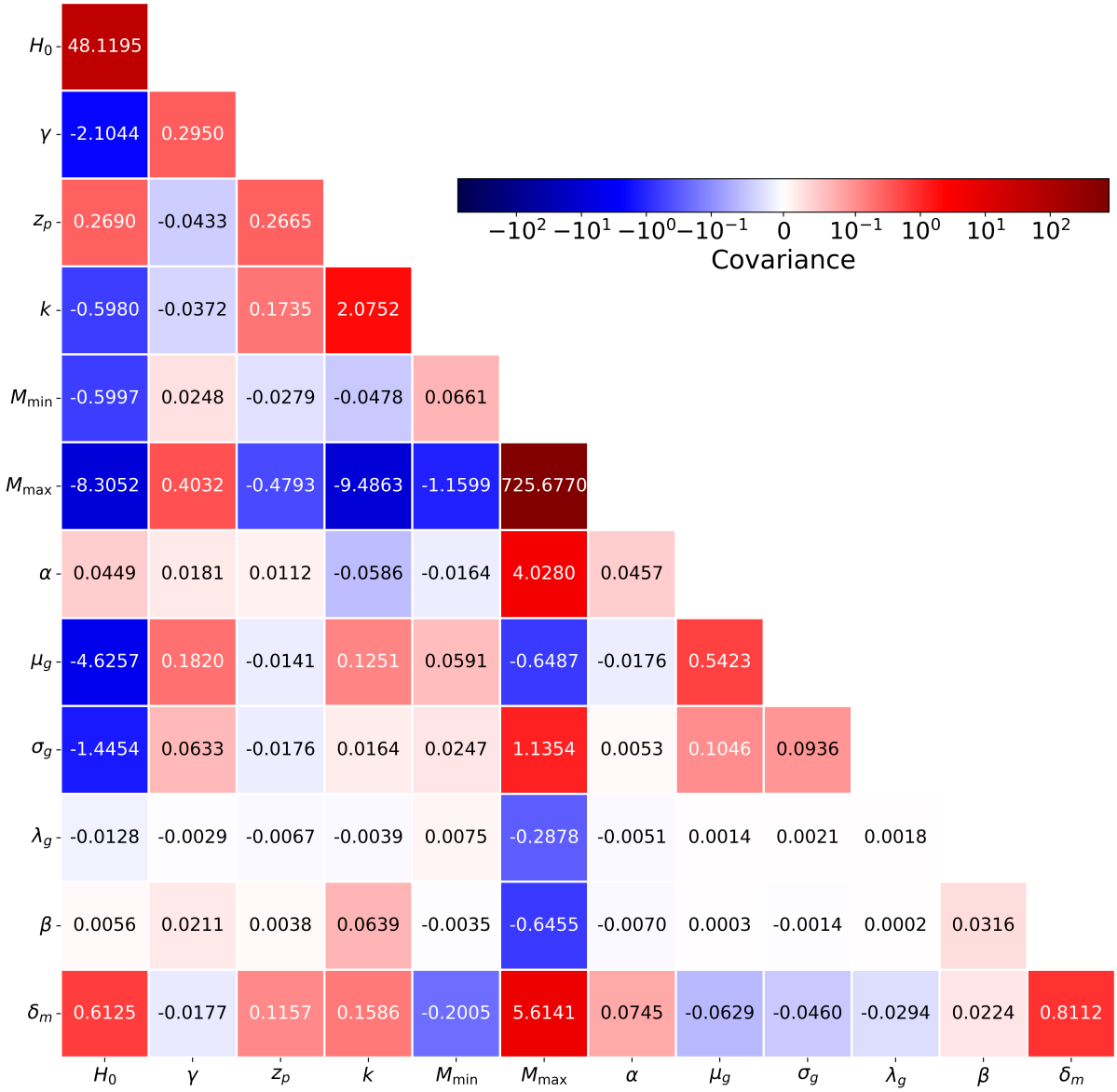

Figure 5.8: Lower triangular part of the covariance matrix obtained from the joint posterior distribution in Fig. 5.7. The colorbar represents the levels of covariance between each pair of parameters. The axes correspond to the 12 hyperparameters of the model, with the diagonal elements representing the variance of each parameter.

a mean value of $H_0 \approx 70\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$, this corresponds to an uncertainty of approximately 9.91%. This standard deviation reflects the spread in the estimation of H_0 given the possible 500 perfectly constrained GW events, indicating that even with this ideal dataset, the uncertainty in estimating H_0 remains at this level.

5.4 Results

5.4.1 Joint inference of cosmological and population parameters

After training the 12D conditional NF and testing its capabilities in constraining the cosmological and population parameters from synthetic GW data (as shown in Fig. 5.7), we are now ready to use this model for hierarchical inference on these 12 parameters using real GW observations. As mentioned in the previous section, the inferred parameters are listed in Tab. 5.1

We performed hierarchical inference on the 12 cosmological and population parameters using 42 BBH GW events, as listed in Tab. 4.1. This analysis was conducted using the `nessai` algorithm. However, unlike the bias test, we now employ 42 GW events selected from the GWTC-3 catalogue, each with a SNR above 11 (see Chapter 4 for details). To evaluate the posterior distribution at various parameter locations, we used Eq. (5.4). For the hierarchical inference of the 12D conditional hyperparameters, we drew 1,000 posterior samples from the parameter posterior distribution of each GW event. These samples were then batched together with prior samples of the conditional hyperparameters to compute the log-likelihood in Eq.(5.4). The specific NF models used in this analysis were made for each group of GW events, taking into account the observational run and the detector configuration. This approach is the same as described in Chapter 4, where the models were trained on datasets corresponding to the particular observation runs and detector setups, with a total of 7 conditional NF models. We set up the `nessai` sampler with 1,000 live points and a stopping criterion of $dZ = 0.1$

The hierarchical inference results are depicted in the corner plot in Fig. 5.9, showcasing the posterior distributions obtained from both `CosmoFlow` and `gwcosmo`. The `gwcosmo` results were generated using its latest version [52], with the `nessai` algorithm configured with 1,000 live points and 8-core parallel processing. The same setup was applied to `CosmoFlow`. Notably, the sampling time for `gwcosmo` was approximately 2.86 days, requiring 144,818 likelihood evaluations. In contrast, `CosmoFlow` completed the sampling in just 7.73 hours, using 1 GPU and performing 120,732 likelihood evaluations. [1] The results highlight discrepancies between the two approaches. For instance, the marginalised posterior distributions for parameters such as H_0, z_p, and k exhibit significant inconsistencies, with `CosmoFlow` revealing features that warrant further investigation. The underlying causes of

1. I would like to express my gratitude to Alex Papadopoulos, a Ph.D. student at the University of Glasgow, for providing the posterior distribution results using `gwcosmo` and the GLADE+ catalogue.

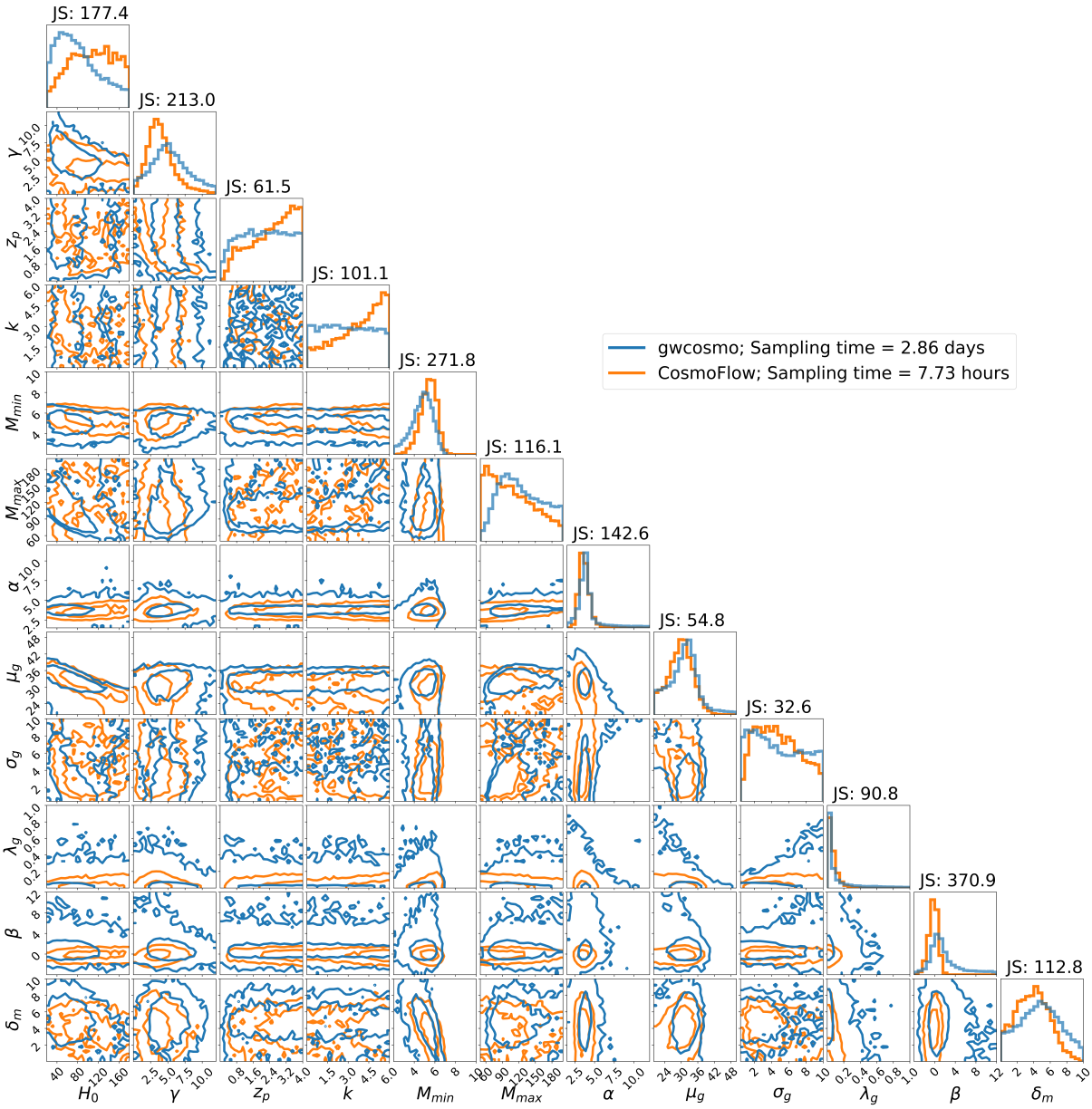

Figure 5.9: Posterior distributions for 12 cosmological and population parameters derived from the 42 BBH events in the GWTC-3 catalogue. Blue lines represent results from `gwcosmo`, while orange lines correspond to `CosmoFlow`. Above each 1D marginalised distribution, the JS divergence between the two analyses is displayed in units of millinats. The legend also includes the approximate sampling times for each analysis. Contour levels indicate the regions containing 50% and 90% of the total posterior distribution.

these discrepancies remain unclear. Ongoing and future work will focus on analysing the contributions of individual events with the aim of identifying the sources of inconsistency more precisely. It is suspected that these differences might stem from the restriction of the redshift space via $z_{\max}$ (described in Sec. 2.6.2), or possibly from the need for more extensive and efficient training of the conditional NF to better capture the parameter space.

Another notable observation is the differing levels of parameter constraints between `CosmoFlow` and `gwcosmo`. Certain parameters, such as α, β, and λ_g, show tighter constraints in the `CosmoFlow` results compared to `gwcosmo`, raising questions about the methods used to constrain these parameters. While it is possible that `CosmoFlow` might be overly restrictive in its parameter constraints, this remains difficult to confirm, despite efforts to align the analysis closely with `gwcosmo`. Remarkably, the `CosmoFlow` analysis was approximately 10 times faster than the `gwcosmo` analysis, using 1,000 posterior samples from each GW event in the GWTC-3 catalog, with the same sampler and number of live points. This speedup allows for the analysis of a larger number of events within the same timeframe as `gwcosmo`, enabling more precise constraints on both cosmological and population parameters. This capability will be increasingly important given the expected surge in events from the upcoming O4 and O5 observation runs. Nonetheless, further improvements are needed to achieve a better match between the two analyses.

We also present the Pearson correlation matrix [153] values for each parameter derived from both `gwcosmo` and `CosmoFlow` in Fig. 5.10. Pearson's correlation, denoted as $\rho(X,Y)$, is a measure of the linear relationship between two variables X and Y. It is defined mathematically as

$$\rho(X,Y) = \frac{\mathrm{Cov}(X,Y)}{\sigma_X \sigma_Y}, \tag{5.6}$$

where $\mathrm{Cov}(X,Y)$ is the covariance of X and Y, as defiend in Eq.(3.4), and σ_X and σ_Y are the standard deviations of X and Y, respectively. The resulting value ranges from -1 to $+1$, with -1 indicating a perfect negative linear relationship, $+1$ indicating a perfect positive linear relationship, and 0 indicating no linear relationship. In comparing the Pearson correlation matrices from both `gwcosmo` and `CosmoFlow`, we observe that while there are clear differences in correlation values between the two approaches, some correlations remain consistent across both methods. Notably, the parameter pairs (H_0, γ), (H_0, μ_g), $(H_0, M_{\min})$, and $(\delta_m, M_{\min})$ exhibit strong anti-correlations in both analyses. Similarly, the pair (λ_g, σ_g) shows a high positive correlation in both approaches, indicating agreement between the two methods for these specific parameter relationships. We also note that the negative correlation between H_0 and γ of -0.2927 is well understood. The parameter γ models a power-law increase in the merger-rate with redshift; hence, higher values of γ support lower values of H_0. This occurs because lowering H_0 places events at lower redshifts, which are incompatible with the observed mass distribution [16]. As a result, γ compensates by favoring higher redshifts. This result, observed in `CosmoFlow`, aligns with the findings from Ref. [16], showing similar correlations between H_0 and γ.

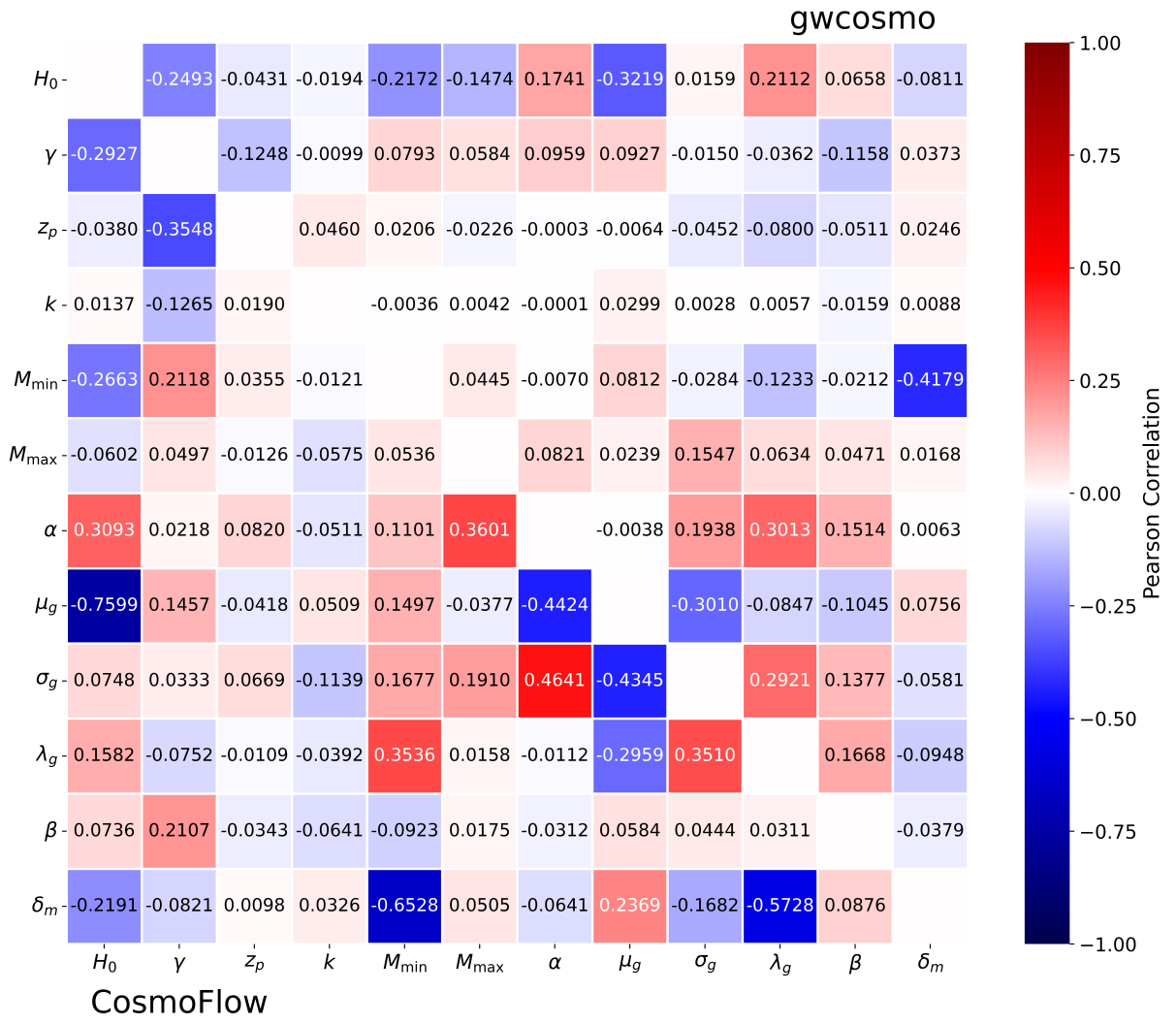

Figure 5.10: Pearson correlation matrices for the joint posterior distributions from `CosmoFlow` (lower triangular) and `gwcosmo` (upper triangular) are presented. These matrices correspond to the results shown in Fig. 5.9. The colorbar reflects the strength of the correlation between parameter sets, ranging from -1 (strong negative correlation) to +1 (strong positive correlation). Ideally, for identical analyses, the upper right and lower left triangles should mirror each other.

However, beyond these consistencies, a broader examination reveals notable inconsistencies. The overall correlation degrees between parameters are generally higher in the `CosmoFlow` results compared to those from `gwcosmo`. This suggests that `CosmoFlow` might be identifying stronger relationships between parameters than `gwcosmo`, which raises questions about the underlying processes driving these correlations. Understanding why `CosmoFlow` produces these stronger correlations will be an important focus of future work. Investigating these differences further will help determine whether these higher correlations are an artifact of the method or if they reveal a deeper, more complex relationship between the parameters that `gwcosmo` might not fully capture.

We finally performed the hierarchical inference of 42 BBH GW events, demonstrating that `CosmoFlow`, in the future, could compete with `gwcosmo`. To determine which analysis is best suited for inference, a Mock Data Challenge (MDC) would allow us to test how these two pipelines constrain the parameters of interest by injecting synthetic GW events with fixed cosmological and population parameters. In this MDC, synthetic GW data would be generated based on predetermined cosmological and population parameters, simulating a realistic set of BBH events. Both `CosmoFlow` and `gwcosmo` would then analyse this mock dataset to recover the injected parameters. The challenge would evaluate the pipelines on accuracy, precision, computational efficiency, and robustness across varying scenarios. A key goal would be to determine which pipeline most effectively constrains the parameters while handling the complexities of large datasets and realistic noise conditions.

Furthermore, the `CosmoFlow` analysis is highly flexible, allowing various parameters to be sampled within the analysis. For instance, one assumption in the data-generation process is when sampling from the luminosity function, each galaxy is weighted by its intrinsic luminosity. This assumes that GW BBH events are linearly correlated with luminosity. However, this is only an hypothesis, and the true universal relationship remains uncertain. It is important to recognize that luminosity is derived from the observed apparent magnitude in a specific band, and different bands can influence the inferred correlation.

5.4.2 Beyond gwcosmo

The `gwcosmo` pipeline is a well-known tool for estimating cosmological parameters from GW events, primarily focusing on H_0. However, it hasn't yet been used to infer other cosmological parameters. In this section, we expand the analysis to include multiple cosmological parameters, showing how we can both marginalise over them and directly infer their values. Along with traditional parameters like H_0 and Ω_m, we introduce two additional parameters: w_0 and η. The parameter w_0 describes the equation of state of dark energy, which is a fundamental aspect of cosmology that relates the pressure p of dark energy to its density ρ through the equation $p = w_0\rho$ [154]. The parameter η represents the correlation between the intrinsic luminosity of host galaxies in a specific magnitude band and the likelihood of those galaxies producing a GW event. This parameter has been employed in studies like the recent `IcaroGW` [155], where it is referred to as ε, though they only consider two cases: $\varepsilon = 1$ and $\varepsilon = 0$. Luminosity here refers to the intrinsic EM radiation output from galaxies, measured in a specific band of wavelengths. For our analysis, we use the K-band from the GLADE+ catalogue, which captures near-infrared light typ-

ically associated with stellar mass. The parameter η modifies how much this luminosity influences the probability of a galaxy being a host, using the function L^{η}, where L is the galaxy's luminosity. By sampling η from a prior distribution, we can account for its uncertainty and use the data to better constrain this relationship.

In this analysis, we consider the three cosmological parameters, H_0, Ω_m, and w_0, alongside η. Including η allows us to better understand how galaxy luminosity relates to the chances of a GW event occurring. We start by exploring how changes in Ω_m, w_0, and η affect the distance-redshift relationship and the luminosity-weighted probability function, specifically when using the K-band luminosity. After this, we compute the joint posterior distributions of these parameters using 21 GW events from the GWTC-3 catalogue detected by the HLV detectors network during the O3 run. This section is still in the early stages of development; therefore, the analysis currently includes only a subset of the events. This is because we have trained only one conditional NF model using training data-generated with the O3 HLV setup, varying the parameters H_0, Ω_m, w_0, and η. Although this analysis isn't fully complete, future work will include all the events, providing a more comprehensive picture.

We begin by defining the complete luminosity-distance function, which is central to our analysis

$$D_{\mathrm{L}}(z, H_0, \Omega_{\mathrm{m}}, w_0) = (1+z)\frac{c}{H_0}\int_0^z \frac{dz'}{\sqrt{\Omega_{\mathrm{m}}(1+z')^3 + (1-\Omega_{\mathrm{m}})(1+z')^{3(1+w_0)}}}, \tag{5.7}$$

where c is the speed of light, Ω_{m} is the dimensionless matter density parameter, w_0 is the dark energy equation of state parameter, H_0 is the Hubble constant, and z is the redshift [154]. This equation is derived from the standard cosmological model and is used to estimate the luminosity distances and the prior redshift distribution of GW host galaxies in our data-generation process, as detailed in Sec. 2.4.1. The choice of the prior distributions for the cosmological parameters Ω_{m} and w_0 is informed by both theoretical considerations and observational constraints. We adopt uniform priors within the ranges $[0,1]$ for Ω_{m} and $[-1.9, 0.9]$ for w_0. The selected range for Ω_{m} reflects our current understanding that the matter density parameter lies within this interval, encompassing both matter dominated and dark energy dominated universes, assuming a flat ΛCDM model. The prior for w_0 was chosen to be uniform in the range $[-1.9, 0.9]$ to ensure it covers the cosmological constant scenario ($w_0 = -1$), as well as alternative models with a weaker or stronger dark-energy component [4].

The parameter η, which modifies the probability of a galaxy being a GW event host based on its luminosity, is assigned a uniform prior over the range $[0.5, 3.0]$. This range is chosen to explore a wide variety of plausible scenarios for how galaxy luminosity might influence the probability of hosting a GW event. The lower and upper bounds of the prior distribution of the η parameter are derived from [156], where it is shown that the relation between halo/subhalo mass and hosted galaxy luminosity is well fit by a double power law, which asymptotes to $L \propto M^4$ at low mass, while at high mass, the former follows $L \propto M^{0.28}$. The assumption we make here is that the probability of a host galaxy sourcing a CBC event is directly proportional to the halo mass of the galaxy, which is proportional to the actual mass of the galaxy. High mass halo galaxies, such as large elliptical and spiral galaxies, typically have higher stellar densities and star formation rates. This increased stellar density implies a greater number of massive stars that can evolve into compact objects like black holes and neutron stars, which are progenitors of GW events. High mass halos provide such environments due to their large stellar masses. Studies show that binary black hole mergers are more likely to occur in galaxies with higher stellar densities [157].

Next, when calculating the luminosity-weighted Schechter function, we modify the traditional approach by multiplying the Schechter function by a factor of L^η instead of the usual L. This adjustment alters the final CDF derived in Sec. 2.4.4 (Eq. (2.51)) as follows:

$$\mathrm{CDF}(L|\eta) = \frac{\Gamma(\alpha+1+\eta, L_{\min}/L^*) - \Gamma(\alpha+1+\eta, L/L^*)}{\Gamma(\alpha+1+\eta, L_{\min}/L^*) - \Gamma(\alpha+1+\eta, L_{\max}/L^*)}. \tag{5.8}$$

In this equation, the index factor in the CDF changes from $\alpha+2$ to $\alpha+1+\eta$, reflecting the impact of the new parameter η. However, it is important to note that the luminosity ratios within this CDF remain independent of the cosmological parameters. This independence arises because, in applying Eq. (5.8), we continue to use the same cosmological framework that was employed when inferring the Schechter function parameters (α, M^*, and ϕ^*), as outlined in Sec. 2.4.2. Specifically, this framework uses the following fixed cosmological parameters: $(H_0, \Omega_\mathrm{m}, w_0) = (100\,\mathrm{km\,s^{-1}\,Mpc^{-1}}, 0.3, -1)$. Consequently, the ratios of luminosity remain invariant to the cosmological values used. Next, using Eq. (5.8), we sample luminosities from a specific band, which alters $L_{\min}$, $L_{\max}$, and α accordingly, depending on the band, in this case, the K-band. We can then sample from each different luminosity distribution using the *snakes* algorithm described in Sec. 2.5.2. Once the luminosities are sampled, they are converted into absolute magnitudes using

$$M = M_\odot - 2.5 \log_{10}\left(\frac{L}{L_\odot}\right), \tag{5.9}$$

where $M_\odot$ and $L_\odot$ are the absolute magnitude and luminosity of the Sun, respectively. These absolute magnitudes are initially sampled based on the fixed cosmological parameters $(H_0, \Omega_\mathrm{m}, w_0) = (100\,\mathrm{km\,s^{-1}\,Mpc^{-1}}, 0.3, -1)$. However, to align them with the actual cosmological parameters used during the sampling process, we need to adjust the magnitudes. This adjustment is done using the following equation:

$$M(H_0, \Omega_\mathrm{m}, w_0) = M_0 + 5\log_{10}\left(\frac{D_L(H_0, \Omega_\mathrm{m}, w_0)}{D_{L,0}}\right), \tag{5.10}$$

where M_0 is the absolute magnitude calculated using the initial fixed cosmological parameters, $D_L(H_0, \Omega_\mathrm{m}, w_0)$ is the luminosity distance corresponding to the actual cosmological parameters, and $D_{L,0}$ is the luminosity distance corresponding to the initial fixed parameters. This conversion ensures that the magnitudes reflect the correct cosmological context for the sampled data.

We then proceed with the data-generation process as outlined in Chapter 2. After sampling a sky location and associating it with the corresponding magnitude threshold value, we determine whether to retain the sample from the prior distributions or to resample a new host galaxy from the GLADE+ catalogue. In the previous 1D analysis focused solely on H_0, when sampling from a specific pixel in the catalogue, the luminosities within that pixel were evaluated, and the weighting factor for each galaxy was computed as

$$w_i = \frac{L_i}{(1+z_i)}, \tag{5.11}$$

where L_i is the luminosity of the ith galaxy and z_i is its redshift. However, in our current multidimensional analysis where we vary the $(H_0, \Omega_\mathrm{m}, w_0, \eta)$, the weighting factor is modified to account for these variations. The new weighting factor is given by:

$$w_i = \frac{L_i^\eta(H_0, \Omega_\mathrm{m}, w_0)}{(1+z_i)}, \tag{5.12}$$

where L_i is raised to the power of η, reflecting the correlation between luminosity and the probability of a galaxy hosting a GW. Since luminosity scales linearly with H_0, the weighting factor is independent of H_0, but not of the other parameters. All other parameters, including the merger-rate parameters (z_p, γ, k) and the population mass model parameters, are held constant at the same values used in the 1D analysis described in Chapter 4.

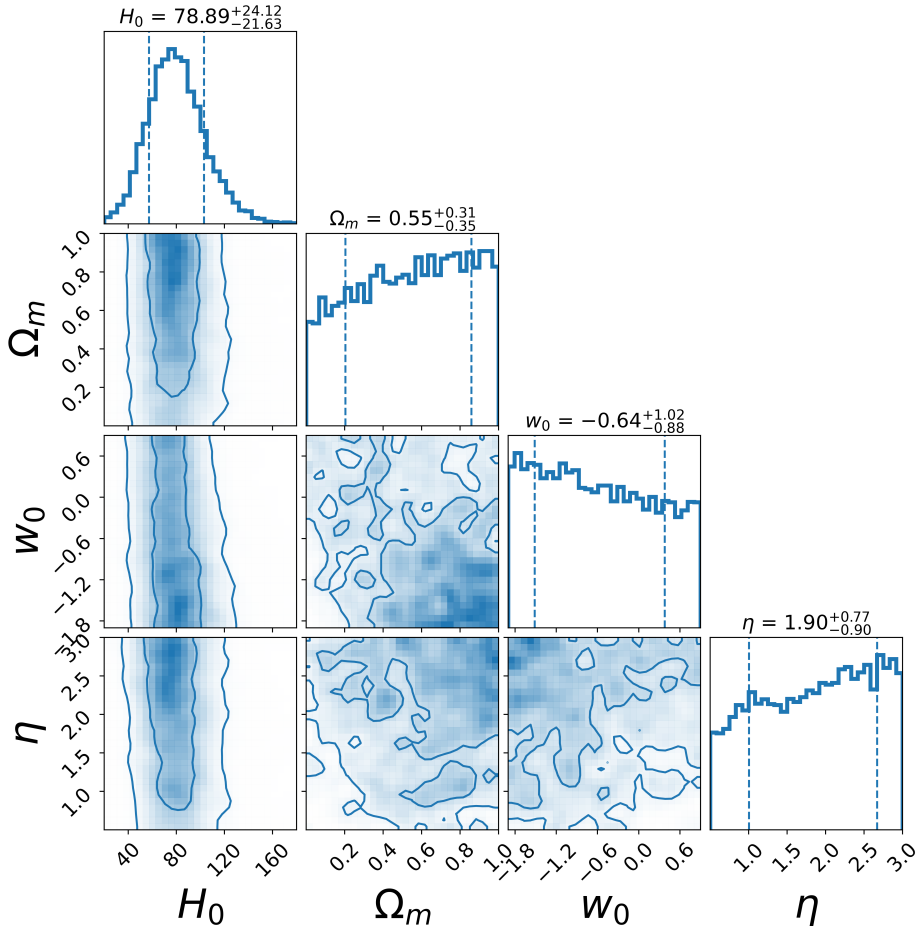

Figure 5.11: Corner plot of the joint posterior distributions for the parameters H_0, Ω_m, w_0, and η. The histograms along the diagonal display the marginalised posterior distributions for each parameter, with the 16% and 84% quantiles highlighted by dashed lines. The off-diagonal panels show the joint posterior distributions between pairs of parameters, with contour levels representing 50% and 90% of the enclosed data. These results are based on the analysis of 21 GW events from the GWTC-3 catalogue during the O3 run using the HLV detector network.

We then trained a NF model with a 14D target space (the GW parameters, excluding phase) and a 4D conditional space for H_0, Ω_m, w_0, and η, using a total of one million data points. The training of the NF model was performed with the following parameters: batch size of 50,000, 1,000 epochs, shuffling enabled, no activation function, no dropout rate, learning rate of 0.0005, Adam optimizer, LU linear transform, 64 neurons per layer, 5 transforms, 2 blocks per transform, 14 inputs, 4 conditional inputs, CouplingNSF flow type, no xyz transformation,and the MinMax scaler (see Chapter 3). We then used the trained NF model to estimate the 4 conditional parameters using 21 GW events from the GWTC-3 catalogue. The results of the parameter estimation process, obtained by combining 21 events from the GWTC-3 catalogue observed during the O3 run with HLV detectors, using `CosmoFlow` as the core likelihood function described in Eq.(5.4), and performed with `nessai` using 1000 live points and a stopping criterion of $dZ = 0.1$, are shown in Fig. 5.11.

As anticipated, the results mainly show that H_0 is well-constrained, which is consistent with our previous 1D analysis presented in Chapter 4. However, due to the limited number of events used, we didn't achieve strong constraints for Ω_m, w_0, and η. Additionally, we couldn't directly compare our results with `gwcosmo` since it has yet to provide estimates for cosmological parameters other than H_0. Future results will also be compared with other pipelines, such as `IcaroGW` [15], which do give estimates on Ω_m and w_0. One important takeaway from this analysis is the added complexity that comes with introducing new parameters into the model. Whether these parameters are cosmological or not, like η in our case, adding them makes the data-generation and analysis processes more complicated. The parameter η, for example, was introduced to explore how galaxy brightness might relate to the probability of hosting a GW event.

Looking ahead, our next steps will include performing detailed tests with synthetic data to check for any biases in our current analysis. We want to understand how well we can estimate these parameters given a fixed number of events and to see if there are any areas where our method might be improved. We also plan to develop better strategies for dealing with the increased complexity that comes with adding more parameters. This might involve finding more efficient ways to handle the data or improving how we guide the training data process when dealing with multiple parameters. Finally, we hope that in a future MDC, `CosmoFlow`, `gwcosmo` and `IcaroGW`, and other pipelines, can be tested on synthetic data to see how well they can recover parameters like Ω_m, w_0, and η. This will give us valuable insights into how these pipelines perform and help us refine our methods for future studies. Despite the challenges, this analysis demonstrates the flexibility and potential of `CosmoFlow` in incorporating different types of parameters and models. This makes it a strong candidate for future cosmological parameter estimation.

5.5 Conclusion

In this chapter, we explored obtaining joint posterior distributions of multiple cosmological and population parameters using a trained conditional NF. Initially, we generated training data using the algorithm presented in Chapter 2, varying not one but twelve conditional parameters, specifically H_0, the cosmological parameter, the merger-rate parameters, and the population parameters first described in Sec. 2.4.1 and Sec. 2.5.1. These parameters

were sampled from the same priors presented in Ref. [16] and listed in Tab. 5.1. After generating data for each detector setup and observational run (7 datasets of 300,000 data points each), we trained a flow for each setup, subsequently applying the specific NF to the GW events observed with their respective detector setups and PSDs.

We then tested the NF performance by conducting a bias test, as described in Chapter 3, but this time over a 12D parameter space. This was carried out using the `nessai` sampler [38], which leverages NFs to enhance the NS process. The results of the bias test, shown in Fig. 5.7, reveal an overall consistent joint posterior distribution with the true values listed in Tab. 5.2. The only two parameters showing some degree of bias are $M_{\rm min}$ and $M_{\rm max}$, both within the 99% contour. This result underlines the ability to constrain the parameters of interest, assuming a perfectly localised and noise-free event. Even after analysing 500 GW events, our precision over H_0 remains within 10%. Therefore, to resolve the H_0 tension between the Planck measurements [5] and SH0ES [6], many more GW events need to be analysed.

Subsequently, we used the trained NF to perform hierarchical Bayesian posterior estimation using 42 GW events from the GWTC-2 and GWTC-3 catalogues [99]. The results, shown in Fig. 5.9, compare the `gwcosmo` and `CosmoFlow` outcomes, revealing both consistencies and inconsistencies across various parameters. Notably, the marginal z_p, k, and H_0 parameters show little alignment with the `gwcosmo` results. This discrepancy suggests potential issues in the data-generation process, which could stem from inaccuracies in estimating GW selection effects, such as using the SNR approximator (MLP) or redshift thresholding errors with $z_{\rm max}(H_0, \rho_{\rm th})$. Additionally, there may be issues in how galaxies are sampled from the galaxy catalogue. Other discrepancies include more constrained power law indices of the population PLP models, α and β, compared to `gwcosmo`, warranting further investigation. Consistencies were observed in parameters like γ, $M_{\rm min}$, and μ_g. Importantly, the results in Fig. 5.9 highlight `CosmoFlow`'s faster sampling time, approximately 10 times quicker than `gwcosmo`.

We also presented results using 21 GW events from the O3 run (with an HLV detector configuration) to estimate three cosmological parameters plus one additional parameter: H_0, $\Omega_{\rm m}$, w_0, and η. The last parameter, η, is the luminosity index for weighting each host galaxy sourcing a GW event. This analysis demonstrates `CosmoFlow`'s flexibility and adaptability to various models and parameterizations. The results, shown in Fig. 5.11, indicate that while there is limited information to constrain the parameters (except for H_0), the results are consistent with the 1D combined posterior presented in Chapter 4. Although the current dataset doesn't provide strong constraints on these additional cos-

mological parameters, we expect this to change as more events are accumulated. A future bias test analysis could be valuable in determining the minimum number of high-quality events needed to achieve meaningful constraints for these other parameters, offering insights into how the results will improve with a larger dataset.

Overall, the multi-dimensional posterior distributions presented in this chapter should be taken cautiously, as the results are from early stages of the analysis. Nevertheless, they are promising, showing both consistencies and inconsistencies with the `gwcosmo` analysis. Future work will involve further testing the multi-parameter data-generation process, aiming to speed it up and reduce any systematic biases present. Ultimately, a mock data challenge will allow the two algorithms to compete under controlled conditions, revealing any biases and flaws systematically present in each algorithm.

Chapter 6

Conclusion and Future Work

The expansion of the universe has long been a focus of study for astrophysicists, raising fundamental questions about the nature of the cosmos. Today, we have a clearer understanding that the universe is continuously expanding. However, the exact rate of this expansion remains uncertain. Measurements from the early universe, such as those from the Planck mission, suggest an expansion rate of $H_0 = 67\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$ [5], while more recent observations by the SH0ES team report a higher value of $H_0 = 74\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$ [6]. These two results are in conflict, differing by about 4.4σ. To date, no systematic errors have been identified in either the Planck or SH0ES measurements, leading to the possibility that either one is incorrect, or there is an aspect of the universe's expansion that we have yet to understand.

In 1986, Bernard F. Schutz wrote a seminal paper [12] describing the potential measurement of the expansion rate of the universe using GW events. He associated the luminosity distance of these exotic astrophysical events with redshift information from galaxy catalogues, opening up a new branch of astrophysics that includes both cosmology and GW astrophysics. One significant advancement in parameter estimation of the Hubble constant using Schutz's idea and standard sirens was the publication of the `gwcosmo` analysis [51]. This analysis performed parameter estimation on H_0 using standard sirens, such as CBCs events including BBH and BNS and NSBH mergers, with and without EM counterparts. It integrated galaxy catalogues to obtain redshift information and accounted for the incompleteness of EM catalogues. This groundbreaking analysis led to the LIGO-Virgo-KAGRA collaboration presenting their cosmology-related paper in 2022, showing an inference of H_0 from both dark and bright standard sirens [16]. Ever since, the `gwcosmo` analysis has been at the forefront of cosmological GW analysis tools. However, it has one drawback: the inference power, although efficient, is not fast enough to handle the upcoming surge of events expected from future detector generations, such as the O5 generation.

This thesis focused on the development of a machine learning-based tool for estimating cosmological parameters, named `CosmoFlow`. Since 2021, my work has centered on creating `CosmoFlow` as a bench-marking tool against `gwcosmo`, designed for future GW detections. The `CosmoFlow` analysis is intended to provide rapid and accurate estimations of cosmological parameters, preparing us for the increase in data from next generation GW detectors and potentially contributing to resolving the H_0 tension. This machine learning-driven approach offers a new method for inferring cosmological parameters from GW events. In this thesis, I detailed the various components of the `CosmoFlow` model and explained how it operates.

After providing an introduction to cosmology, the Hubble tension, GW, and ML in Chapter 1, Chapter 2 focused on the `CosmoFlow` analysis. This chapter provided a comprehensive overview of the methods used, detailing the generation of training data for the *conditional* NF model that was employed to estimate posterior distributions of cosmological and population parameters based on GW posterior samples from detected events. Chapter 2 described the Bayesian framework in which `CosmoFlow` operates and outlined the process of generating the distribution $p(\theta|\Omega,D)$, representing the probability distribution of GW parameters conditioned on detectability and cosmological and population parameters. The data generation process started by sampling cosmological parameters, focused initially on varying only H_0. For each sampled H_0, a redshift was sampled from a prior independent of H_0, followed by calculating the corresponding luminosity distance. Sky locations were sampled uniformly, and host galaxy luminosities were drawn using a Schechter function fitted to the GLADE+ catalogue. This function characterised the luminosity distribution of galaxies, and GLADE+ was chosen for its extensive sky coverage, despite its limited depth with only 20% completeness at 800 Mpc. Apparent magnitudes of sampled galaxies were compared to a magnitude threshold map of GLADE+, determining whether galaxies are considered part of the catalogue or not. This step helped refine the catalogue's boundaries, allowing accurate integration of EM information. Detected galaxies with apparent magnitudes brighter than the threshold were substituted with real galaxies from GLADE+, matched by location, luminosity, and redshift.

Next, the intrinsic GW parameters were sampled, with all but the source masses, m_1 and m_2, drawn from uniform priors. The masses were sampled from a power law plus peak (PLP) distribution, which captured the observed trend of a high number of low-mass BBHs and fewer high-mass BBHs. The matched-filter SNR was then computed for each event, considering the contributions of each detector setup and applying an SNR threshold of 11, consistent with the `gwcosmo` approach to account for selection effects. The key aspect of this approach was maintaining consistency in the cosmological parameter distribution conditioned on detectability. By reusing the initial H_0 sample for undetected

events, the method prevented biased trends toward higher H_0 values, ensuring that the model explores regions of the parameter space evenly and effectively. This methodology allowed for robust training of the NF model without skewing the data distribution based on detectability.

In Chapter 3, we focused on training the NF and evaluated its performance using injected GW events with fixed cosmological and population parameters. The training was performed using `glasflow` [135], built on the widely used `nflows` module [136]. The primary goal was to condition the NF on cosmological parameters to transform the input data comprising the 5D space of luminosity distance, sky location, and detector frame component masses—into a simple latent space distribution, such as a unit Gaussian. To achieve this transformation, the Jacobian of the change of variable equation was parameterised and optimised, ensuring that it remained invertible and computationally feasible. Training was conducted for each detector setup and observational run using one million data points per flow and a consistent set of hyperparameters. The results demonstrated that the latent space marginalised distributions closely approximate a unit Gaussian, with JS divergences around 0.1 millinats, indicating successful modeling of the training data.

To further assess the performance of the NF, three key tests were conducted:

1. *Resampling Test*: New samples were drawn from the conditional space, with H_0 sampled from a uniform prior, and passed through the trained NF. This test evaluated the ability of the NF to recover original data from the latent space, though some inefficiencies were noted in sampling distributions with hard boundaries, which can be mitigated with transformations like the *logit* function.
2. PP *Test* : This test involved sampling both the input and conditional spaces to identify any systematic biases. The p-values were plotted for the 14 target dimensions and the single conditional dimension, providing insights into the overall accuracy and potential biases of the model.
3. *Bias Test*: Synthetic data with a fixed H_0 value was generated, and the main equation for the posterior of H_0 was applied to noise-free GW events. By combining posteriors from multiple events, hierarchical inference was performed to obtain the final posterior over H_0. Results showed minimal systematic bias when combining 1000 events, highlighting the robustness of the model.

In Chapter 4, the trained conditional flow was used to estimate the 1D posterior distribution of H_0 and compared the results with those from `gwcosmo`. Although the initial results were promising, they highlighted the need for further development. Benchmarking was conducted to assess the consistency and validity of `CosmoFlow` results compared to

`gwcosmo`. Posterior samples from 42 BBH events observed during the O1, O2, and O3 runs (catalogued in GWTC-2.1 and GWTC-3) were analysed. Seven datasets were generated by considering different combinations of detector setups and observing runs, and a separate NF model was trained for each to learn the data distribution. A sanity check was performed before estimating the single event posterior distributions by using the GW posterior sample distribution as "cookie cutters "to segment the generated data distribution and examine the H_0 distribution. This step validated the data generation process by showing distinct contributions from in catalogue and out of catalogue events. The individual 1D posterior distributions of H_0 for each GW event were then evaluated and compared between `CosmoFlow` and `gwcosmo`. The comparisons showed good agreement overall, though some discrepancies were observed, possibly due to data sampling issues.

The combined posteriors provided an overall estimate of $H_0 = 76.51^{+15.15}_{-11.54}\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$ using the `CosmoFlow` analysis. The analysis was further extended to include NSBHs, with adjustments made to the priors. The results displayed varying degrees of consistency with `gwcosmo`, with some mismatches likely arising from selection effects or differences in data coverage, particularly when comparing the well localised event GW190814. Additionally, the potential of the trained conditional NF to accelerate the inference process was explored by batching GW posterior sample data, allowing vectorised calculation of the posterior distribution. This approach achieved a significant speed up, enabling one likelihood evaluation for 1000 posterior samples in about 4 ms, thus enhancing the efficiency of joint parameter estimation analyses.

In Chapter 5, the analysis was extended to consider multiple cosmological and population parameters, including H_0 and various parameters describing the black hole mass distribution and event rate evolution. The data generation process was changed to account for the variation of these additional parameters, and a conditional NF model was trained on this expanded multi-parameter data to handle the higher-dimensional space. The NF was trained using a 14-dimensional target space, incorporating all GW parameters for BBH events except the phase component. The model's performance was validated through a bias test using 500 synthetic GW events with fixed cosmological and population parameters. The test indicated minimal bias, suggesting that the `CosmoFlow` analysis is inherently unbiased with respect to its data generation algorithm. To efficiently explore the complex parameter space involving numerous parameters, a machine learning-driven nested sampling approach, `nessai`, was employed, allowing for the retrieval of posterior distributions when combining 500 events.

Hierarchical inference of the cosmological and population parameters was performed by estimating the joint posterior distributions using real GW events. The results were compared to those from `gwcosmo`, showing good consistency for some parameters but discrepancies for others. The `CosmoFlow` analysis was notably faster, allowing for the analysis of more events in less time. The flexibility of `CosmoFlow` was further demonstrated by exploring the inclusion of additional cosmological parameters beyond H_0, such as Ω_m and the dark energy equation of state parameter (w_0). This was achieved by training the NF model to vary these parameters while keeping the population parameters fixed. The joint posterior distributions of H_0, Ω_m, w_0, and a new parameter η (describing the correlation between galaxy luminosity and the likelihood of hosting a GW event) were estimated using 21 GW events. While the results did not reveal significant new features, they demonstrated `CosmoFlow`'s capability to efficiently introduce and infer new parameters with ease.

In the future, we see `CosmoFlow` playing a key role in the LIGO collaborations by inferring cosmological parameters such as H_0, as well as other fundamental parameters that describe our universe. This tool could be crucial for analysing the vast number of GW events expected with the upcoming enhancements in detector sensitivity, potentially handling thousands or even tens of thousands of events. Additionally, `CosmoFlow` aims to contribute to resolving the Hubble tension. We also plan to evaluate `CosmoFlow` against other pipelines in a Mock Data Challenge (MDC) to identify and understand any systematic biases present in both `CosmoFlow` and alternative methods. This comparison will help us refine the most effective approach for inferring cosmological parameters.

Future work will explore several ways to expand this analysis. One approach is to generate training data that includes all detector setups and PSDs using a one-hot encoding mechanism. This would allow a single flow to be trained for the entire analysis, avoiding the need to store multiple separate conditional NF models. Instead of encoding the detector data, the NF could also be conditioned on the time-of-day PSD, which would help avoid using a single time-averaged PSD for the entire observing run. Another direction is to use multiple galaxy catalogues that cover different depths and sky areas to generate training data. This approach would allow the integration of multiple EM catalogues, increasing the EM information available when inferring H_0. We also plan to improve the process of generating synthetic GW data quickly, potentially up to millions of data points, which would be highly valuable for scientists conducting population studies or creating mock data challenge (MDC) data. Overall, `CosmoFlow` has shown to be adaptable and capable of addressing new challenges, helping us investigate the relationship between cosmological parameters and GW events. Continued development will enable deeper analysis and understanding of these complex interactions.

Finally, as we move forward in the field of cosmology, it's becoming increasingly clear that integrating machine learning tools into our analyses is essential. These advanced techniques can handle complex problems, like inferring the Hubble constant from GW events, with greater efficiency and accuracy. Embracing machine learning will help us tackle the challenges of the future and deepen our understanding of the universe's expansion. It's time to adapt and innovate, using the best tools available to push the boundaries of our knowledge.

Appendix

A Appendix A: SNR approximator with MLP

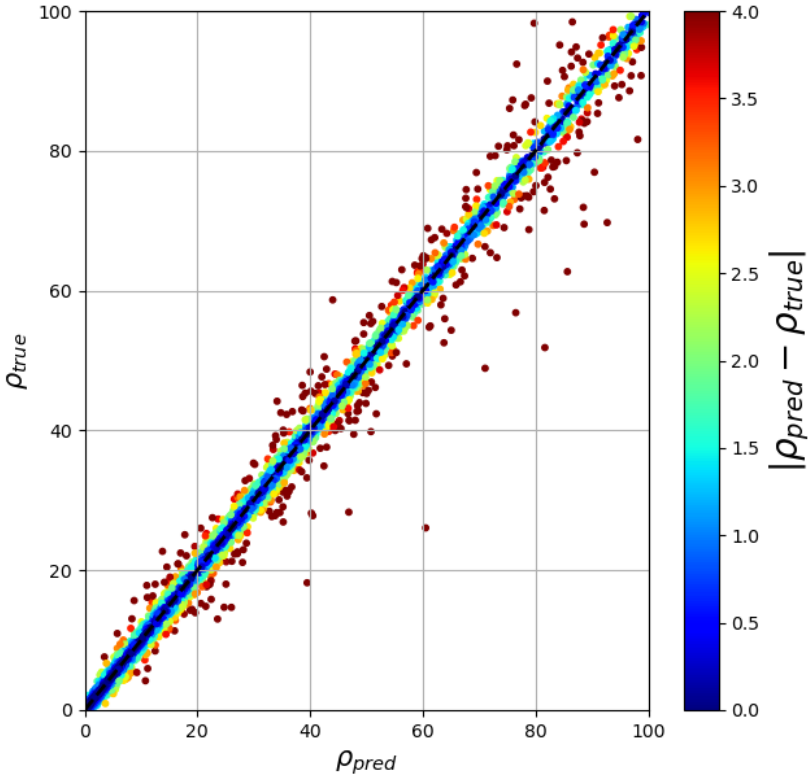

Figure A1: Comparison of predicted vs. true SNR values (V1, O3).

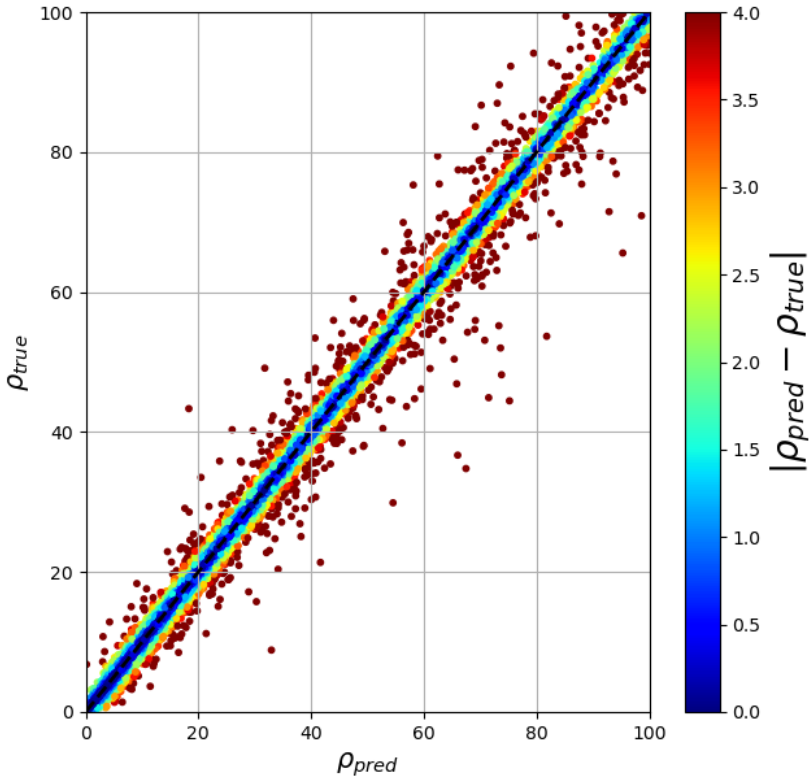

Figure A2: Comparison of predicted vs. true SNR values (L1, O3).

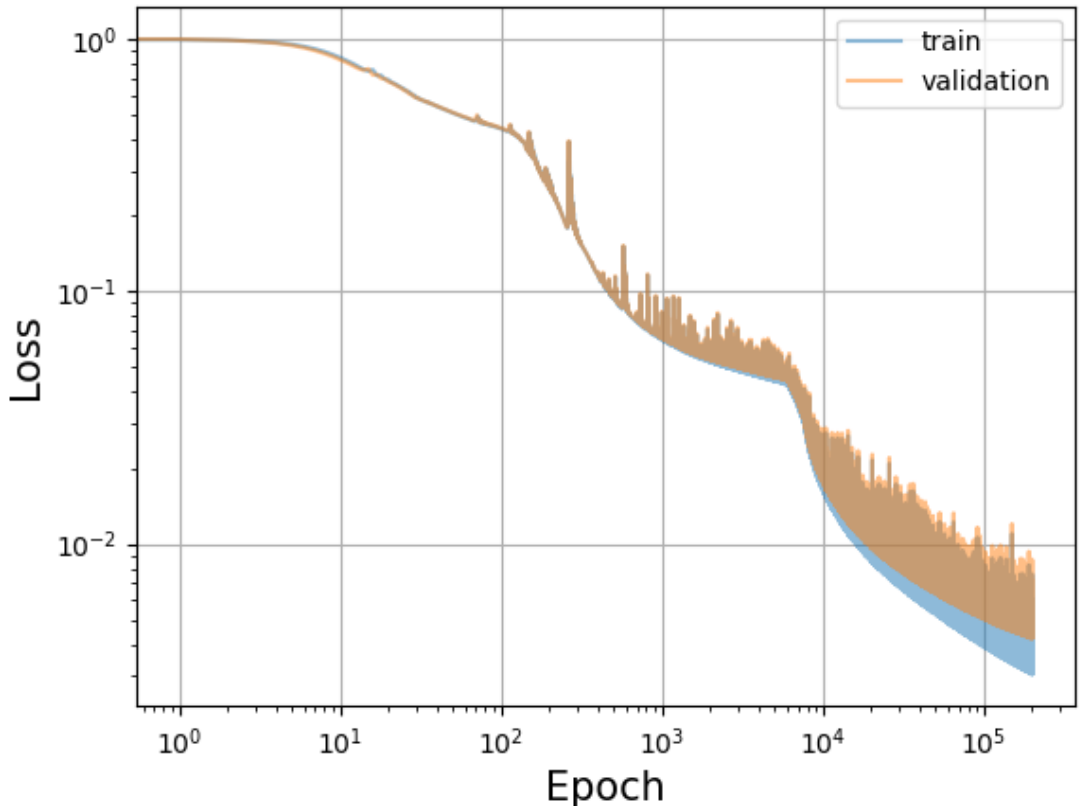

Figure A3: Training and validation loss curve (O2).

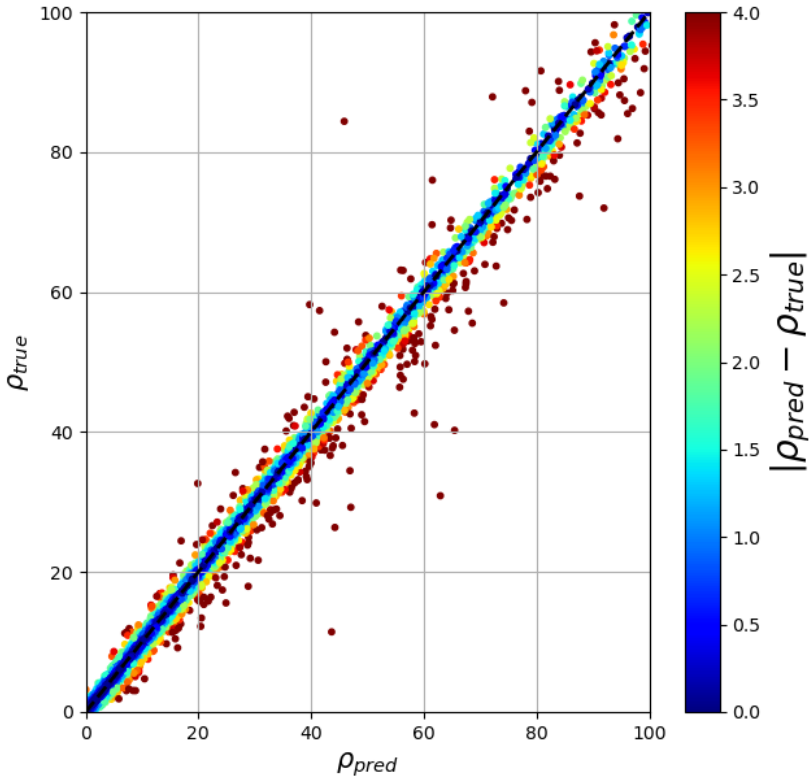

Figure A4: Comparison of predicted vs. true SNR values (H1, O2).

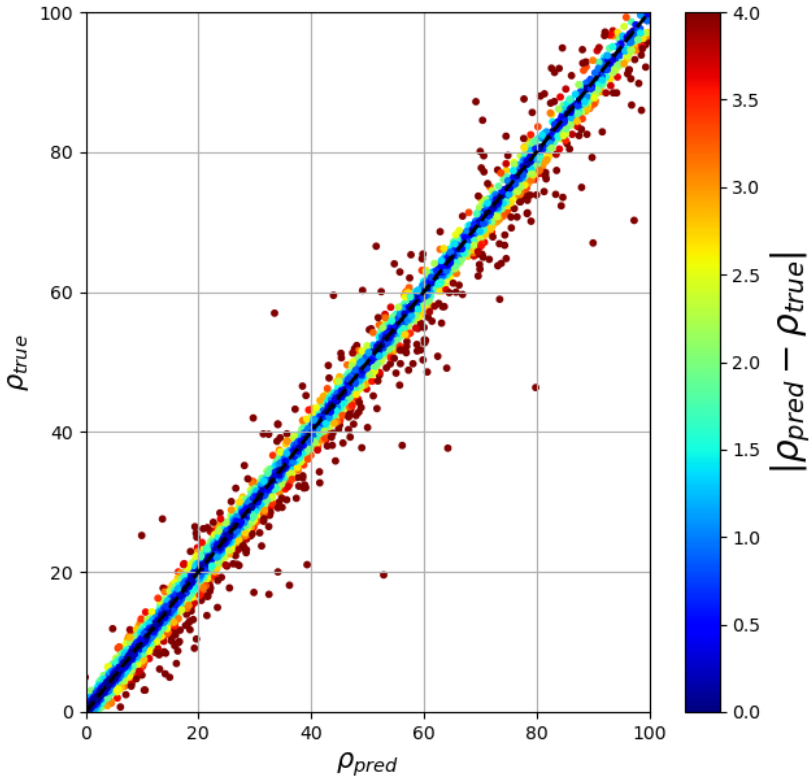

Figure A5: Comparison of predicted vs. true SNR values (L1, O2).

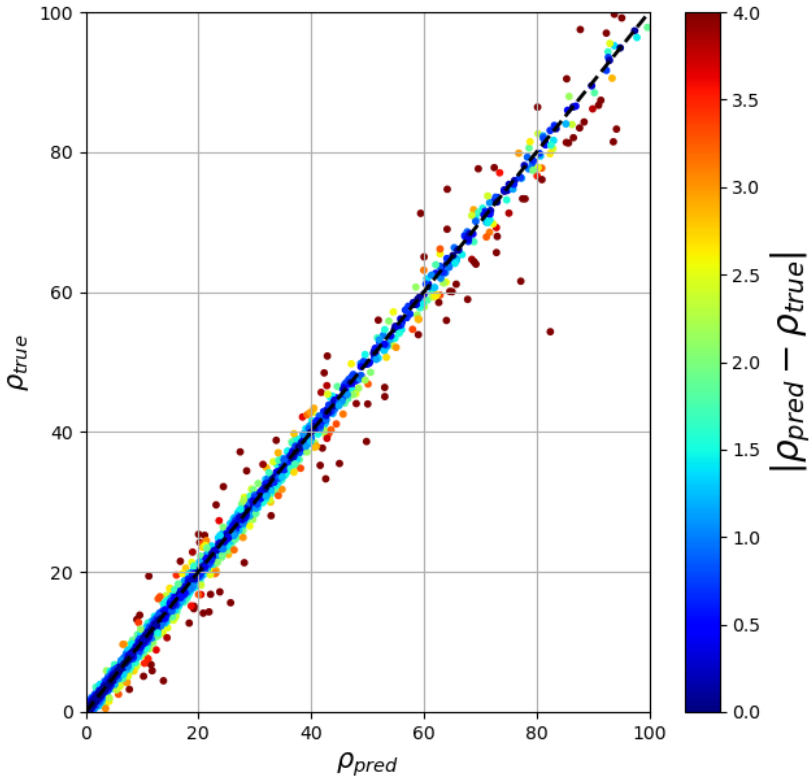

Figure A6: Comparison of predicted vs. true SNR values (V1, O2).

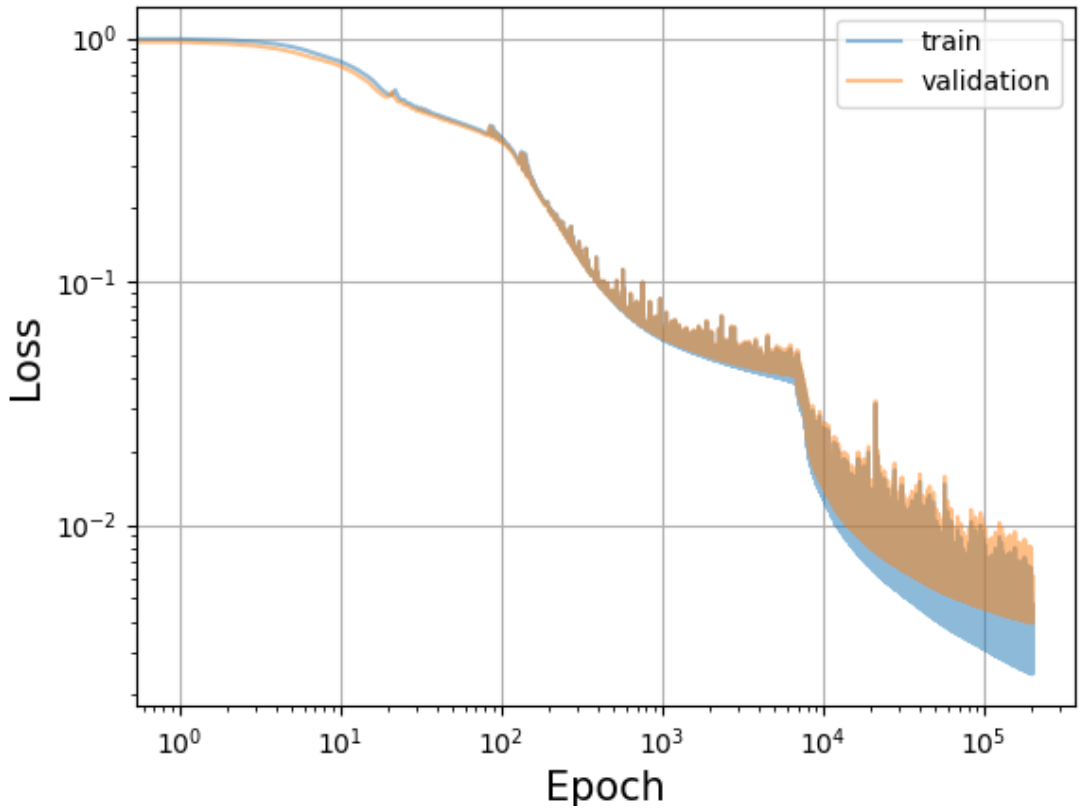

Figure A7: Training and validation loss curve (O1).

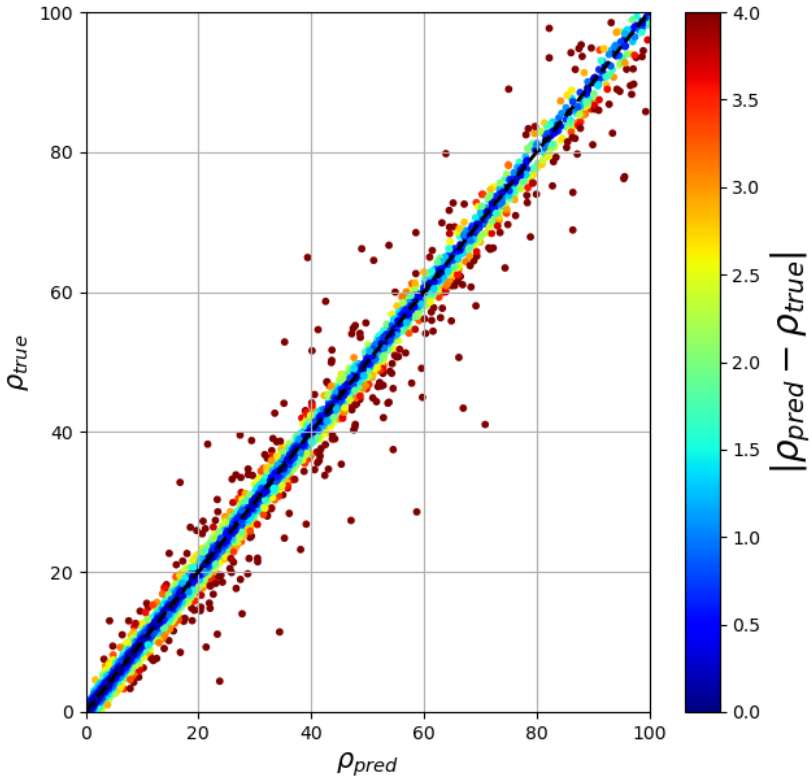

Figure A8: Comparison of predicted vs. true values SNR (H1, O1).

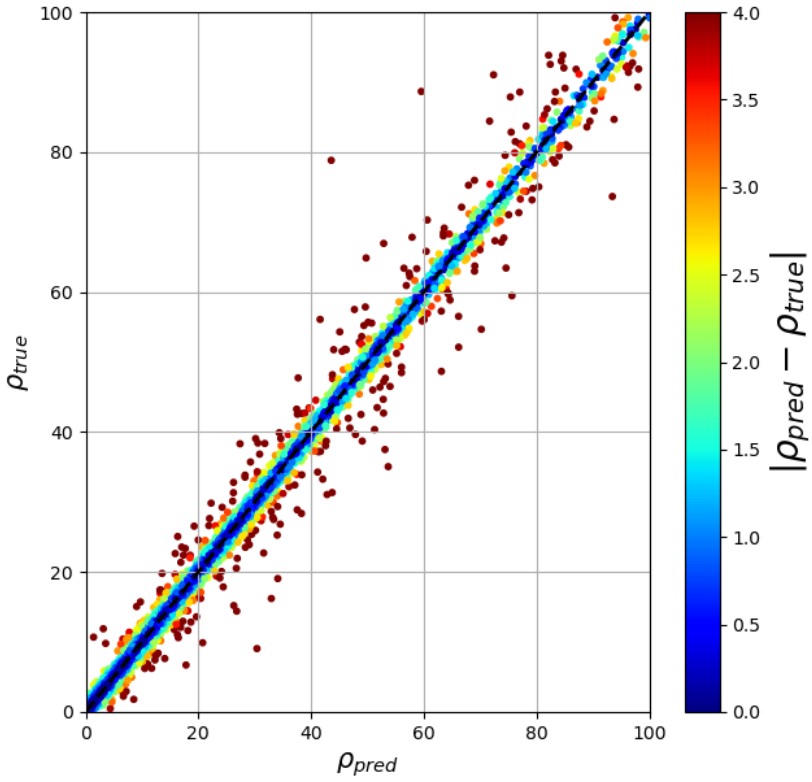

Figure A9: Comparison of predicted vs. true SNR values (L1, O1).

B Appendix B: Comparison between the 5D and the 14D condtional NF

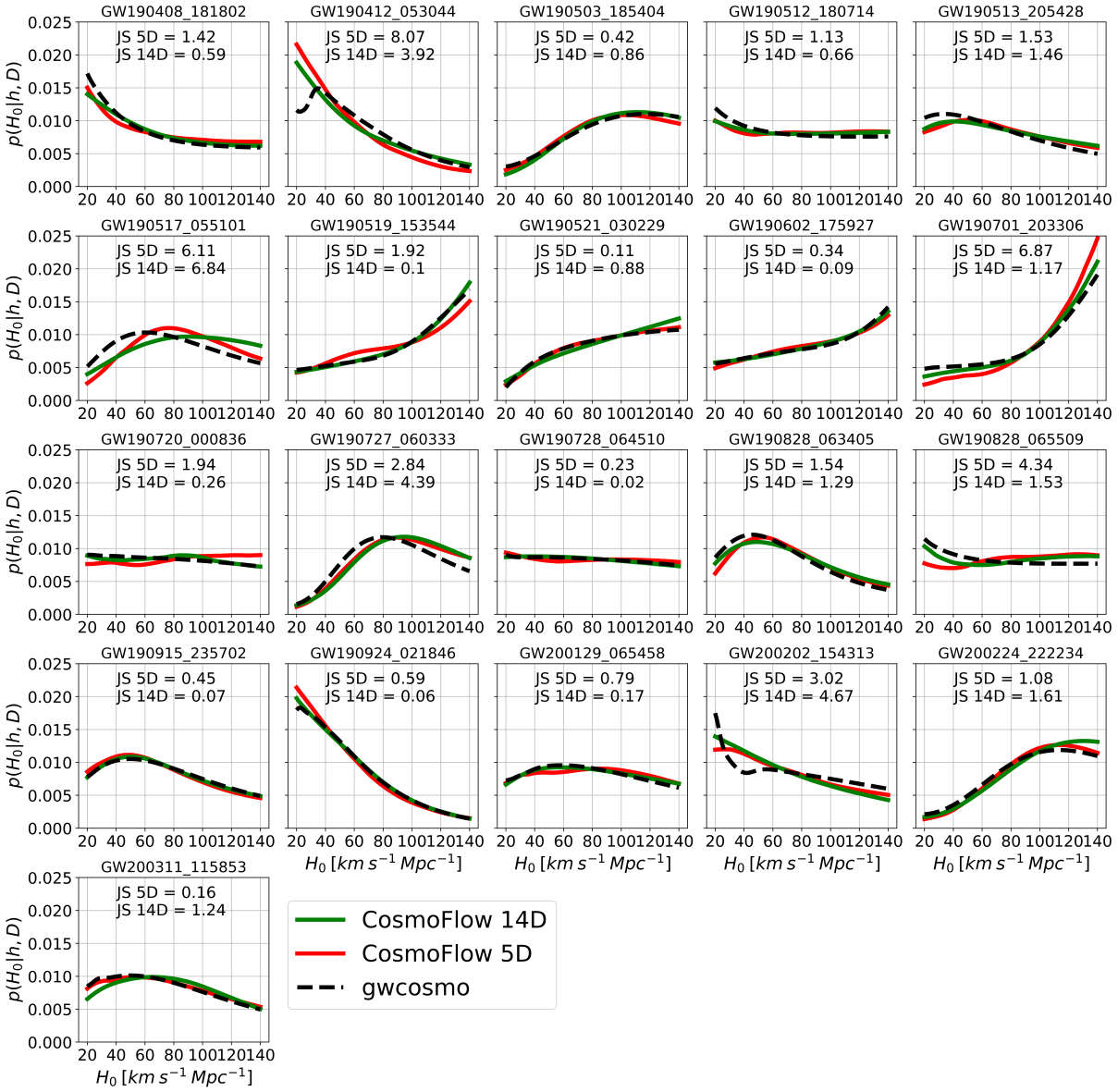

Figure A10: Comparison of Hubble constant H_0 posterior distributions for individual GW events detected during the O3 observing run with HLV detectors. The distributions are shown for `CosmoFlow` with 5D (red solid lines) and 14D (green solid lines) NF, along with `gwcosmo` (black dashed lines). The JS divergence is noted for each event, highlighting the similarity between the 5D and 14D posteriors and providing a measure of the consistency of each model's predictions.

C Appendix C: `nessai` diagnostic plots for nested sampling

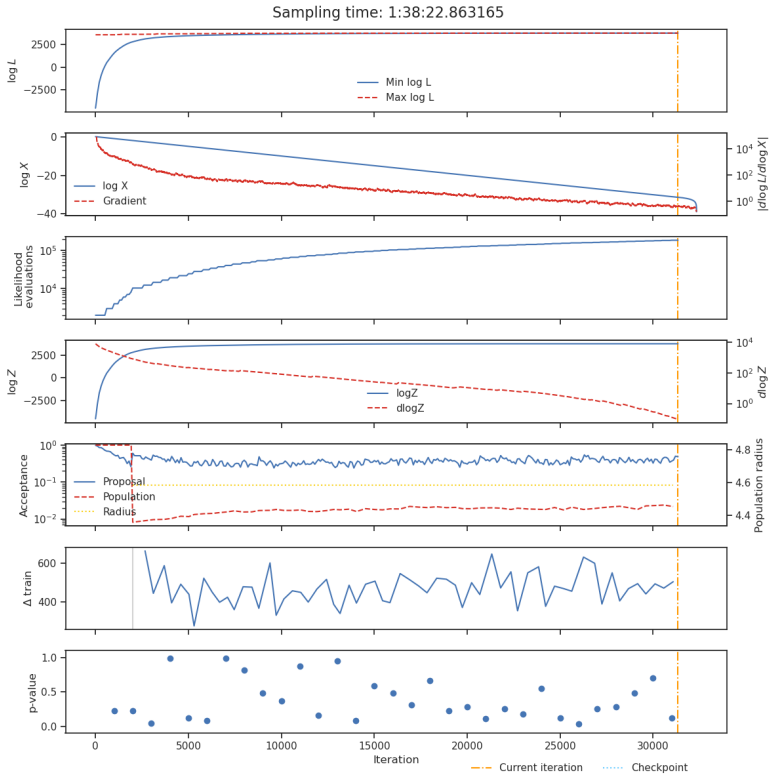

Figure B11: State of the model for 500 test samples, showing log-likelihood, population radius, and acceptance rates during the parameter estimation process for Fig.5.7.

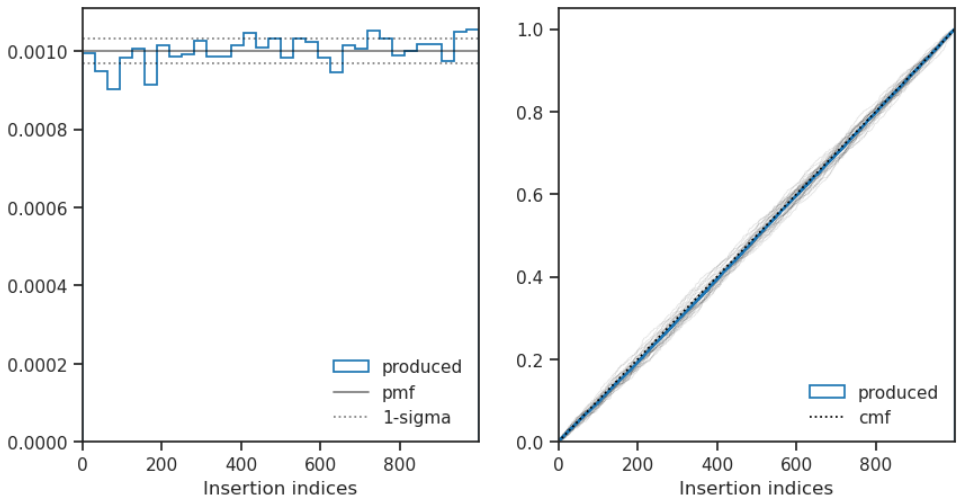

Figure B12: Insertion indices with 500 test samples for Fig.5.7. If the sampler is working properly, the insertion indices should be uniformly distributed between 0 and 1, indicating that the true values are equally likely to be located at any position within the range of posterior samples.

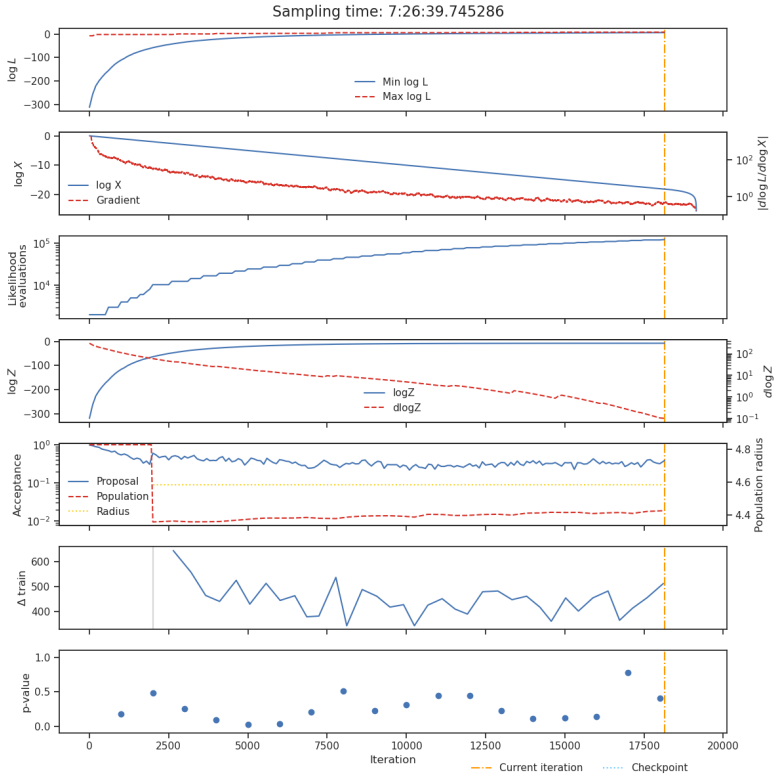

Figure B13: State of the model over iterations, showing log-likelihood, population radius, and acceptance rates for Fig.5.9.

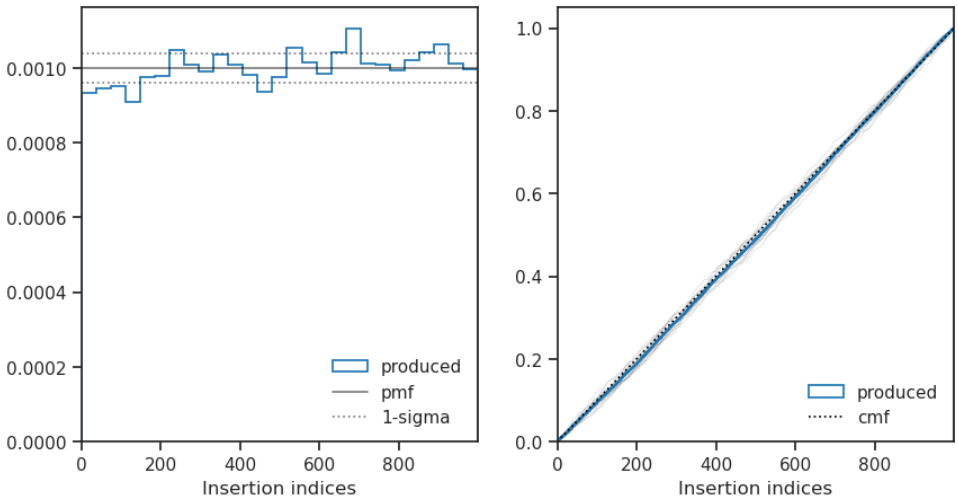

Figure B14: Distribution of insertion indices for Fig.5.9..

Bibliography

[1] David W. Hogg. 'Distance measures in cosmology'. In: *arXiv e-prints*, astro-ph/9905116 (May 1999), astro–ph/9905116. arXiv: astro-ph/9905116 [astro-ph] (cit. on pp. 1, 2, 54).

[2] B.Ryden. *Introduction to cosmology*. Cambridge University Press, 2017 (cit. on pp. 2, 4).

[3] Eleonora Di Valentino, Alessandro Melchiorri and Joseph Silk. 'Planck evidence for a closed Universe and a possible crisis for cosmology'. In: *Nature Astronomy* 4.2 (Nov. 2019), pp. 196–203. ISSN: 2397-3366. DOI: 10.1038/s41550-019-0906-9. URL: http://dx.doi.org/10.1038/s41550-019-0906-9 (cit. on p. 3).

[4] Planck Collaboration. 'Planck 2018 results. VI. Cosmological parameters (Corrigendum)'. In: *aap* 652, C4 (Aug. 2021), p. C4. DOI: 10.1051/0004-6361/201833910e (cit. on pp. 3, 172).

[5] N. Aghanim et al. 'Planck 2018 results'. In: *Astronomy & Astrophysics* 641 (Sept. 2020), A6. ISSN: 1432-0746. DOI: 10.1051/0004-6361/201833910. URL: http://dx.doi.org/10.1051/0004-6361/201833910 (cit. on pp. 3, 140, 177, 179).

[6] Adam G. Riess et al. 'Large Magellanic Cloud Cepheid Standards Provide a 1% Foundation for the Determination of the Hubble Constant and Stronger Evidence for Physics beyond ΛCDM'. In: *The Astrophysical Journal* 876.1 (May 2019), p. 85. DOI: 10.3847/1538-4357/ab1422. URL: https://dx.doi.org/10.3847/1538-4357/ab1422 (cit. on pp. 4, 177, 179).

[7] Wendy L. Freedman. *Cosmology at at Crossroads: Tension with the Hubble Constant*. 2017. arXiv: 1706.02739 [astro-ph.CO] (cit. on pp. 4, 5).

[8] Vivien Bonvin. *H0 tension: A compilation of measurements of the Hubble constant, and scripts to visualize the Hubble tension*. 2023. URL: github.com/vbonvin/H0_tension (cit. on p. 4).

[9] A. Drlica-Wagner and DES Collaboration. 'Dark Energy Survey Year 1 Results: The Photometric Data Set for Cosmology'. In: *The Astrophysical Journal Supplement Series* 235.2 (Apr. 2018), p. 33. DOI: 10.3847/1538-4365/aab4f5. URL: https://dx.doi.org/10.3847/1538-4365/aab4f5 (cit. on pp. 5, 65, 142).

[10] T. M. C. Abbott et al. 'Dark Energy Survey Year 1 Results: A Precise H0 Estimate from DES Y1, BAO, and D/H Data'. In: *Monthly Notices of the Royal Astronomical Society* 480.3 (July 2018), pp. 3879–3888. DOI: 10.1093/mnras/sty1939. URL: http://dx.doi.org/10.1093/mnras/sty1939 (cit. on p. 5).

[11] Caroline D. Huang et al. 'Hubble Space Telescope Observations of Mira Variables in the SN Ia Host NGC 1559: An Alternative Candle to Measure the Hubble Constant'. In: *The Astrophysical Journal* 889.1 (Jan. 2020), p. 5. ISSN: 1538-4357. DOI: 10.3847/1538-4357/ab5dbd. URL: http://dx.doi.org/10.3847/1538-4357/ab5dbd (cit. on p. 5).

[12] Bernard F. Schutz. 'Determining the Hubble constant from gravitational wave observations'. In: *Nature* 323 (1986), pp. 310–311 (cit. on pp. 5, 23, 44, 179).

[13] M. Maggiore. *Gravitational Waves: Volume 1: Theory and Experiments*. Gravitational Waves. OUP Oxford, 2008. ISBN: 9780198570745. URL: https://books.google.co.uk/books?id=AqVpQgAACAAJ (cit. on pp. 5, 9, 13, 158).

[14] R Gray, C Messenger and J Veitch. 'A pixelated approach to galaxy catalogue incompleteness: improving the dark siren measurement of the Hubble constant'. In: *Monthly Notices of the Royal Astronomical Society* 512.1 (Feb. 2022), pp. 1127–1140. ISSN: 0035-8711. DOI: 10.1093/mnras/stac366. eprint: https://academic.oup.com/mnras/article-pdf/512/1/1127/45303118/stac366.pdf. URL: https://doi.org/10.1093/mnras/stac366 (cit. on pp. 5, 26, 29, 30, 47, 57).

[15] Simone Mastrogiovanni et al. *ICAROGW: A python package for inference of astrophysical population properties of noisy, heterogeneous and incomplete observations*. 2023. arXiv: 2305.17973 [astro-ph.CO] (cit. on pp. 5, 176).

[16] The LIGO Scientific Collaboration, The Virgo Collaboration and The KAGRA Collaboration. *Constraints on the cosmic expansion history from GWTC-3*. 2021. DOI: 10.48550/ARXIV.2111.03604. URL: https://arxiv.org/abs/2111.03604 (cit. on pp. 5, 25, 26, 30, 51, 52, 54, 55, 57, 58, 61, 63, 65–67, 69, 75, 77, 81, 84, 124, 125, 128, 129, 132, 134, 138, 140, 141, 148–151, 169, 177, 179).

[17] A. Einstein. 'Die Grundlage der allgemeinen Relativitätstheorie'. In: *Annalen der Physik* 354.7 (1916), pp. 769–822. DOI: https://doi.org/10.1002/andp.19163540702. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/andp.19163540702. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/andp.19163540702 (cit. on p. 6).

[18] Charles W. Misner, K. S. Thorne and J. A. Wheeler. *Gravitation*. San Francisco: W. H. Freeman, 1973. ISBN: 978-0-7167-0344-0, 978-0-691-17779-3 (cit. on pp. 6–8, 10, 11, 13).

[19] Alexandre Le Tiec and Jerome Novak. 'Theory of Gravitational Waves'. In: (July 2016) (cit. on p. 10).

[20] Kip S. Thorne. 'Multipole expansions of gravitational radiation'. In: *Rev. Mod. Phys.* 52 (2 Apr. 1980), pp. 299–339. DOI: 10.1103/RevModPhys.52.299. URL: https://link.aps.org/doi/10.1103/RevModPhys.52.299 (cit. on p. 12).

[21] Lee Samuel Finn and David F. Chernoff. 'Observing binary inspiral in gravitational radiation: One interferometer'. In: *Phys. Rev. D* 47 (6 Mar. 1993), pp. 2198–2219. DOI: 10.1103/PhysRevD.47.2198. URL: https://link.aps.org/doi/10.1103/PhysRevD.47.2198 (cit. on p. 12).

[22] LIGO Scientific Collaboration. *LIGO's Interferometer*. 2023. URL: https://www.ligo.caltech.edu/page/ligos-ifo (cit. on p. 12).

[23] Craig Cahillane and Georgia Mansell. 'Review of the Advanced LIGO Gravitational Wave Observatories Leading to Observing Run Four'. In: *Galaxies* 10.1 (2022), p. 36. DOI: 10.3390/galaxies10010036. URL: https://www.mdpi.com/2075-4434/10/1/36 (cit. on p. 13).

[24] J. Aasi et al. 'Advanced LIGO'. In: *Class. Quant. Grav.* 32 (2015), p. 074001. DOI: 10.1088/0264-9381/32/7/074001. arXiv: 1411.4547 [gr-qc] (cit. on pp. 14, 15).

[25] Benjamin P Abbott et al. 'A guide to LIGO–Virgo detector noise and extraction of transient gravitational-wave signals'. In: *Class. Quant. Grav.* 37.5 (2020), p. 055002. DOI: 10.1088/1361-6382/ab685e. arXiv: 1908.11170 [gr-qc] (cit. on p. 15).

[26] B. P. Abbott et al. 'Characterization of transient noise in Advanced LIGO relevant to gravitational wave signal GW150914'. In: *Class. Quant. Grav.* 33.13 (2016), p. 134001. DOI: 10.1088/0264-9381/33/13/134001. arXiv: 1602.03844 [gr-qc] (cit. on p. 15).

[27] D. M. Macleod et al. 'GWpy: A Python package for gravitational-wave astrophysics'. In: *SoftwareX* 13 (2021), p. 100657. ISSN: 2352-7110. DOI: 10.1016/j.softx.2021.100657. URL: https://www.sciencedirect.com/science/article/pii/S2352711021000029 (cit. on p. 16).

[28] D.O. North. 'An Analysis of the factors which determine signal/noise discrimination in pulsed-carrier systems'. In: *Proceedings of the IEEE* 51.7 (1963), pp. 1016–1027. DOI: 10.1109/PROC.1963.2383 (cit. on pp. 15, 17).

[29] Samantha A Usman et al. 'The PyCBC search for gravitational waves from compact binary coalescence'. In: *Classical and Quantum Gravity* 33.21 (Oct. 2016), p. 215004. ISSN: 1361-6382. DOI: 10.1088/0264-9381/33/21/215004. URL: http://dx.doi.org/10.1088/0264-9381/33/21/215004 (cit. on p. 16).

[30] B. P. Abbott, R. Abbott and T. D. Abbott. 'Observation of Gravitational Waves from a Binary Black Hole Merger'. In: *Phys. Rev. Lett.* 116 (6 Feb. 2016), p. 061102. DOI: 10.1103/PhysRevLett.116.061102. URL: https://link.aps.org/doi/10.1103/PhysRevLett.116.061102 (cit. on pp. 17, 45, 126).

[31] "Andrew Gelman. *"Bayesian Data Analysis Third edition"*. "Colombia University Press", 2020. URL: http://www.stat.columbia.edu/~gelman/book/BDA3.pdf (cit. on pp. 18, 19).

[32] B. P. Abbott et al. 'Parameter estimation for compact binary coalescence signals with the first observing run of Advanced LIGO'. In: *Physical Review D* 94.6 (2016), p. 062001 (cit. on pp. 20, 23).

[33] Stephen Fairhurst et al. 'Simple parameter estimation using observable features of gravitational-wave signals'. In: *Phys. Rev. D* 108 (8 Oct. 2023), p. 082006. DOI: 10.1103/PhysRevD.108.082006. URL: https://link.aps.org/doi/10.1103/PhysRevD.108.082006 (cit. on p. 20).

[34] W. K. Hastings. 'Monte Carlo sampling methods using Markov chains and their applications'. In: *Biometrika* 57.1 (Apr. 1970), pp. 97–109. ISSN: 0006-3444. DOI: 10.1093/biomet/57.1.97. eprint: https://academic.oup.com/biomet/article-pdf/57/1/97/23940249/57-1-97.pdf. URL: https://doi.org/10.1093/biomet/57.1.97 (cit. on p. 21).

[35] Gregory Ashton and Colm Talbot. 'B ilby-MCMC: an MCMC sampler for gravitational-wave inference'. In: *Mon. Not. Roy. Astron. Soc.* 507.2 (2021), pp. 2037–2051. DOI: 10.1093/mnras/stab2236. arXiv: 2106.08730 [gr-qc] (cit. on p. 21).

[36] John Skilling. 'Nested Sampling'. In: *AIP Conference Proceedings* 735.1 (Nov. 2004), pp. 395–405. ISSN: 0094-243X. DOI: 10.1063/1.1835238. eprint: https://pubs.aip.org/aip/acp/article-pdf/735/1/395/11702789/395_1_online.pdf. URL: https://doi.org/10.1063/1.1835238 (cit. on pp. 21, 22, 162).

[37] Michael J. Williams, John Veitch and Chris Messenger. 'Nested sampling with normalizing flows for gravitational-wave inference'. In: *Physical Review D* 103.10 (May 2021). DOI: 10.1103/physrevd.103.103006. URL: https://doi.org/10.1103%2Fphysrevd.103.103006 (cit. on pp. 22, 37, 128, 162, 163).

[38] Michael J. Williams, John Veitch and Chris Messenger. *Importance nested sampling with normalising flows*. 2023. arXiv: 2302.08526 [astro-ph.IM] (cit. on pp. 22, 37, 151, 162, 163, 177).

[39] Hunter Gabbard et al. 'Bayesian parameter estimation using conditional variational autoencoders for gravitational-wave astronomy'. In: *Nature Physics* 18.1 (Dec. 2021), pp. 112–117. ISSN: 1745-2481. DOI: 10.1038/s41567-021-01425-7. URL: http://dx.doi.org/10.1038/s41567-021-01425-7 (cit. on p. 23).

[40] Maximilian Dax et al. 'Real-Time Gravitational Wave Science with Neural Posterior Estimation'. In: *Physical Review Letters* 127.24 (Dec. 2021). DOI: 10.1103/physrevlett.127.241103. URL: https://doi.org/10.1103%2Fphysrevlett.127.241103 (cit. on pp. 23, 37).

[41] David F. Chernoff and Lee S. Finn. 'Gravitational radiation, inspiraling binaries, and cosmology'. In: *The Astrophysical Journal* 411 (July 1993), p. L5. ISSN: 1538-4357. DOI: 10.1086/186898. URL: http://dx.doi.org/10.1086/186898 (cit. on p. 24).

[42] David Camarena and Valerio Marra. 'Local determination of the Hubble constant and the deceleration parameter'. In: *Physical Review Research* 2.1 (Jan. 2020). ISSN: 2643-1564. DOI: 10.1103/physrevresearch.2.013028. URL: http://dx.doi.org/10.1103/PhysRevResearch.2.013028 (cit. on p. 24).

[43] Walter Del Pozzo. 'Inference of cosmological parameters from gravitational waves: Applications to second generation interferometers'. In: *Phys. Rev. D* 86 (4 Aug. 2012), p. 043011. DOI: 10.1103/PhysRevD.86.043011. URL: https://link.aps.org/doi/10.1103/PhysRevD.86.043011 (cit. on p. 24).

[44] Samaya Nissanke et al. *Determining the Hubble constant from gravitational wave observations of merging compact binaries*. 2013. arXiv: 1307.2638 [astro-ph.CO]. URL: https://arxiv.org/abs/1307.2638 (cit. on p. 24).

[45] C. Messenger and J. Read. 'Measuring a Cosmological Distance-Redshift Relationship Using Only Gravitational Wave Observations of Binary Neutron Star Coalescences'. In: *Physical Review Letters* 108.9 (Feb. 2012). ISSN: 1079-7114. DOI: 10.1103/physrevlett.108.091101. URL: http://dx.doi.org/10.1103/PhysRevLett.108.091101 (cit. on p. 25).

[46] Stephen R. Taylor, Jonathan R. Gair and Ilya Mandel. 'Cosmology using advanced gravitational-wave detectors alone'. In: *Physical Review D* 85.2 (Jan. 2012). ISSN: 1550-2368. DOI: 10.1103/physrevd.85.023535. URL: http://dx.doi.org/10.1103/PhysRevD.85.023535 (cit. on p. 25).

[47] R. Abbott et al. 'GWTC-2: Compact Binary Coalescences Observed by LIGO and Virgo During the First Half of the Third Observing Run'. In: *Phys. Rev. X* 11 (2021), p. 021053. DOI: 10.1103/PhysRevX.11.021053. arXiv: 2010.14527 [gr-qc] (cit. on pp. 25, 54).

[48] LIGO Scientific Collaboration, Virgo Collaboration and KAGRA Collaboration. *GWTC-3: Compact Binary Coalescences Observed by LIGO and Virgo During the Second Part of the Third Observing Run — Parameter estimation data release*. 2023. DOI: 10.5281/zenodo.8177023. URL: https://doi.org/10.5281/zenodo.8177023 (cit. on pp. 26, 54, 127).

[49] B. P. Abbott et al. 'GW170817: Observation of Gravitational Waves from a Binary Neutron Star Inspiral'. In: *Phys. Rev. Lett.* 119.16 (2017), p. 161101. DOI: 10.1103/PhysRevLett.119.161101. arXiv: 1710.05832 [gr-qc] (cit. on pp. 26, 45).

[50] R. Abbott et al. 'GW190814: Gravitational Waves from the Coalescence of a 23 Solar Mass Black Hole with a 2.6 Solar Mass Compact Object'. In: *Astrophys. J. Lett.* 896.2 (2020), p. L44. DOI: 10.3847/2041-8213/ab960f. arXiv: 2006.12611 [astro-ph.HE] (cit. on pp. 26, 141).

[51] Rachel Gray et al. 'Cosmological inference using gravitational wave standard sirens: A mock data analysis'. In: *Phys. Rev. D* 101 (12 June 2020), p. 122001. DOI: 10.1103/PhysRevD.101.122001. URL: https://link.aps.org/doi/10.1103/PhysRevD.101.122001 (cit. on pp. 26–29, 47, 81, 125, 179).

[52] Rachel Gray et al. *Joint cosmological and gravitational-wave population inference using dark sirens and galaxy catalogues*. 2023. arXiv: 2308.02281 [astro-ph.CO] (cit. on pp. 26, 47, 150, 151, 153, 167).

[53] Maya Fishbach, Daniel E. Holz and Will M. Farr. 'Does the Black Hole Merger Rate Evolve with Redshift?' In: *The Astrophysical Journal Letters* 863.2 (Aug. 2018), p. L41. ISSN: 2041-8213. DOI: 10.3847/2041-8213/aad800. URL: http://dx.doi.org/10.3847/2041-8213/aad800 (cit. on p. 27).

[54] Federico Stachurski, Christopher Messenger and Martin Hendry. *Cosmological Inference using Gravitational Waves and Normalising Flows*. 2023. arXiv: 2310.13405 [gr-qc] (cit. on pp. 31, 45, 124, 131).

[55] Gio Wiederhold and John McCarthy. 'Arthur Samuel: Pioneer in Machine Learning'. In: *IBM Journal of Research and Development* 36 (June 1992), pp. 329–331. DOI: 10.1147/rd.363.0329 (cit. on p. 31).

[56] Ethem Alpaydin. *Introduction to machine learning*. The MIT Press, 2020 (cit. on pp. 32–34).

[57] Hassan et al. 'Assessment of Artificial Neural Network for Bathymetry Estimation Using High Resolution Satellite Imagery in Shallow Lakes: Case Study El Burullus Lake.' In: *International Water Technology Journal* 5 (Dec. 2015) (cit. on p. 32).

[58] Seppo Linnainmaa. 'Taylor expansion of the accumulated rounding error'. In: *BIT Numerical Mathematics* 16 (1976), pp. 146–160. DOI: 10.1007/BF01931367. URL: https://doi.org/10.1007/BF01931367 (cit. on p. 33).

[59] David E. Rumelhart, Geoffrey E. Hinton and Ronald J. Williams. 'Learning Representations by Back-propagating Errors'. In: *Nature* 323 (1986), pp. 533–536. DOI: 10.1038/323533a0. URL: https://doi.org/10.1038/323533a0 (cit. on p. 33).

[60] Sebastian Ruder. *An overview of gradient descent optimization algorithms*. 2017. arXiv: 1609.04747 [cs.LG] (cit. on p. 34).

[61] Herbert Robbins and Sutton Monro. 'A Stochastic Approximation Method'. In: *The Annals of Mathematical Statistics* 22.3 (1951), pp. 400–407. DOI: 10.1214/aoms/1177729586 (cit. on p. 34).

[62] Alexander J. Smola and S.V.N. Vishwanathan. *Introduction to Machine Learning*. Cambridge, United Kingdom: Cambridge University Press, 2008. ISBN: 0521825830. URL: svn://smola@repos.stat.purdue.edu/thebook/trunk/Book/thebook.tex (cit. on p. 35).

[63] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG] (cit. on pp. 35, 106, 107).

[64] Richard S. Sutton. 'Learning from Delayed Rewards'. PhD Dissertation. University of Massachusetts, Amherst, 1984 (cit. on p. 36).

[65] Richard S Sutton and Andrew G Barto. 'Reinforcement learning: An introduction'. In: *Robotica* 17.2 (1999), pp. 229–235 (cit. on p. 36).

[66] Alex Lamb. *A Brief Introduction to Generative Models*. 2021. arXiv: 2103.00265 [cs.LG]. URL: https://arxiv.org/abs/2103.00265 (cit. on p. 36).

[67] Ian J. Goodfellow et al. *Generative Adversarial Networks*. 2014. arXiv: 1406.2661 [stat.ML] (cit. on p. 36).

[68] Diederik P. Kingma and Max Welling. 'An Introduction to Variational Autoencoders'. In: *Foundations and Trends® in Machine Learning* 12.4 (2019), pp. 307–392. ISSN: 1935-8245. DOI: 10.1561/2200000056. URL: http://dx.doi.org/10.1561/2200000056 (cit. on p. 36).

[69] I. Kobyzev, S. Prince and M. Brubaker. 'Normalizing Flows: An Introduction and Review of Current Methods'. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020), pp. 1–1. DOI: 10.1109/TPAMI.2020.2992934 (cit. on pp. 36, 37, 40, 103).

[70] Danilo Jimenez Rezende and Shakir Mohamed. *Variational Inference with Normalizing Flows*. 2016. arXiv: 1505.05770 [stat.ML] (cit. on pp. 37–39).

[71] George Papamakarios et al. 'Normalizing Flows for Probabilistic Modeling and Inference'. In: *Journal of Machine Learning Research* (2019). DOI: 10.48550/ARXIV.1912.02762. URL: https://arxiv.org/abs/1912.02762 (cit. on pp. 37, 103).

[72] Solomon Kullback and Richard A. Leibler. 'On Information and Sufficiency'. In: *The Annals of Mathematical Statistics* 22.1 (1951), pp. 79–86. DOI: 10.1214/aoms/1177729694 (cit. on pp. 38, 39).

[73] Laurent Dinh, David Krueger and Yoshua Bengio. *NICE: Non-linear Independent Components Estimation*. 2015. arXiv: 1410.8516 [cs.LG] (cit. on p. 40).

[74] Diederik P. Kingma et al. *Improving Variational Inference with Inverse Autoregressive Flow*. 2017. arXiv: 1606.04934 [cs.LG] (cit. on p. 40).

[75] Conor Durkan et al. *Cubic-Spline Flows*. 2019. arXiv: 1906.02145 [stat.ML] (cit. on pp. 40, 41).

[76] Laurent Dinh, Jascha Sohl-Dickstein and Samy Bengio. *Density estimation using Real NVP*. 2017. arXiv: 1605.08803 [cs.LG] (cit. on pp. 40, 41).

[77] Conor Durkan et al. *Neural Spline Flows*. 2019. arXiv: 1906.04032 [stat.ML] (cit. on pp. 41–43, 107, 114).

[78] J. A. Gregory and R. Delbourgo. 'Piecewise Rational Quadratic Interpolation to Monotonic Data'. In: *IMA Journal of Numerical Analysis* 2.2 (Apr. 1982), pp. 123–130. ISSN: 0272-4979 (cit. on p. 41).

[79] Jarno Lintusaari et al. 'Fundamentals and Recent Developments in Approximate Bayesian Computation'. In: *Systematic Biology* 66.1 (Sept. 2016), e66–e82. ISSN: 1063-5157. DOI: 10.1093/sysbio/syw077. eprint: https://academic.oup.com/sysbio/article-pdf/66/1/e66/24194739/syw077.pdf. URL: https://doi.org/10.1093/sysbio/syw077 (cit. on p. 46).

[80] Kyle Cranmer, Johann Brehmer and Gilles Louppe. 'The frontier of simulation-based inference'. In: *Proceedings of the National Academy of Sciences* 117.48 (May 2020), pp. 30055–30062. ISSN: 1091-6490. DOI: 10.1073/pnas.1912789117. URL: http://dx.doi.org/10.1073/pnas.1912789117 (cit. on p. 46).

[81] G Dálya et al. 'GLADE+: an extended galaxy catalogue for multimessenger searches with advanced gravitational-wave detectors'. In: *Monthly Notices of the Royal Astronomical Society* 514.1 (May 2022), pp. 1403–1411. ISSN: 1365-2966. DOI: 10.1093/mnras/stac1443. URL: http://dx.doi.org/10.1093/mnras/stac1443 (cit. on pp. 47, 50, 63).

[82] Federico Stachurski. *CosmoFlow: Git Repository*. Accessed: 2024-06-12. 2024. URL: https://git.ligo.org/federico.stachurski/cosmoflow (cit. on p. 47).

[83] LSCSoft. *gwcosmo: Git Repository*. Accessed: 2024-06-12. 2024. URL: https://git.ligo.org/lscsoft/gwcosmo/-/tree/master (cit. on pp. 47, 83).

[84] C. Messenger and J. Veitch. 'Avoiding selection bias in gravitational wave astronomy'. In: *New Journal of Physics* 15.5, 053027 (May 2013), p. 053027. DOI: 10.1088/1367-2630/15/5/053027. arXiv: 1206.3461 [astro-ph.IM] (cit. on p. 47).

[85] Rachel Gray et al. 'Cosmological inference using gravitational wave standard sirens: A mock data analysis'. In: *Physical Review D* 101.12 (June 2020). ISSN: 2470-0029. DOI: 10.1103/physrevd.101.122001. URL: http://dx.doi.org/10.1103/PhysRevD.101.122001 (cit. on p. 47).

[86] LIGO-Virgo Collaboration. 'GWTC-2: Compact Binary Coalescences Observed by LIGO and Virgo during the First Half of the Third Observing Run'. In: *Physical Review X* 11.2 (June 2021). ISSN: 2160-3308. DOI: 10.1103/physrevx.11.021053. URL: http://dx.doi.org/10.1103/PhysRevX.11.021053 (cit. on pp. 49, 51, 127, 141).

[87] The LIGO Scientific Collaboration, The Virgo Collaboration and The KAGRA Collaboration. *The population of merging compact binaries inferred using gravitational waves through GWTC-3*. 2021. DOI: 10.48550/ARXIV.2111.03634. URL: https://arxiv.org/abs/2111.03634 (cit. on pp. 49, 61, 63, 74, 75, 77).

[88] R. Abbott, T. D. Abbott and Acernese. 'GWTC-3: Compact Binary Coalescences Observed by LIGO and Virgo during the Second Part of the Third Observing Run'. In: *Physical Review X* 13.4 (Dec. 2023). ISSN: 2160-3308. DOI: 10.1103/physrevx.13.041039. URL: http://dx.doi.org/10.1103/PhysRevX.13.041039 (cit. on pp. 51, 127).

[89] Curt Cutler and Éanna E. Flanagan. 'Gravitational waves from merging compact binaries: How accurately can one extract the binary's parameters from the inspiral waveform?' In: *Phys. Rev. D* 49 (6 Mar. 1994), pp. 2658–2697. DOI: 10.1103/PhysRevD.49.2658. URL: https://link.aps.org/doi/10.1103/PhysRevD.49.2658 (cit. on p. 51).

[90] Pauli Virtanen et al. 'SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python'. In: *Nature Methods* 17 (2020), pp. 261–272. DOI: 10.1038/s41592-019-0686-2 (cit. on pp. 52, 70).

[91] B. P. Abbott et al. 'GWTC-1: A Gravitational-Wave Transient Catalog of Compact Binary Mergers Observed by LIGO and Virgo during the First and Second Observing Runs'. In: *Phys. Rev. X* 9.3 (2019), p. 031040. DOI: 10.1103/PhysRevX.9.031040. arXiv: 1811.12907 [astro-ph.HE] (cit. on p. 54).

[92] R. Gray. 'Gravitational wave cosmology: measuring the Hubble constant with dark standard sirens'. In: *https://theses.gla.ac.uk/82438/* (2021) (cit. on pp. 54, 83).

[93] Tom Callister et al. 'Shouts and Murmurs: Combining Individual Gravitational-wave Sources with the Stochastic Background to Measure the History of Binary Black Hole Mergers'. In: *The Astrophysical Journal Letters* 896.2 (June 2020), p. L32. ISSN: 2041-8213. DOI: 10.3847/2041-8213/ab9743. URL: http://dx.doi.org/10.3847/2041-8213/ab9743 (cit. on pp. 55, 62).

[94] Sezgin Secil and Metin Ozkan. 'A collision-free path planning method for industrial robot manipulators considering safe human–robot interaction'. In: *Intelligent Service Robotics* 16 (June 2023), pp. 1–37. DOI: 10.1007/s11370-023-00465-7 (cit. on p. 56).

[95] P. Schechter. 'An analytic expression for the luminosity function for galaxies.' In: 203 (Jan. 1976), pp. 297–306. DOI: 10.1086/154079 (cit. on pp. 56, 66).

[96] C. S. Kochanek et al. 'The K-Band Galaxy Luminosity Function* **'. In: *The Astrophysical Journal* 560.2 (Oct. 2001), p. 566. DOI: 10.1086/322488. URL: https://dx.doi.org/10.1086/322488 (cit. on p. 57).

[97] Andrea Zonca et al. 'healpy: equal area pixelization and spherical harmonics transforms for data on the sphere in Python'. In: *Journal of Open Source Software* 4.35 (Mar. 2019), p. 1298. DOI: 10.21105/joss.01298. URL: https://doi.org/10.21105/joss.01298 (cit. on p. 57).

[98] K. M. Górski et al. 'HEALPix: A Framework for High-Resolution Discretization and Fast Analysis of Data Distributed on the Sphere'. In: 622 (Apr. 2005), pp. 759–771. DOI: 10.1086/427976. eprint: arXiv:astro-ph/0409513 (cit. on p. 57).

[99] The LIGO Scientific Collaboration, the Virgo Collaboration and the KAGRA Collaboration. *GWTC-3: Compact Binary Coalescences Observed by LIGO and Virgo During the Second Part of the Third Observing Run*. 2021. arXiv: 2111.03606 [gr-qc] (cit. on pp. 59, 74–76, 102, 148, 150, 177).

[100] Gregory Ashton et al. 'Bilby: A User-friendly Bayesian Inference Library for Gravitational-wave Astronomy'. In: *The Astrophysical Journal Supplement Series* 241.2 (Apr. 2019), p. 27. DOI: 10.3847/1538-4365/ab06fc. URL: https://doi.org/10.3847%2F1538-4365%2Fab06fc (cit. on pp. 59, 84, 117).

[101] Piero Madau and Mark Dickinson. 'Cosmic Star-Formation History'. In: *Annual Review of Astronomy and Astrophysics* 52.1 (2014), pp. 415–486. DOI: 10.1146/annurev-astro-081811-125615. eprint: https://doi.org/10.1146/annurev-astro-081811-125615. URL: https://doi.org/10.1146/annurev-astro-081811-125615 (cit. on pp. 62, 63).

[102] G Dálya et al. 'GLADE: A galaxy catalogue for multimessenger searches in the advanced gravitational-wave detector era'. In: *Monthly Notices of the Royal Astronomical Society* 479.2 (June 2018), pp. 2374–2381. ISSN: 0035-8711. DOI: 10.1093/mnras/sty1703. eprint: https://academic.oup.com/mnras/article-pdf/479/2/2374/25145227/sty1703.pdf. URL: https://doi.org/10.1093/mnras/sty1703 (cit. on p. 63).

[103] Darren J White, E J Daw and V S Dhillon. 'A list of galaxies for gravitational wave searches'. In: *Classical and Quantum Gravity* 28.8 (Mar. 2011), p. 085016. DOI: 10.1088/0264-9381/28/8/085016. URL: https://dx.doi.org/10.1088/0264-9381/28/8/085016 (cit. on p. 63).

[104] Maciej Bilicki et al. 'TWO MICRON ALL SKY SURVEY PHOTOMETRIC REDSHIFT CATALOG: A COMPREHENSIVE THREE-DIMENSIONAL CENSUS OF THE WHOLE SKY'. In: *The Astrophysical Journal Supplement Series* 210.1 (Dec. 2013), p. 9. DOI: 10.1088/0067-0049/210/1/9. URL: https://dx.doi.org/10.1088/0067-0049/210/1/9 (cit. on p. 63).

[105] T. H. Jarrett et al. '2MASS Extended Source Catalog: Overview and Algorithms'. In: 119.5 (May 2000), pp. 2498–2531. DOI: 10.1086/301330. arXiv: astro-ph/0004318 [astro-ph] (cit. on p. 63).

[106] Dmitry Makarov et al. 'HyperLEDA. III. The catalogue of extragalactic distances'. In: 570, A13 (Oct. 2014), A13. DOI: 10.1051/0004-6361/201423496. arXiv: 1408.3476 [astro-ph.GA] (cit. on p. 63).

[107] Maciej Bilicki et al. 'WISE × SuperCOSMOS PHOTOMETRIC REDSHIFT CATALOG: 20 MILLION GALAXIES OVER 3π STERADIANS'. In: *The Astrophysical Journal Supplement Series* 225.1 (July 2016), p. 5. DOI: 10.3847/0067-0049/225/1/5. URL: https://dx.doi.org/10.3847/0067-0049/225/1/5 (cit. on p. 63).

[108] Maciej Bilicki et al. 'WISE x SuperCOSMOS Photometric Redshift Catalog: 20 Million Galaxies over 3/pi Steradians'. In: 225.1, 5 (July 2016), p. 5. DOI: 10.3847/0067-0049/225/1/5. arXiv: 1607.01182 [astro-ph.CO] (cit. on p. 63).

[109] Brad W. Lyke et al. 'The Sloan Digital Sky Survey Quasar Catalog: Sixteenth Data Release'. In: 250.1, 8 (Sept. 2020), p. 8. DOI: 10.3847/1538-4365/aba623. arXiv: 2007.09001 [astro-ph.GA] (cit. on p. 63).

[110] Russell Johnston, Luís Teodoro and Martin Hendry. 'Completeness – I. Revisited, reviewed and revived'. In: *Monthly Notices of the Royal Astronomical Society* 376.4 (Mar. 2007), pp. 1757–1766. ISSN: 1365-2966. DOI: 10.1111/j.1365-2966.2007.11559.x. URL: http://dx.doi.org/10.1111/j.1365-2966.2007.11559.x (cit. on p. 64).

[111] B. P. Abbott et al. 'A Gravitational-wave Measurement of the Hubble Constant Following the Second Observing Run of Advanced LIGO and Virgo'. In: *The Astrophysical Journal* 909.2 (Mar. 2021), p. 218. DOI: 10.3847/1538-4357/abdcb7. URL: https://dx.doi.org/10.3847/1538-4357/abdcb7 (cit. on p. 64).

[112] Hu Zou et al. 'Photometric Redshifts and Stellar Masses for Galaxies from the DESI Legacy Imaging Surveys'. In: *The Astrophysical Journal Supplement Series* 242.1 (May 2019), p. 8. DOI: 10.3847/1538-4365/ab1847. URL: https://dx.doi.org/10.3847/1538-4365/ab1847 (cit. on p. 65).

[113] Adrian A. Collister and Ofer Lahav. 'ANNz: Estimating Photometric Redshifts Using Artificial Neural Networks'. In: *Publications of the Astronomical Society of the Pacific* 116.818 (Mar. 2004), p. 345. DOI: 10.1086/383254. URL: https://dx.doi.org/10.1086/383254 (cit. on p. 65).

[114] M. Bilicki et al. 'WISE x SuperCOSMOS photometric redshift catalog: 20 million galaxies over 3pi steradians'. In: *Astrophys. J. Suppl.* 225 (2016), p. 5. DOI: 10.3847/0067-0049/225/1/5. arXiv: 1607.01182 [astro-ph.CO] (cit. on p. 65).

[115] Mukherjee, Suvodip et al. 'Velocity correction for Hubble constant measurements from standard sirens'. In: *AA* 646 (2021), A65. DOI: 10.1051/0004-6361/201936724. URL: https://doi.org/10.1051/0004-6361/201936724 (cit. on p. 65).

[116] Milton Abramowitz and Irene A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Washington, D.C.: Supt. of Docs., U.S. G.P.O., 1968 (cit. on p. 66).

[117] R. C. Bohlin and R. L. Gilliland. 'Hubble Space Telescope Absolute Spectrophotometry of Vega from the Far-Ultraviolet to the Infrared'. In: *The Astronomical Journal* 127.6 (June 2004), p. 3508. DOI: 10.1086/420715. URL: https://dx.doi.org/10.1086/420715 (cit. on p. 66).

[118] C. S. Kochanek et al. 'The K-Band Galaxy Luminosity Function'. In: 560.2 (Oct. 2001), pp. 566–579. DOI: 10.1086/322488. arXiv: astro-ph/0011456 [astro-ph] (cit. on pp. 66, 67).

[119] Milton Abramowitz and Irene A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. ninth Dover printing, tenth GPO printing. New York: Dover, 1964 (cit. on p. 71).

[120] Athanasios Papoulis and S. Unnikrishna Pillai. *Probability, Random Variables, and Stochastic Processes*. Fourth. Boston: McGraw Hill, 2002. ISBN: 0071122567 9780071122566 0073660116 9780073660110 0071226613 9780071226615. URL: http://www.worldcat.org/search?qt=worldcat_org_all&q=0071226613 (cit. on p. 71).

[121] Nelson Christensen and Renate Meyer. 'Parameter estimation with gravitational waves'. In: *Reviews of Modern Physics* 94.2 (Apr. 2022). ISSN: 1539-0756. DOI: 10.1103/revmodphys.94.025001. URL: http://dx.doi.org/10.1103/RevModPhys.94.025001 (cit. on p. 73).

[122] Warren G. Anderson et al. 'Excess power statistic for detection of burst sources of gravitational radiation'. In: *Phys. Rev. D* 63 (4 Jan. 2001), p. 042003. DOI: 10.1103/PhysRevD.63.042003. URL: https://link.aps.org/doi/10.1103/PhysRevD.63.042003 (cit. on pp. 73, 74).

[123] J. Veitch et al. 'Parameter estimation for compact binaries with ground-based gravitational-wave observations using the LALInference software library'. In: *Phys. Rev. D* 91 (4 Feb. 2015), p. 042003. DOI: 10.1103/PhysRevD.91.042003. URL: https://link.aps.org/doi/10.1103/PhysRevD.91.042003 (cit. on pp. 74, 128).

[124] *What are Gravitational Waves?* http://web.archive.org/web/20080207010024/http://www.808multimedia.com/winnt/kernel.htm. Accessed: 2023-12-27 (cit. on p. 75).

[125] B. P. Abbott, R. Abbott and et al. Abbott. 'Binary Black Hole Population Properties Inferred from the First and Second Observing Runs of Advanced LIGO and Advanced Virgo'. In: *The Astrophysical Journal Letters* 882.2 (Sept. 2019), p. L24. ISSN: 2041-8213. DOI: 10.3847/2041-8213/ab3800. URL: http://dx.doi.org/10.3847/2041-8213/ab3800 (cit. on pp. 75–77).

[126] S. E. Woosley. 'Pulsational Pair-instability Supernovae'. In: *The Astrophysical Journal* 836.2 (Feb. 2017), p. 244. ISSN: 1538-4357. DOI: 10.3847/1538-4357/836/2/244. URL: http://dx.doi.org/10.3847/1538-4357/836/2/244 (cit. on p. 76).

[127] Matt Bonakdarpour. *Inverse Transform Sampling — stephens999.github.io*. https://stephens999.github.io/fiveMinuteStats/inverse_transform_sampling.html. [Accessed 13-03-2024] (cit. on p. 78).

[128] Charles R. Harris et al. 'Array programming with NumPy'. In: *Nature* 585.7825 (Sept. 2020), pp. 357–362. DOI: 10.1038/s41586-020-2649-2. URL: https://doi.org/10.1038/s41586-020-2649-2 (cit. on p. 79).

[129] R. Abbott and LVK collaboration. 'All-sky search for continuous gravitational waves from isolated neutron stars using Advanced LIGO and Advanced Virgo O3 data'. In: *Phys. Rev. D* 106 (10 Nov. 2022), p. 102008. DOI: 10.1103/PhysRevD.106.102008. URL: https://link.aps.org/doi/10.1103/PhysRevD.106.102008 (cit. on p. 81).

[130] Christian E A Chapman-Bird, Christopher P L Berry and Graham Woan. 'Rapid determination of LISA sensitivity to extreme mass ratio inspirals with machine learning'. In: *Monthly Notices of the Royal Astronomical Society* 522.4 (May 2023), pp. 6043–6054. ISSN: 0035-8711. DOI: 10.1093/mnras/stad1397. eprint: https://academic.oup.com/mnras/article-pdf/522/4/6043/50390741/stad1397.pdf. URL: https://doi.org/10.1093/mnras/stad1397 (cit. on p. 83).

[131] The LIGO Scientific Collaboration The Virgo Collabo- ration and The KAGRA Collaboration. 'Prospects for observing and localizing gravitational-wave transients with Advanced LIGO, Advanced Virgo and KAGRA'. In: *Living Reviews in Relativity* 23.1 (Sept. 2020). DOI: 10.1007/s41114-020-00026-9. URL: https://doi.org/10.1007%2Fs41114-020-00026-9 (cit. on p. 83).

[132] Geraint Pratten et al. 'Computationally efficient models for the dominant and subdominant harmonic modes of precessing binary black holes'. In: *Physical Review D* 103.10 (May 2021). ISSN: 2470-0029. DOI: 10.1103/physrevd.103.104056. URL: http://dx.doi.org/10.1103/PhysRevD.103.104056 (cit. on pp. 84, 141).

[133] C.E. Shannon. 'Communication in the Presence of Noise'. In: *Proceedings of the IRE* 37.1 (Jan. 1949), pp. 10–21. DOI: 10.1109/jrproc.1949.232969. URL: https://doi.org/10.1109/jrproc.1949.232969 (cit. on p. 84).

[134] Xavier Glorot, Antoine Bordes and Yoshua Bengio. 'Deep Sparse Rectifier Neural Networks'. In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. PMLR. 2011, pp. 315–323 (cit. on p. 85).

[135] Michael J. Williams et al. *uofgravity/glasflow: v0.2.0*. Version v0.2.0. Feb. 2023. DOI: 10.5281/zenodo.7598678. URL: https://doi.org/10.5281/zenodo.7598678 (cit. on pp. 105, 107, 181).

[136] Conor Durkan et al. *nflows: normalizing flows in PyTorch*. Version v0.14. Nov. 2020. DOI: 10.5281/zenodo.4296287. URL: https://doi.org/10.5281/zenodo.4296287 (cit. on pp. 105, 106, 181).

[137] Sebastian Ruder. *An overview of gradient descent optimization algorithms*. 2017. arXiv: 1609.04747 [cs.LG] (cit. on p. 106).

[138] Adam Paszke et al. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. 2019. arXiv: 1912.01703 [cs.LG]. URL: https://arxiv.org/abs/1912.01703 (cit. on p. 107).

[139] NVIDIA. *CUDA Toolkit Documentation*. 2024. URL: https://developer.nvidia.com/cuda-toolkit (cit. on p. 107).

[140] George Papamakarios, Theo Pavlakou and Iain Murray. *Masked Autoregressive Flow for Density Estimation*. 2018. arXiv: 1705.07057 [stat.ML] (cit. on p. 114).

[141] Jesper W. Schneider. *Null hypothesis significance tests: A mix-up of two different theories, the basis for widespread confusion and numerous misinterpretations*. 2014. arXiv: 1402.1089 [stat.OT]. URL: https://arxiv.org/abs/1402.1089 (cit. on p. 117).

[142] Greg Ashton. *Understanding P-P Plots*. https://greg-ashton.physics.monash.edu/understanding-pp-plots.html. 2021 (cit. on p. 117).

[143] Gregory Ashton et al. 'BILBY: A user-friendly Bayesian inference library for gravitational-wave astronomy'. In: *Astrophys. J. Suppl.* 241.2 (2019), p. 27. DOI: 10.3847/1538-4365/ab06fc. arXiv: 1811.02042 [astro-ph.IM] (cit. on p. 126).

[144] I. M. Romero-Shaw et al. 'Bayesian inference for compact binary coalescences with bilby: validation and application to the first LIGO–Virgo gravitational-wave transient catalogue'. In: *Mon. Not. Roy. Astron. Soc.* 499.3 (2020), pp. 3295–3319. DOI: 10.1093/mnras/staa2850. arXiv: 2006.00714 [astro-ph.IM] (cit. on p. 126).

[145] Gregory Ashton and Colm Talbot. 'B ilby-MCMC: an MCMC sampler for gravitational-wave inference'. In: *Mon. Not. Roy. Astron. Soc.* 507.2 (2021), pp. 2037–2051. DOI: 10.1093/mnras/stab2236. arXiv: 2106.08730 [gr-qc] (cit. on p. 126).

[146] Joshua S Speagle. 'dynesty: a dynamic nested sampling package for estimating Bayesian posteriors and evidences'. In: *Monthly Notices of the Royal Astronomical Society* 493.3 (Feb. 2020), pp. 3132–3158. ISSN: 1365-2966. DOI: 10.1093/mnras/staa278. URL: http://dx.doi.org/10.1093/mnras/staa278 (cit. on pp. 127, 128).

[147] Dansbecker. *Using categorical data with one hot encoding*. Jan. 2018. URL: https://www.kaggle.com/code/dansbecker/using-categorical-data-with-one-hot-encoding/notebook (cit. on p. 130).

[148] B. P. et al. Abbott. 'GW170814: A Three-Detector Observation of Gravitational Waves from a Binary Black Hole Coalescence'. In: *Physical Review Letters* 119.14 (Oct. 2017). ISSN: 1079-7114. DOI: 10.1103/physrevlett.119.141101. URL: http://dx.doi.org/10.1103/PhysRevLett.119.141101 (cit. on p. 132).

[149] Adam G. Riess et al. 'Large Magellanic Cloud Cepheid Standards Provide a 1% Foundation for the Determination of the Hubble Constant and Stronger Evidence for Physics beyond ΛCDM'. In: *The Astrophysical Journal* 876.1 (May 2019), p. 85. ISSN: 1538-4357. DOI: 10.3847/1538-4357/ab1422. URL: http://dx.doi.org/10.3847/1538-4357/ab1422 (cit. on p. 140).

[150] R. Abbott et al. 'Observation of Gravitational Waves from Two Neutron Star–Black Hole Coalescences'. In: *Astrophys. J. Lett.* 915.1 (2021), p. L5. DOI: 10.3847/2041-8213/ac082e. arXiv: 2106.15163 [astro-ph.HE] (cit. on p. 141).

[151] A. Palmese et al. 'A Standard Siren Measurement of the Hubble Constant Using Gravitational-wave Events from the First Three LIGO/Virgo Observing Runs and the DESI Legacy Survey'. In: *The Astrophysical Journal* 943.1 (Jan. 2023), p. 56. ISSN: 1538-4357. DOI: 10.3847/1538-4357/aca6e3. URL: http://dx.doi.org/10.3847/1538-4357/aca6e3 (cit. on pp. 142, 144).

[152] Rubén Ballester et al. *Predicting the generalization gap in neural networks using topological data analysis*. 2023. arXiv: 2203.12330 [cs.LG]. URL: https://arxiv.org/abs/2203.12330 (cit. on p. 162).

[153] R.A. Johnson and D.W. Wichern. *Applied Multivariate Statistical Analysis: Pearson New International Edition*. Pearson Education, 2013. ISBN: 9781292037578. URL: https://books.google.co.uk/books?id=xCipBwAAQBAJ (cit. on pp. 165, 169).

[154] Scott Dodelson. *Modern Cosmology*. Amsterdam: Academic Press, 2003. ISBN: 978-0-12-219141-1 (cit. on pp. 171, 172).

[155] Simone Mastrogiovanni et al. *A novel approach to infer population and cosmological properties with gravitational waves standard sirens and galaxy surveys*. 2023. arXiv: 2305.10488 [id='astro-ph.CO' full$_n$ame$='$CosmologyandNongalacticAstrophysics$'$is$_a$ct Truealt$_n$ame$=$Nonein$_a$rchive$='$astro$-$ph$'$is$_g$eneral$=$Falsedescription$='$Phenomenologyof scalestructureoftheuniverse.Groups, superclusters, voids, intergalacticmedium.Particleastropl darkenergy, darkmatter, baryogenesis, leptogenesis, inflationarymodels, reheating, monopoles, (cit. on p. 171).

[156] A. Vale and J. P. Ostriker. 'Linking halo mass to galaxy luminosity'. In: *Monthly Notices of the Royal Astronomical Society* 353.1 (Sept. 2004), pp. 189–200. ISSN: 1365-2966. DOI: 10.1111/j.1365-2966.2004.08059.x. URL: http://dx.doi.org/10.1111/j.1365-2966.2004.08059.x (cit. on p. 173).

[157] Benjamin P. Moster et al. 'CONSTRAINTS ON THE RELATIONSHIP BETWEEN STELLAR MASS AND HALO MASS AT LOW AND HIGH REDSHIFT'. In: *The Astrophysical Journal* 710.2 (Jan. 2010), pp. 903–923. ISSN: 1538-4357. DOI: 10.1088/0004-637x/710/2/903. URL: http://dx.doi.org/10.1088/0004-637X/710/2/903 (cit. on p. 173).