

# FYP

Sean White

January 2025

## 1 Background

The gold standard gravitational waveforms are obtained by numerical relativity simulations which involve directly solving Einstein's equations of general relativity which is massively computationally expensive. For this reason, only several thousand simulations are currently available. As a result, GW models rely on analytical or semi-analytical prescriptions that are calibrated to the numerical relativity simulations.

Each of these modelling approaches will incur a certain amount of error. We quantify this error as the mismatch between the "true" relativity simulation of a gravitational waveform and the analytic model for the waveform. The mismatch [2] varies between 0, signifying that the model and the true waveform are identical (up to an overall amplitude rescaling), and 1, meaning that the two are completely orthogonal. To begin we will delve into the methods used before to "combine" these models and then discuss the new proposed method. This will lead us onto my project.

### Methods Used Before

hello buddy

#### Standard Method

We build a distribution based off of each of the 3 models. We then combine these distributions with equal weight to form a new total distribution from which we make our predictions from.

#### Evidence-Informed Method

This method again builds a distribution based off of each of the 3 models. The distributions are then combined with weights which are determined by how well

the model explains the observed data. From this new distribution predictions are made.

## **New Proposed Method: Numerical Relativity Informed Method [1]**

### **Numerical Relativity Informed Method**

This method builds a distribution based off of each of the 3 models. Then the mismatch (difference between NR simulations and model) is calculated. We then find the ratios of the mismatch between each model, for example:

$$\frac{\text{mismatch}(\text{model 1})}{\text{mismatch}(\text{model 2})}.$$

A distribution using the ratio of mismatch of each model is built. This distribution is used to determine the probability of selecting each model in each region of the parameter space. Predictions are then made using the model selected in that region of the parameter space using the ratio distribution. This selection of the model is probabilistic and so the best model in a certain region may be chosen 85% of the time and the second best 10% etc etc.

### **Comparison**

The Standard Method and the Evidence-Informed Method build a distribution from which we make our predictions. The Numerical Relativity Informed method builds a distribution which helps us choose the best model in that parameter space “most of the time”.

Now the issue is that to calculate the mismatch for the models needed in the NR informed method, we must have gravitational waveforms generated from NR simulations. This as we discussed is a computationally expensive process and is often not feasible. The goal of my project is to build a Gaussian Process Regression model to model the mismatch of the model’s to the NR simulations.

## **2 Gaussian Process Regression GPR Background**

### **Definition of a Gaussian Process**

A Gaussian Process (**GP**) defines a probabilistic model over all possible functions rather than assuming a single function to be true:

$$f(X) \sim \mathcal{GP}(m(X), k(X, X'))$$

where  $m(X)$  is the mean function, specifying the expected function value at each  $X$ :

$$m(X) = E[f(X)]$$

$k(X, X')$  is the covariance function (kernel), encoding the relationships between function values at different points:

$$k(X, X') = \text{Cov}(f(X), f(X'))$$

Since a function is defined over a continuous input space, a GP is formally a distribution over infinitely many function values. We approximate the function by evaluating it at a finite set of points. These function values are then assumed to follow a multivariate normal Gaussian distribution.

Mathematically, for a finite set of input points

$$X = \{X_1, X_2, \dots, X_n\}$$

the corresponding function values

$$f = \{f(X_1), f(X_2), \dots, f(X_n)\}$$

follow a multivariate normal distribution:

$$f \sim \mathcal{N}(m(X), K(X, X))$$

Each sample from this multivariate distribution represents a function evaluated at  $n$  different points. Every function sample will have a different shape but will respect the prior assumptions encoded in  $m(X)$  and  $k(X, X')$ .

## 2.1 The Prior Distribution

Initially, we build the prior distribution of our GP without knowing any of the true data. We make assumptions about our model. We assume that the training points are distributed as:

$$f(X) \sim \mathcal{N}(m(X), K(X, X))$$

and that the test points are distributed as:

$$f(X_*) \sim \mathcal{N}(m(X_*), K(X_*, X_*))$$

The joint distribution of this prior is then:

$$\begin{bmatrix} f(X) \\ f(X_*) \end{bmatrix} \sim \mathcal{N} \left( \begin{bmatrix} m(X) \\ m(X_*) \end{bmatrix}, \underbrace{\begin{bmatrix} K(X, X) & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix}}_{\mathcal{C}=\text{Covariance Matrix}} \right)$$

We use different Kernel functions (Covariance) to see which one results in the best model (detailed in methods/results). Note: From now on we deal with normalized data and so take  $\mu(X) = \mu(X_*) = 0$ .

## 2.2 How the Kernel effects the Prior

In this section we show kernel effects the distributions for different kernels. We define the different kernels and will later show in methods/results how the process we used to choose which kernel fits our data best

## 2.3 Adding Data: Building the Posterior

By conditioning our prior distribution on our training data (true data points), we build our posterior distribution from which we can make predictions.

The function values at test points are conditioned on the function values at training points:

$$P(f_*|f, X, X_*).$$

Functionally, since the covariance structure already incorporates  $X$  and  $X_*$  through the kernel function, this reduces to computing:

$$P(f(X_*)|f(X)).$$

Since we are dealing with continuous distributions, we have that the conditional probability is represented by the probability density function  $p(f_*|f)$ . Using Bayes' Theorem applied to continuous probabilities, we have:

$$p(f_*|f) = \frac{p(f_*, f)}{p(f)}.$$

we have:

$$p(f_*, f) = \frac{1}{2\pi\sqrt{|\mathbf{C}|}} \exp\left(-\frac{1}{2} \begin{bmatrix} f \\ f_* \end{bmatrix}^T \mathbf{C}^{-1} \begin{bmatrix} f \\ f_* \end{bmatrix}\right)$$

and

$$p(f) = \frac{1}{\sqrt{2\pi}|K|^{1/2}} \exp\left(-\frac{1}{2} f^T K^{-1} f\right).$$

**Legend:**

- $K = K(X, X)$ ,  $K_{**} = K(X_*, X_*)$ ,  $K_* = K(X, x_*) = K(X_*, X)^T$
- $C = \begin{bmatrix} K & K_* \\ K_*^T & K_{**} \end{bmatrix}$
- $|\mathbf{C}| = KK_{**} - K_*K_*^T$ : Determinant of the covariance matrix.
- $\mathbf{C}^{-1} = \frac{1}{|\mathbf{C}|} \begin{bmatrix} K_{**} & -K_* \\ -K_*^T & K \end{bmatrix}$
- $m(X) = m(X_*) = 0$

We get out (do out in appendix):

$$p(f_*|f) = \frac{1}{\sqrt{2\pi}|K_{**} - K_*^T K^{-1} K_*|^{1/2}} \exp\left(-\frac{1}{2}f_*^T (K_{**} - K_*^T K^{-1} K_*)^{-1} f_*\right).$$

Since this follows a multivariate Gaussian distribution, we write:

$$p(f_*|f) \sim \mathcal{N}(m(f_*), \text{Var}(f_*)).$$

where:

$$\begin{aligned} m(f_*) &= K_*^T K^{-1} f. \\ \text{Var}(f_*) &= K_{**} - K_*^T K^{-1} K_*. \end{aligned}$$

## 2.4 Graphical Representation of prior to posterior

## 2.5 Hyper-parameter Optimisation

We have seen how we condition our prior on data with kernel function. But we can optimise our hyper-parameters to find the best fitting posterior distribution.

Done by log-likelihood and L2 norm cross validation

## 3 Next steps for results

- Find a scientific approach and well justifiable reason on what kernel was picked and for what reasons.
- Use Different Cross Validation and optimising techniques and again find a justifiable reason on which one performs best and for what reasons.
- Include the variance in the Kernel hyper-parameters into my posterior. Accounts for more general results, less effected by noise
- Compare under computational cost and accuracy the model done in different dimensional spaces. 4d,7d,8d.
- Run these different models on other mismatch data from other functional fit and interpolant models to find which model generalizes best.
- justify 4 dimensional hyper-parameters

## 4 Kernels

## 5 Different Models

- White Kernel without Error Knowledge
- White Kernel With Error Knowledge
- Using  $\alpha = err^2$
- Using Monte Carlo Sampling, Sampling from the Error
- Using MCMCing to build a posterior on hyper-parameter distribution
- Picking values from this posterior distribution to use

## 6 Choosing the Best Performing Model take performance and Efficiency into account

### 6.1 Kernel Selection

## References

- [1] Charlie Hoy et al. “Incorporating multi-model uncertainty into gravitational-wave Bayesian inference”. In: In preparation (2024). To be published. URL: <https://dcc.ligo.org/LIGO-T2000012/public>.
- [2] Benjamin J. Owen. “Search templates for gravitational waves from inspiraling binaries: Choice of template spacing”. In: arXiv preprint gr-qc/9511032 (1995). Submitted to Physical Review D: November 7, 1995. URL: <https://arxiv.org/abs/gr-qc/9511032>.