

# Machine Learning for Gravitational Wave data analysis

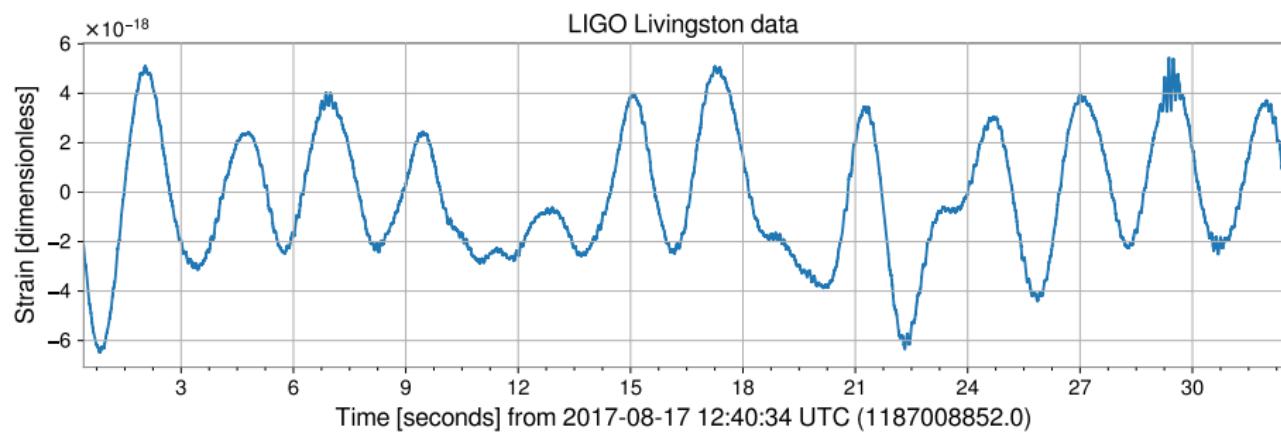
Jacopo Tissino

2022-04-08

# Virgo interferometer



# Bare interferometer data



# Describing Gaussian noise

We can completely characterize Gaussian noise through its **power** or **amplitude** spectral density:

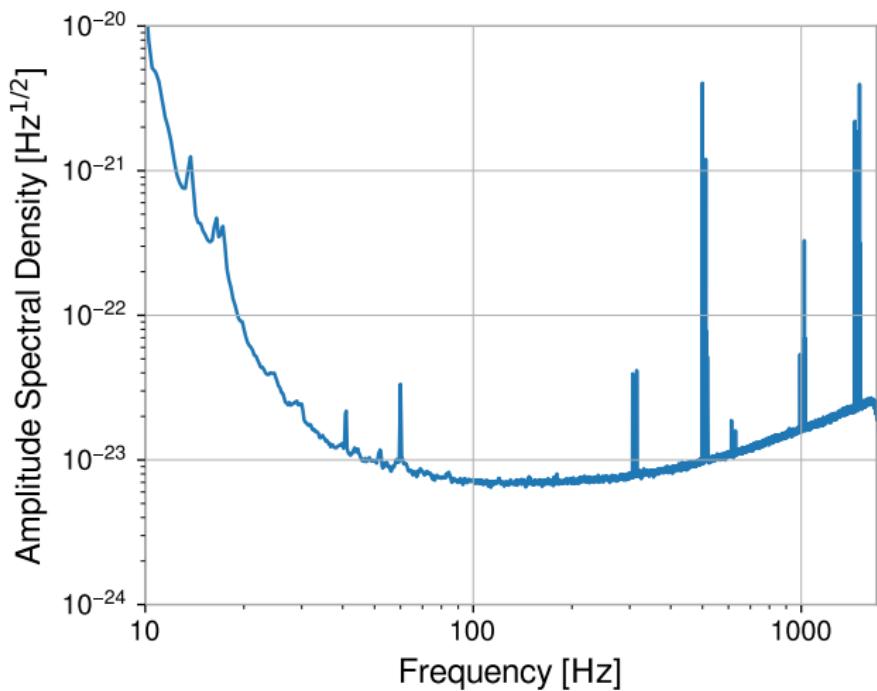
$$\text{PSD}(f) = S_n(f) = \lim_{T \rightarrow \infty} \frac{\left| \tilde{d}(f) \right|^2}{T}, \quad (1)$$

$$\text{ASD}(f) = \sqrt{\text{PSD}(f)}, \quad (2)$$

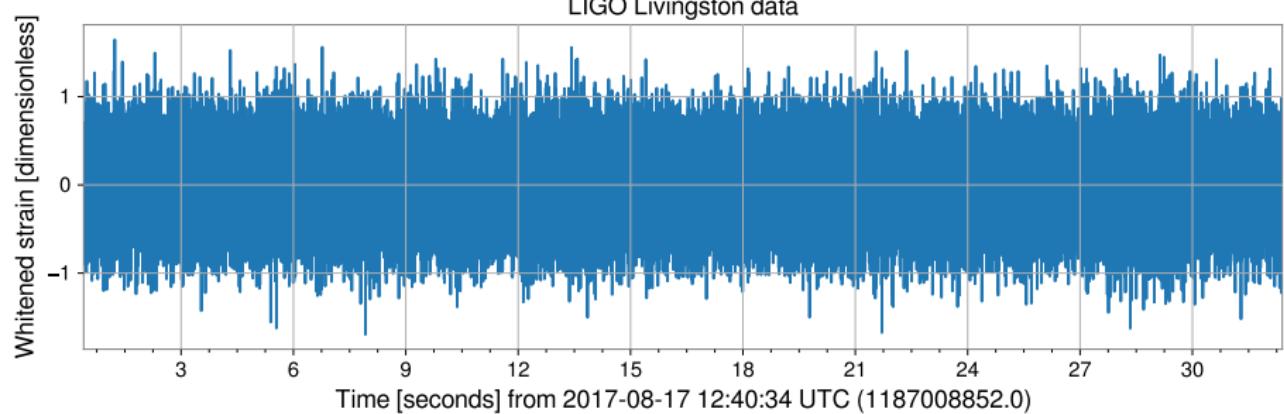
and then we can whiten the signal as

$$\tilde{d}_w(f) = \frac{\tilde{d}(f)}{\sqrt{S_n(f)}}. \quad (3)$$

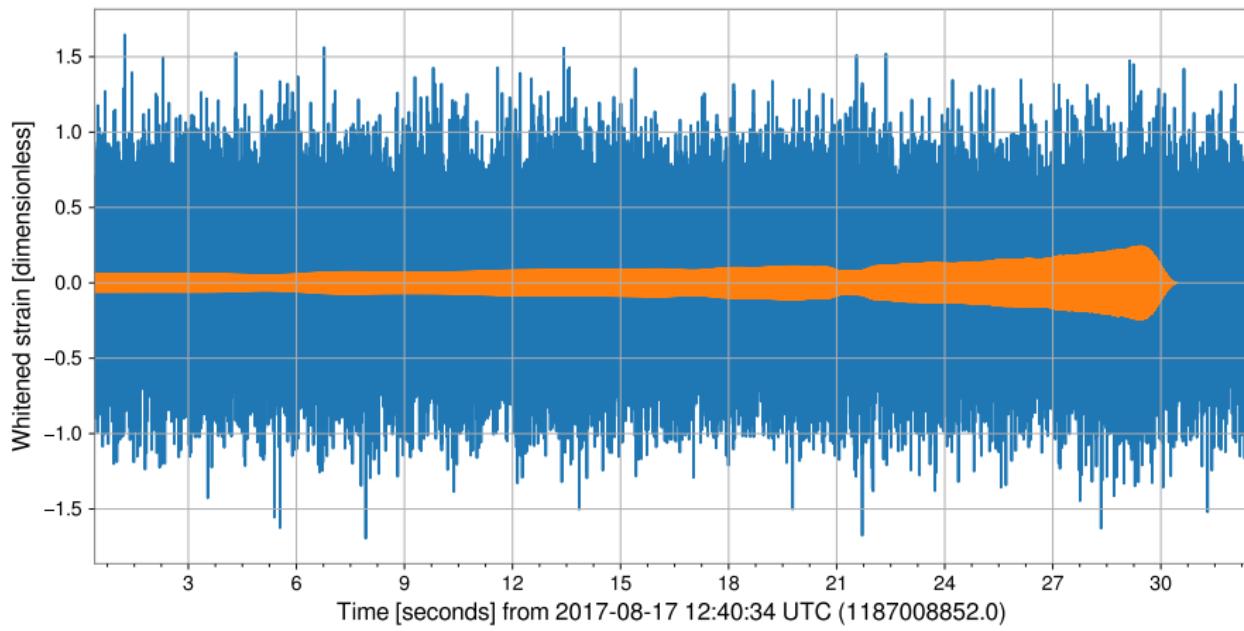
# Amplitude spectral density



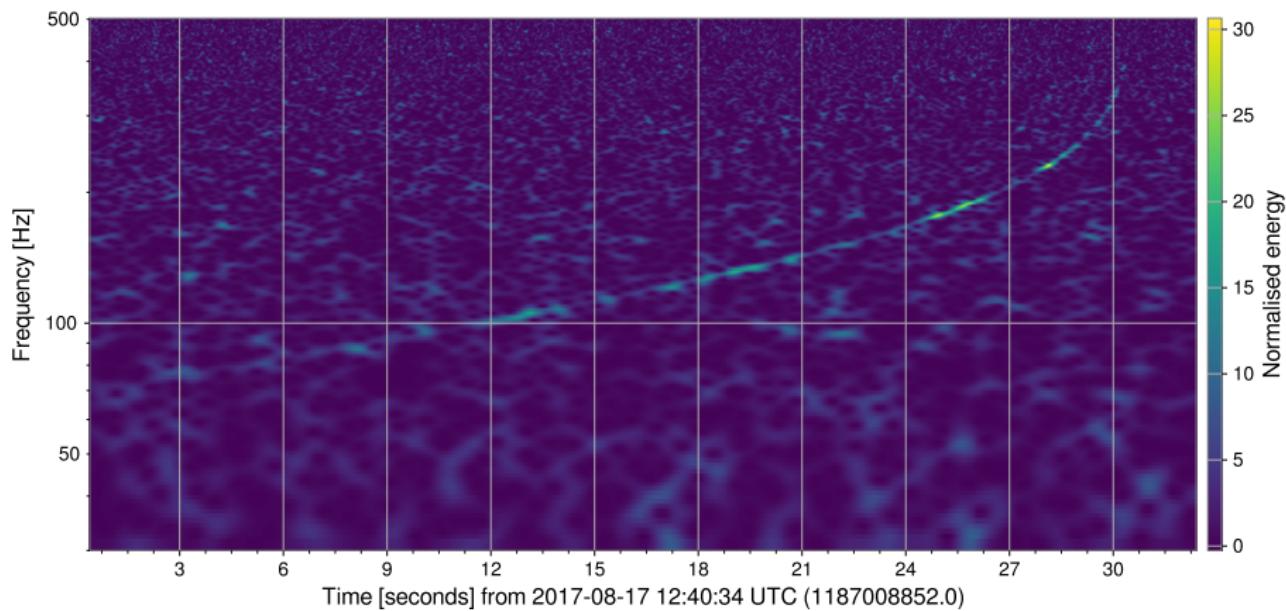
# Whitened, bandpassed data



The signal is small



## Q-transform



# Signal parametrization

The strain at the detector is modelled as  $s(t) = h_\theta(t) + n(t)$ , where:

- the noise  $n(t)$  is taken to be stationary, with zero mean, and Gaussian with power spectral density  $S_n(f)$ ;
- the signal  $h_\theta(t)$  can depend on:
  - intrinsic parameters: total mass  $M = m_1 + m_2$ , mass ratio  $q = m_1 / m_2$ , spins  $\vec{\chi}_1$  and  $\vec{\chi}_2$ , tidal polarizabilities  $\Lambda_1$  and  $\Lambda_2$ ;
  - extrinsic parameters: luminosity distance  $D_L$ , inclination  $\iota \dots$

# The Wiener distance

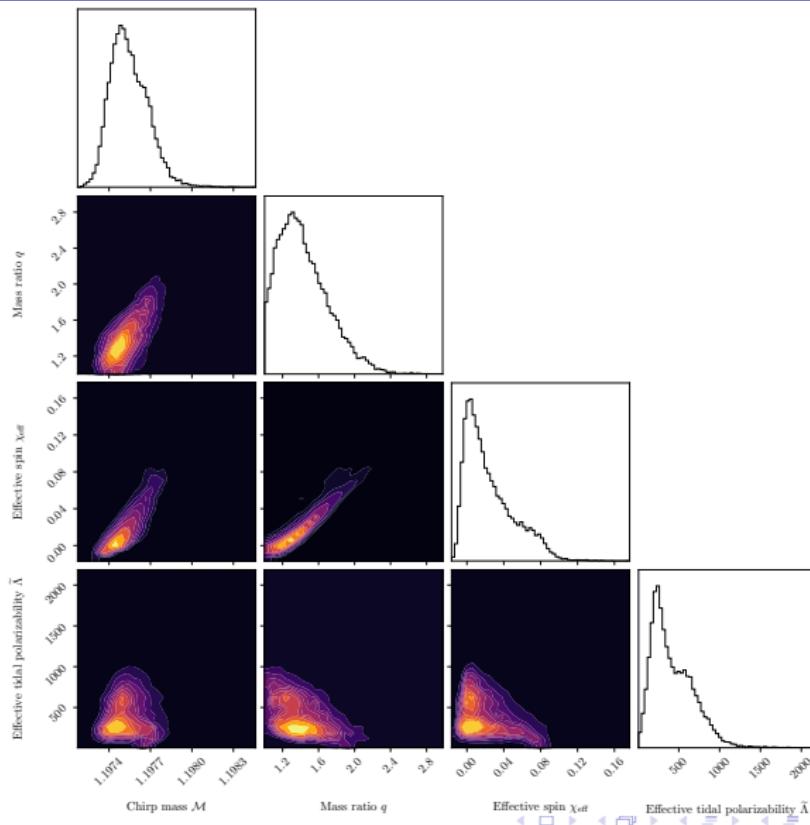
The likelihood used in parameter estimation reads:

$$\Lambda(s|\theta) \propto \exp\left((h_\theta|s) - \frac{1}{2}(h_\theta|h_\theta)\right), \quad (4)$$

where  $(a|b)$  is the Wiener product:

$$(a|b) = 4 \operatorname{Re} \int_0^\infty \frac{\tilde{a}^*(f)\tilde{b}(f)}{S_n(f)} df = 4 \operatorname{Re} \int_0^\infty a_w^*(f)b_w(f) df. \quad (5)$$

## A posterior distribution: GW170817



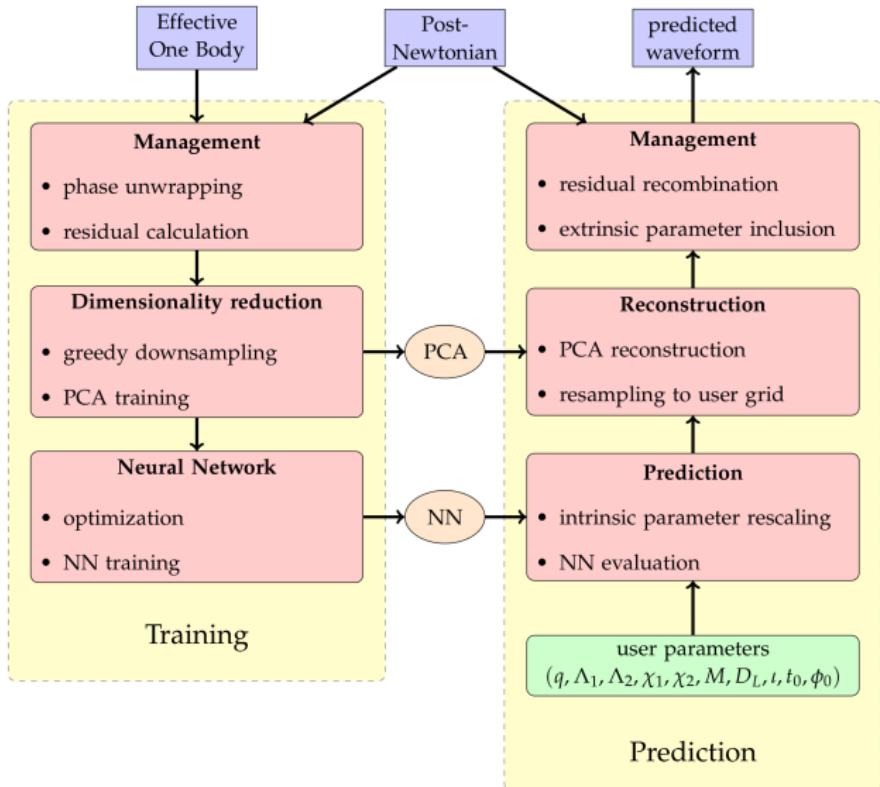
# Theoretical signal models

The main strategies for the generation of theoretical waveforms are:

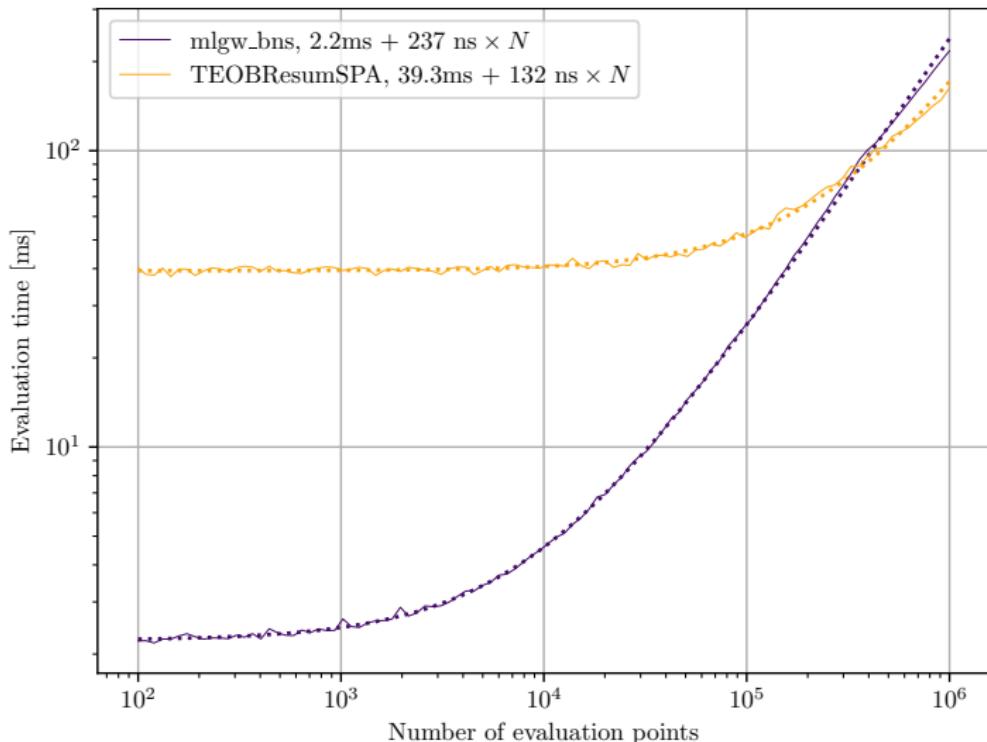
- numerical relativity;
- effective one body;
- post-Newtonian.

Other methods mix and match these: hybrid waveforms, phenomenological models, **surrogates**.

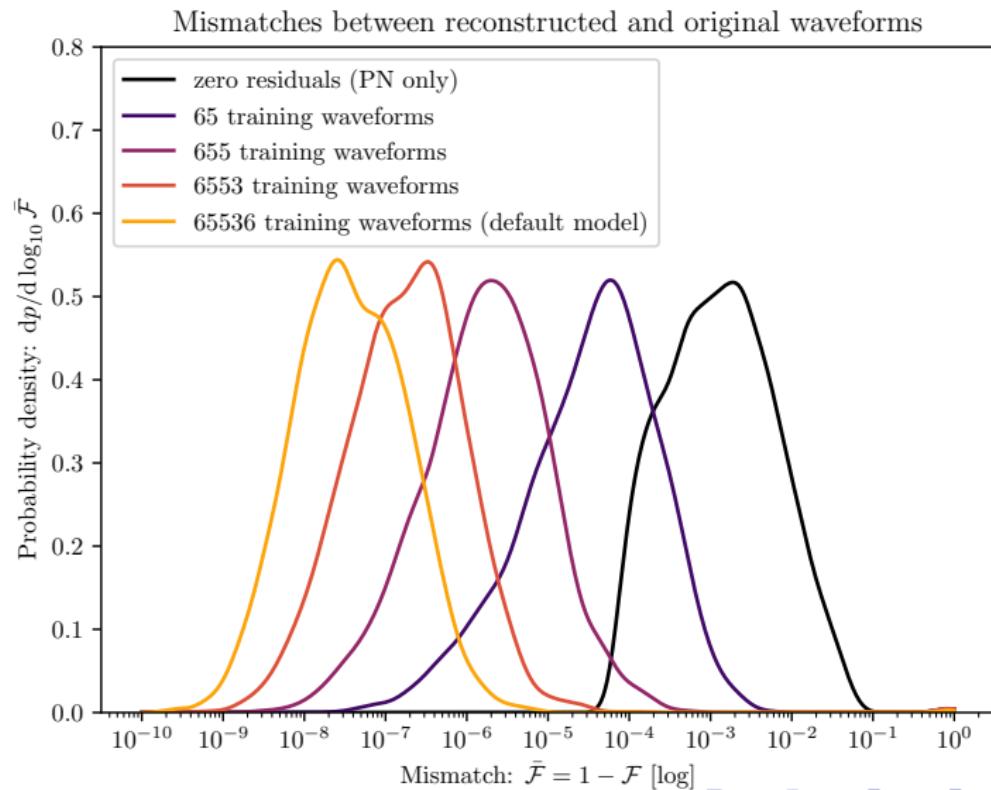
# mlgw\_bns structure



# Evaluation time



# Fidelity



## More information

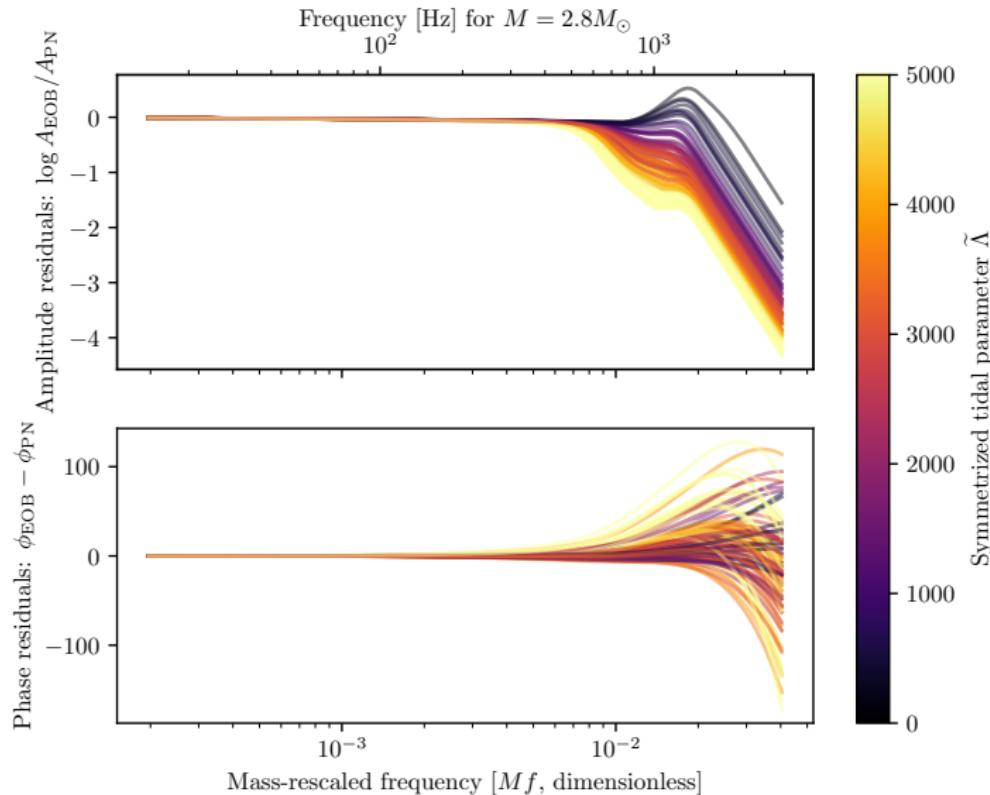
- Learn about GW data analysis at [gw-openscience.org](http://gw-openscience.org);
- documentation for `mlgw_bns` is available at [mlgw-bns.readthedocs.io](https://mlgw-bns.readthedocs.io).

# Technologies

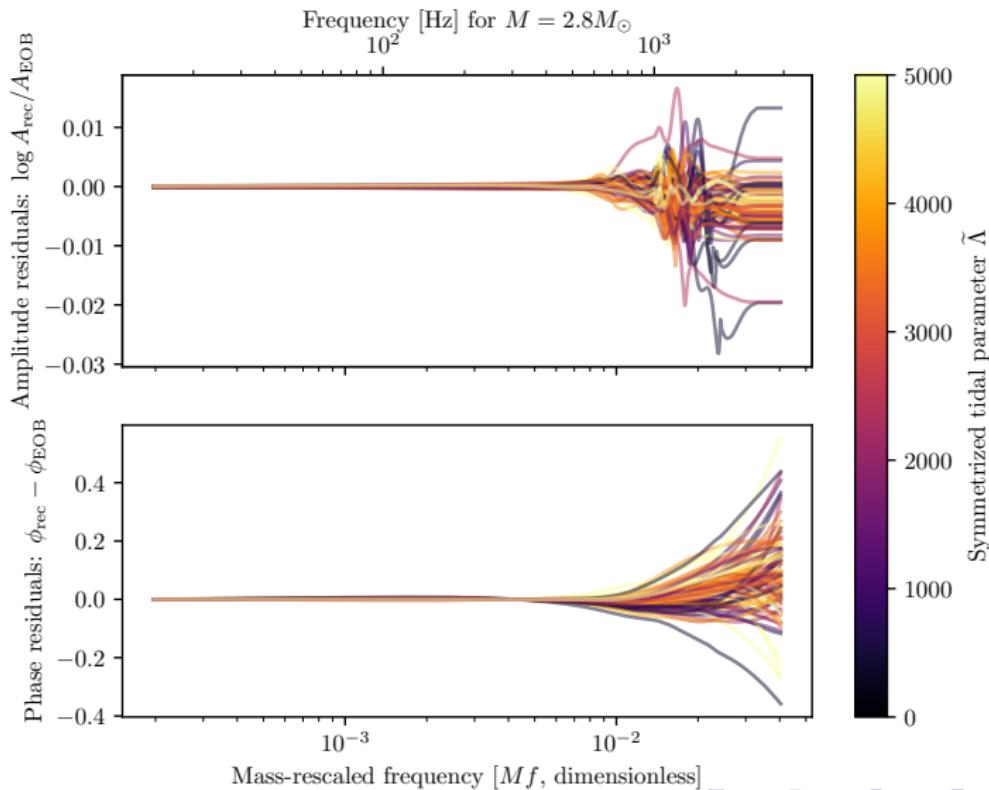
`mlgw_bns` is implemented as a python package, and it makes use of

- scikit-learn for the neural network (upgrading to pytorch);
- optuna for the hyperparameter optimization;
- pytest and tox for automated testing;
- numba for just-in-time compilation and acceleration.

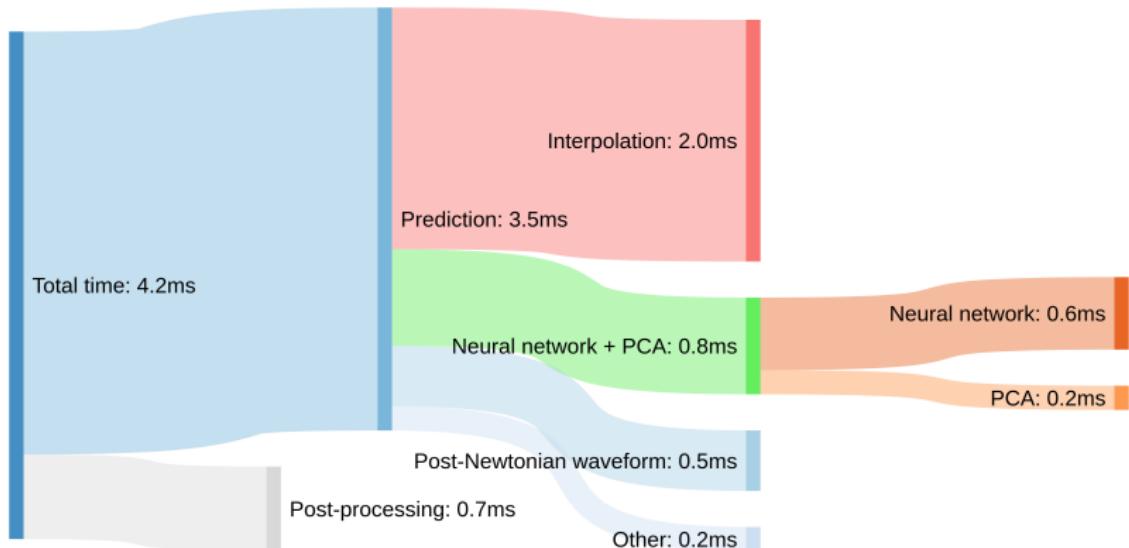
# Original residuals



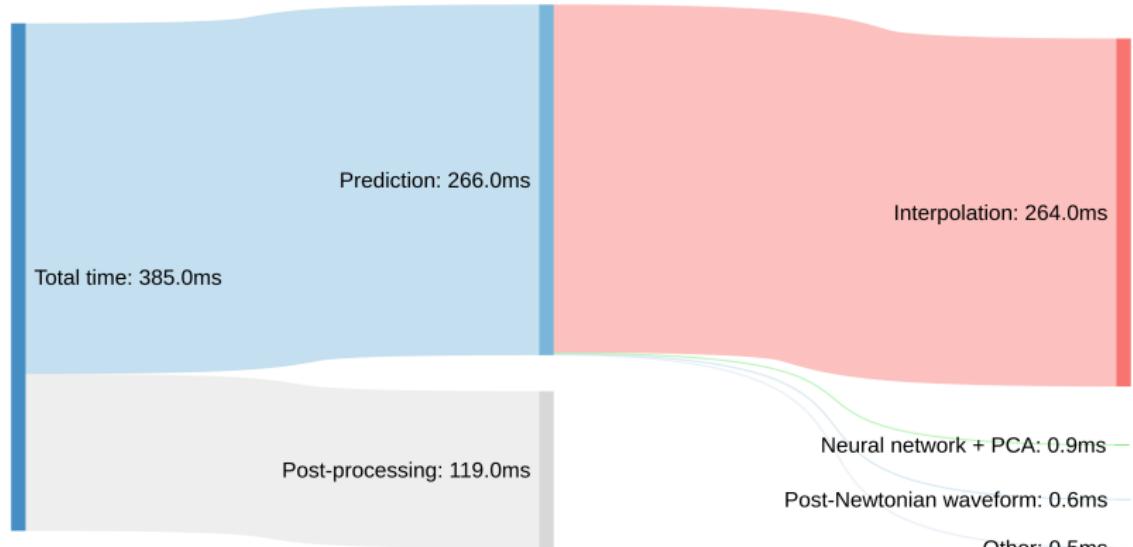
# Reconstruction residuals



## Profiling the evaluation: $8 \times 10^3$ interpolation points



Profiling the evaluation:  $2 \times 10^6$  interpolation points

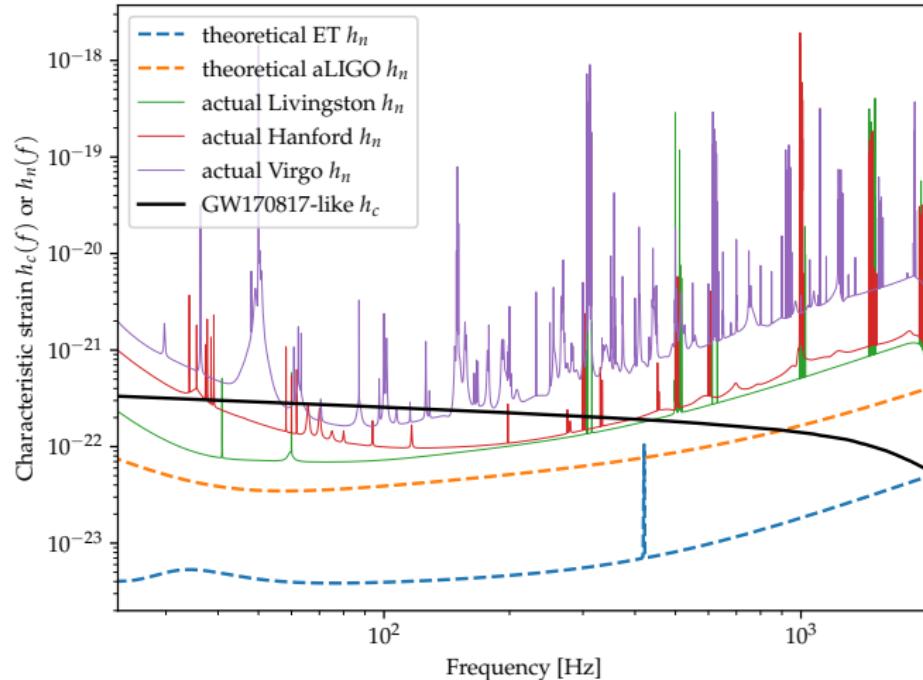


# Hyperparameter optimization

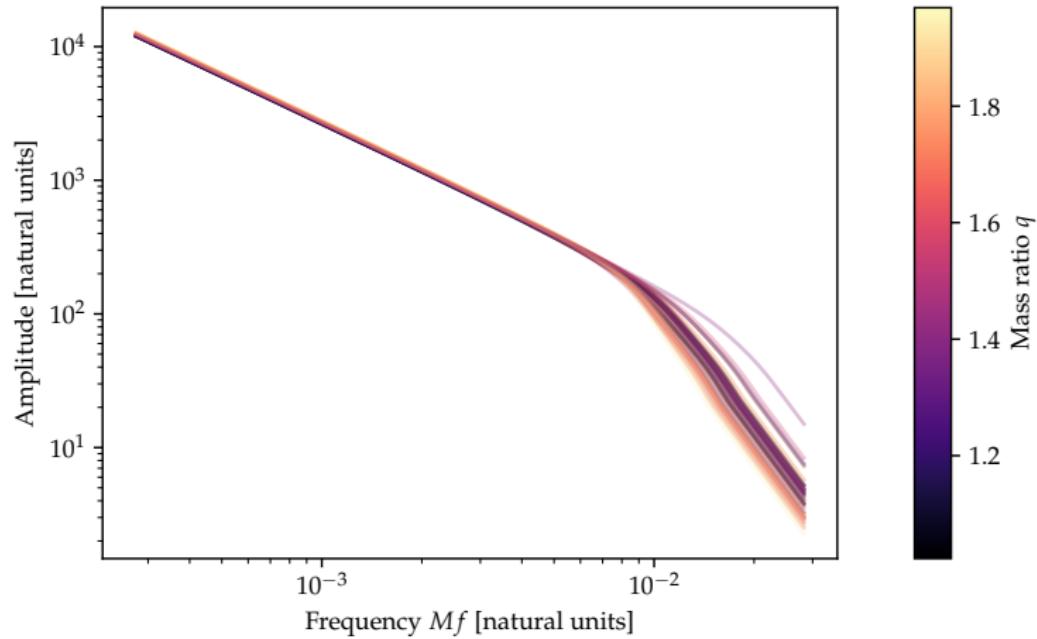
Pareto-front Plot



# Power Spectral densities and GW170817



# Amplitudes



# Phases

