

Annual Progression Review

Cecilia Maria Fabbri

1 September 2025

CONTENTS

Literature Review	iii
0.1 Bayesian statistics	iv
0.2 Hierarchical Bayesian statistics	v
0.3 Simulation based inference	vi
Completed Work	ix
Work Plan	xi
Personal Development Plan	xiii

LITERATURE REVIEW

Gravitational waves (GWs) are perturbations of spacetime generated by mass distributions with non-null second derivative of the quadrupole mass-moment. Binary systems of stellar objects are exemplary GW emitters and the more compact the object, the bigger the wave produced, making the system more suitable for detection.

The LIGO-Virgo-KAGRA (LVK) collaboration uses a system of interferometers to detect gravitational waves. The latest complete catalogue published by the LVK collaboration is the third gravitational-wave transient catalogue (GWTC-3) and contains about a hundred signals generated by the coalescence of binaries of neutron stars (NSs) and stellar-mass black holes (BHs).

When LVK detect an astrophysical signal, they analyze the data to infer the properties of the event, such as the masses of the objects in the binary, their spins, the distance from the observer and its sky localization. The statistical framework used for individual event analysis is Bayesian inference. With the number of events growing, it becomes crucial to do population analyses, as they allow to set further constraints on the astrophysics underlying stellar objects. To do so, one uses Hierarchical Bayesian statistics.

The models used have multi-dimensional parameters, resulting in probability distributions computationally prohibitive. Traditional methods rely on stochastic sampling methods, such as Markov Chain Monte Carlo (MCMC) or Nested sampling, to infer the probability distribution. However powerful, these methods are expensive, as probability distributions have to be evaluated iteratively million of times for each individual event analysis. Each evaluation requires to generate a waveform on the fly, which heavily contributes to the overall costs. Machine learning approaches have the advantage of moving most of the costs in the training stage, dramatically speeding up inference time, which is crucial for multimessenger astronomy. Several attempts have

been made in this direction, from architectures that model the noise more realistically than traditional methods, to architectures that employ complex physical models without an analytical expression, which is unfeasible with stochastic samplers.

0.1 BAYESIAN STATISTICS

Let us refer to the vector of binary parameters as θ . It consists of intrinsic parameters, such as the masses and the spins, and extrinsic parameters, such as the redshift and the binary orientation. The goal of the analysis is to infer the posterior distribution $p(\theta|d)$, which associates to each point in the binary-parameter space the probability of that binary to generate the observed data d . Observed data is the sum of the astrophysical signal, which is assumed to be deterministic and a function of binary-parameters, and a noise realization in the detector, which is assumed to be stochastic. To infer the posterior one needs a prior $\pi(\theta)$ and a Likelihood $\mathcal{L}(d|\theta)$. The prior is the probability distribution that an event with parameters θ takes place in the Universe. A possible approach to build is based on theory or previous experiments. Nevertheless, the preferred approach is often conservative, and uninformative priors are used, such as priors with constant probability in the binary-parameters dominium and zero outside. The Likelihood is the probability in the data space, given θ parameters. To evaluate the Likelihood one needs to generate waveforms for θ and a noise model. To build the noise model one often assumes that noise is stationary and Gaussian. Bayes theorem links the posterior to the prior and the Likelihood:

$$p(\theta|d) = \frac{\pi(\theta) \times \mathcal{L}(d|\theta)}{p(d)}. \quad (1)$$

The term $p(d)$ is called evidence, and it is the probability of observing the data d under the assumed model, which is often general relativity. The evidence is defined as

$$p(d) = \int d\theta \pi(\theta) \times \mathcal{L}(d|\theta). \quad (2)$$

The evidence is a normalization factor and does not influence the posterior shape. For this reason one does not compute it during inference, except when doing model comparison.

0.2 HIERARCHICAL BAYESIAN STATISTICS

The goal of population analysis is to infer how binary parameters are distributed at the population level. To do so, one parametrizes the distribution of binary parameters with the population parameters λ , that will determine the shape of the distribution. Examples of hyperparameters are the slope of the binary black hole (BBH) mass distribution, the minimum and maximum mass of stellar-mass black holes. The prior on binary parameters is conditioned on the population parameters, and we refer to it as the population model $p_{\text{pop}}(\theta|\lambda)$. For the analysis one needs the prior on population parameters, called hyperprior, $\pi(\lambda)$.

The choice of the population model is crucial, and several approaches have been proposed to construct it. The state of the art are parametric models, which are based on easy to evaluate parametric forms built based on a qualitative astrophysical motivation. Other approaches move towards highly flexible models, called semi-parametric or non-parametric models, where the hyperparameters are for instance spline perturbations. These models allow for data-driven fits but the results are difficult to interpret physically. Moving towards astrophysical approaches, other attempts include comparing data with the output of simulations that synthesize binary populations.

In population analysis one needs to take into account selection effects. Since the events have different probabilities to be detected, the observed population is biased. An event is considered detectable if data exceeds some threshold. The detection probability for an event is thus defined as

$$p_{\text{det}}(\theta) = \int_{d > \text{threshold}} dd p(d|\theta). \quad (3)$$

Let us refer to the dataset of observed events as d and to individual events as d_i , with $i = 1, \dots, N_{\text{obs}}$, where N_{obs} number of observed events. The posterior for the astrophysical distribution, called hyperposterior, is:

$$p(\lambda|d) = \frac{\pi(\lambda)}{p(d)} \prod_{i=1}^{N_{\text{obs}}} \frac{\int d\theta \mathcal{L}(d_i|\theta) p_{\text{pop}}(\theta|\lambda)}{\int d\theta p_{\text{pop}}(\theta|\lambda) p_{\text{det}}(\theta)}, \quad (4)$$

with $\mathcal{L}(d_i|\theta)$ the likelihood of individual events in Eq. 1. The term $p(d)$ is the hyperevidence, and it is the probability of the observed dataset under the assumed population model, i.e.:

$$p(d) = \int d\lambda' \pi(\lambda') \prod_{i=1}^{N_{\text{obs}}} \mathcal{L}(d_i|\lambda'), \quad (5)$$

with the likelihood respect to the hyperparameters $\mathcal{L}(d_i|\lambda')$ being linked to the individual-event likelihood through

$$\mathcal{L}(d_i|\lambda') = \int d\theta \mathcal{L}(d_i|\theta) p_{\text{pop}}(\theta|\lambda'). \quad (6)$$

The hyperevidence is often computed in population analysis to compare different population models and their goodness in describing the observed dataset.

Population analysis is divided in two steps. First, individual events are analyzed with Bayesian statistics. Then, one uses the individual-event posteriors and the assumed population model to infer the hyperposterior in Eq. 4.

0.3 SIMULATION BASED INFERENCE

Simulation-based inference (SBI) is a deep learning technique to approximate a probability distribution using simulated data. I am working with the DINGO code, which uses SBI to model a posterior. To simulate data, we need to draw samples from the prior and the likelihood. Contrarily to traditional methods, the likelihood is not evaluated and an analytical expression is not necessary. In principle, more complex models can be used, for instance dropping the stationary and gaussian noise assumption or using waveforms from simulations.

Normalizing flows are often used in SBI to approximate complex distributions. A normalizing flow is a mapping from a simple $\pi(u)$ to an approximation of the posterior $q_\phi(\theta|d)$, with ϕ parameters of the mapping. The mapping $f_\phi(u)$ must be invertible and with a simple Jacobian determinant, so that:

$$q_\phi(\theta|d) = \pi(f_\phi^{-1}(\theta))|\det J_{f_\phi}^{-1}|. \quad (7)$$

To evaluate $q_\phi(\theta|d)$ one uses Eq. 7. Drawing samples from $q_\phi(\theta|d)$ is equivalent to drawing $u \sim \pi(u)$ and transforming $\theta = f_\phi(u)$. Often one chooses the base distribution $\pi(u)$ is a multivariate standard normal. In the case of DINGO code, the flow f_ϕ is a sequence of transforms $f_\phi^{(i)}$, so that

$$f_\phi^i(u) = u_i \text{ if } i \leq d/2, c_i(u_i; u_{1:d,d}) \text{ if } i > d/2, \quad (8)$$

so that half of the latent variables u are unchanged while the other half is transformed. At each $f_\phi^{(i)}$ the indices of the u that transform and that do not are permuted. The transformation is the neural spline coupling transform c_i , whose parameters depend on the unchanged u_i and on data strain d .

During training the neural network learns the mapping by learning the ϕ parameters that minimize a chosen loss function.

The loss function used in DINGO is:

$$L = \frac{-1}{N} \quad (9)$$

CAPIRE MEGLIO E RISPIEGARE.

- we need to draw from the likelihood -> can use non-analytical models - faster because we do all at the beginning and then tada (gia' detto nell'intro)

COMPLETED WORK

a description of work completed to date - small project description

Transfer learning is a series of techniques used in machine learning to transfer knowledge from a neural network to another. Each time one modifies a neural network, for instance changing the model assumed in the analysis, the neural network needs to be trained again. The goal of transfer learning is to fine tune a new training run using some knowledge acquired during a previous training. These strategies allow to reduce training time, and, more importantly, could even improve the results respect to training the new run without any information from previous trainings.

WORK PLAN

a plan of work for the next 12 months - pop growing pains for sure - we talked about using astrophysical models as pop models. We need to understand better. Surely something more astro - another idea is to keep developing the current project (NR simulations, beyond GR models)

PERSONAL DEVELOPMENT PLAN

Copy of personal development plan summary and a statement of progress made towards those training goals

