Addressing out-of-distribution data for flow-based gravitational wave inference

Maximilian Dax*
MPI for Intelligent Systems, Tübingen
maximilian.dax@tuebingen.mpg.de

Stephen R. Green*
MPI for Gravitational Physics, Potsdam
University of Nottingham, Nottingham
stephen.green@aei.mpg.de

Jonas Wildberger*
MPI for Intelligent Systems, Tübingen
wildberger.jonas@tuebingen.mpg.de

Jonathan Gair
MPI for Gravitational Physics, Potsdam

Michael Pürrer
MPI for Gravitational Physics, Potsdam
University of Rhode Island, Kingston

Jakob H. Macke
MPI for Intelligent Systems, Tübingen

Alessandra Buonanno
MPI for Gravitational Physics, Potsdam

Bernhard Schölkopf
MPI for Intelligent Systems, Tübingen

Abstract

Simulation-based inference and normalizing flows have recently demonstrated excellent performance when applied to gravitational-wave parameter estimation. These methods can provide accurate results within seconds, in cases where classical methods based on stochastic samplers may take days or even weeks. However, such methods are typically based on deep neural networks and thus unable to reliably deal with out-of-distribution data, such as may arise when predicted signal and noise models do not precisely fit observations. We here present two innovations to deal with this challenge. First, we introduce a probabilistic noise model to augment the training data, making the inference network substantially more robust to distribution shifts in experimental noise. Second, we apply importance sampling to independently verify and correct inference results. This compensates for network inaccuracies and flags failure cases via low sample efficiencies. We expect these methods to be key components for the integration of deep learning techniques into production pipelines for gravitational-wave analysis.

1 Introduction

Since 2015, the LIGO [1], Virgo [2] and KAGRA [3–5] gravitational wave (GW) observatories have detected gravitational radiation from 90 astrophysical mergers of black holes or neutron stars [6–8]. Each of these is analyzed using Bayesian inference to compare against predictions from Einstein's theory of general relativity and determine the properties of the source. In turn, this has informed our knowledge of extreme matter and gravity [9, 10], the formation and evolution of binaries [11], and even the expansion of the universe [12].

*Equal contribution

Machine Learning and the Physical Sciences workshop, NeurIPS 2022.

Observed data in each detector is a time series d, assumed to consist of a GW signal $h(\theta)$ and additive noise n. The signal depends on 15 parameters θ, corresponding to the masses and spins of the binary components, along with the orientation and location of the binary in space and time. The noise is assumed to be stationary and Gaussian, described by a power spectral density (PSD) S_n, which can vary across detectors and from event to event, and is typically estimated based on data around the time of the event. Given the pair (d, S_n), Bayes' theorem gives the posterior over parameters,

$$p(\theta|d, S_\mathrm{n}) = \frac{p(d|\theta, S_\mathrm{n})p(\theta)}{p(d|S_\mathrm{n})}, \tag{1}$$

and the task of GW parameter estimation is to draw samples from this distribution.

The growing rate of GW detections demands fast sampling techniques to accurately infer parameters for all observations. Standard inference codes are based on stochastic samplers [13, 14] that require millions of expensive likelihood evaluations for each event. However, deep-learning techniques have recently emerged as a promising tool [15–18]. In particular, the DINGO code often matches standard samplers in terms of accuracy, while being orders of magnitude faster [15, 19].

Artificial neural networks require that training and test data be independent draws from the same distribution. This is only the case when the measured GW data is consistent with the signal and noise models. Since the detector noise PSDs vary with time [20] (e.g., due to detector improvements or variations of the seismic noise), a network trained with an empiric PSD distribution at the beginning of a LIGO-Virgo-KAGRA (LVK) observing run can only be used for a limited time—once the PSDs change too much, the measured data become out-of-distribution (OOD). In section 2 we propose a parameterized latent variable model for detector noise PSDs that can be used to augment training data, enabling DINGO to better adapt to shifting distributions.

Even with accurate modeling of PSD variations one occasionally encounters OOD data due to transient noise artifacts (glitches) or inaccuracies of the signal models [21]. In such cases, the inference network could infer inaccurate results and one would have no means of knowing. In section 3 we propose to combine neural importance sampling with DINGO. This provides rapid verification and (in many cases) correction of results, and furthermore flags OOD data for further investigation. This extended abstract summarizes two recent studies [22, 23]; for further details and extensive empiric results on real data we refer to the original publications.

2 Adapting to noise distribution shifts

DINGO trains a conditional density estimator $q(\theta|d, S_\mathrm{n})$ parameterized with a normalizing flow [24–26] to estimate $p(\theta|d, S_\mathrm{n})$. A trained network can then be used to draw posterior samples given any d, S_n consistent with the training distributions. However, the PSD distribution $p(S_\mathrm{n})$ covering all *future* events is unavailable at the time of training. For real-time inference (especially after experimental upgrades) we therefore propose to use a *synthetic* PSD distribution $q(S_\mathrm{n})$ based on past PSDs and a one-shot observation from an upgraded detector [22].

Methods.—We define the synthetic PSD distribution $q(S_\mathrm{n})$ as a latent variable model

$$q(S_\mathrm{n}) = \int q(S_\mathrm{n}|z)q_z(z)dz, \tag{2}$$

where z refers to a set of latent variables that we describe below. We use domain knowledge to define $q(S_\mathrm{n}|z)$ explicitly and integrate causal knowledge about the data generating process. A PSD can be decomposed into broad-band noise b (estimated with variance σ^2) and a sum of spectral features $\sum_i s_i$ [27], which constitute the main factors of variation between and throughout observing runs. Thus, our model reads:

$$q(S_\mathrm{n}|z) = \mathcal{N}(b + \sum_{i=1}^{l} s_i, \sigma^2). \tag{3}$$

The latent representation of the broad-band noise is given by $y_1, \dots, y_k \in \mathbb{R}_+$ on a logarithmically distributed, fixed frequency grid $x_1, \dots, x_k$. b can then be reconstructed by interpolating pairs $(x_1, y_1), \dots, (x_k, y_k)$ using a cubic spline. Each spectral line s_i is represented by three parameters f_{0i}, A_i, Q_i modeling the center, amplitude and width of a truncated Cauchy distribution, respectively [27]. We segment the frequency space into l equally wide sub-intervals and model a single

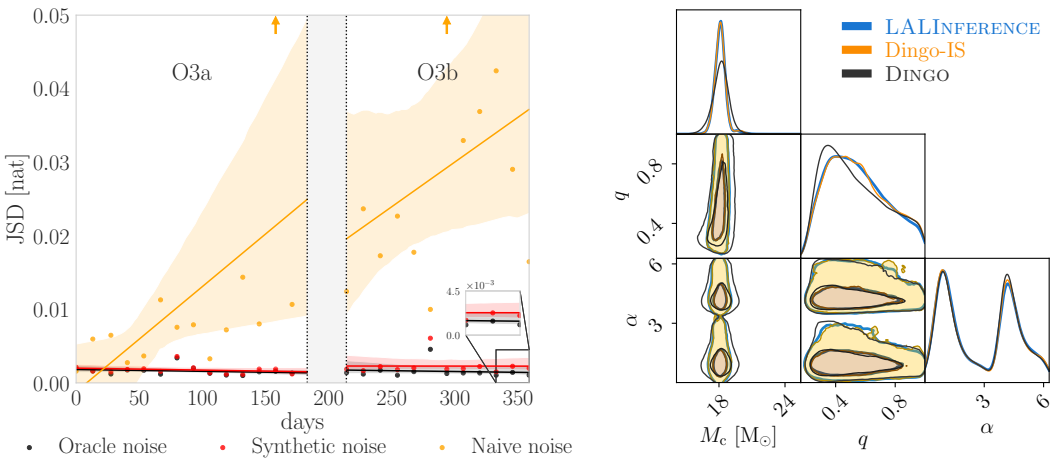

Figure 1: Left: Time development of DINGO performance since the beginning of O3 (Day 0) for models trained with different PSD datasets. With our *Synthetic* PSDs, we achieve comparable performance to the model trained with *Oracle* PSDs, despite only using a single O3 PSD. Right: Chirp mass (M_c), mass ratio (q) and sky position (α) parameters for GW151012. Even when initial DINGO results (black) deviate from LALINFERENCE-MCMC posteriors (blue), importance sampling leads to close agreement (orange).

spectral line in each of them.[2] In summary, the latent representation of a PSD is given by parameters $z = (y_1, \dots, y_k, f_{01}, A_1, Q_1, \dots f_{0l}, A_l, Q_l) \in \mathbb{R}^{k+3l}$.

We first project all PSDs from past observing runs onto their latent representation z using maximum-likelihood estimates. We then use these latent samples to fit Gaussian kernel density estimates (KDEs) over independent noise sources, modelling a disentangled latent distribution $q_z(z)$. The disentanglement enables meaningful interventions on the latent space, which we leverage to adjust the distribution for future data. Indeed, we first use a single PSD at the start of an observing run to rescale the mean of $q_z(y_1, \dots, y_k)$ to account for increased detector sensitivity. And then we broaden the latent distribution $q_z(z)$ by increasing the KDE bandwidth to account for uncertainty in the estimated model and fluctuations in the latent features that may occur over the course of an observing run. With this approach, we can prepare DINGO networks for an entire observing run without having access to future PSDs.

Results.—We evaluate our approach in a real-world setting by reanalyzing the third LVK observing run (O3). As a baseline, we train a DINGO model only with PSD data from the first four days of O3; this *Naive* model would be available shortly after the start of O3. We also train a DINGO model using a *Synthetic* PSD dataset from $q(S_n)$ based on past PSDs and scaled by the first PSD of O3. Finally, we train an *Oracle* model based on PSDs from all of O3 ($\approx$ 1 year), which upper bounds the potential performance of our noise model.

We first analyze a simulated GW signal injected into Gaussian noise drawn from a series of O3 PSDs. As a performance metric, we compare the inferred DINGO results to reference posteriors generated with DINGO-IS (see section 3) in terms of the mean Jensen-Shannon divergence (JSD) of the 1D marginals. Fig. 1 shows the time development of the DINGO performance. Indeed, the accuracy of the *Naive* baseline degrades with time. The *Oracle* model shows consistent performance throughout O3, however, in reality such a model could only be trained at the end of O3. DINGO models trained with the *Synthetic* PSD dataset almost match the *Oracle* performance, enabling accurate online analysis.

Finally, we analyze 37 real GW events from O3. Averaging across all events (except for an outlier event GW190517_055101) and parameters, our *Synthetic* PSD dataset achieves a mean JSD of $1.4 \cdot 10^{-3}$ nat, which is only slightly worse than with the *Oracle* PSDs ($1.2 \cdot 10^{-3}$ nat) and far

[2]This design choice restricts the number of spectral lines per segment to zero (via a vanishing amplitude parameter) or one. More than one spectral line cannot be modeled, but we found this to be irrelevant in practice if l is sufficiently large.

	Sample Efficiency	Mean JSD	Max JSD	$\log p(d)$
DINGO		2.2	7.2 (α)	-
DINGO-IS	28.8%	0.5	1.4 (d_L)	-15831.87 ± 0.01
BILBY	0.14%	1.8	4.0 ($d_{\rm L}$)	-15831.78 ± 0.10
DINGO		9.0	53.4 (M_c)	-
DINGO-IS	12.5%	0.7	2.2 (α)	-16412.88 ± 0.01
BILBY	0.16%	1.1	4.1 (α)	-16412.73 ± 0.09

Table 1: DINGO performance for GW150914 (upper block) and GW151012 (lower). The JSD quantifies the deviation to LALINFERENCE-MCMC, all values in 10^{-3} nat. The mean is taken across all parameters. Results with a maximum JSD $\leq 2 \times 10^{-3}$ nat are considered indistinguishable [32]. Here, maxima occur for right ascension α, luminosity distance $d_{\rm L}$, and chirp mass $M_{\rm c}$. For comparison, we also report results from BILBY-DYNESTY.

outperforms the *Naive* baseline ($2.7 \cdot 10^{-3}$ nat). This shows how the generative PSD model can be used to enhance the generalization capability of DINGO.

3 Neural importance sampling

Even accounting for shifts in detector PSDs, observed GW data can still be OOD if the noise is non-Gaussian or the real signal is inconsistent with signal models. We here propose to combine amortized neural posterior estimation (NPE) [28]—the method underlying DINGO—with importance sampling. This extension ("DINGO-IS" [23]) provides interpretable diagnostics to flag failure cases and asymptotically recovers the true posterior. Establishing this network-independent verification and correction mechanism makes DINGO substantially more reliable for dealing with real data.

Methods.—Given a set of n samples $\theta_i \sim q(\theta|d, S_{\rm n}) \equiv q(\theta|d)$, we assign each an importance weight $w_i = p(d|\theta_i)p(\theta_i)/q(\theta_i|d)$. These can be computed since the likelihood and the DINGO proposal are both tractable. If DINGO samples matched the true posterior perfectly, then $w_i =$ constant. More generally, the *effective sample size* is $n_{\rm eff} = (\sum_i w_i)^2 / \sum_i w_i^2$ and we can quantify the quality of the proposal with the *sample efficiency* $\epsilon = n_{\rm eff}/n$. Further, the Bayesian evidence $p(d)$ can be estimated as $p(d) = 1/n \sum_i w_i$. The variance of $p(d)$ scales with $1/n$ allowing for very precise estimates.

Importance sampling asymptotically recovers the exact posterior if the proposal distribution is mass-covering, i.e., $\text{supp}(p(d|\theta)p(\theta)) \subseteq \text{supp}(q(\theta|d))$. This is ensured by DINGO (and NPE in general) through minimizing the *forward* KL-divergence $D_{\rm KL}(p(\theta|d)||q(\theta|d))$ during training, which diverges unless $q(\theta|d)$ indeed covers the entire posterior probability space. This property is not guaranteed by classical stochastic samplers or by other machine learning methods optimizing different objectives (e.g., variational inference [24, 29]), so NPE is particularly well suited for importance sampling.

Although importance sampling requires likelihood evaluations at inference time, a high sample efficiency (due to a high-quality DINGO proposal) and parallelizability make it much faster than other likelihood-based methods. The combination of NPE and importance sampling is an amortized extension of neural importance sampling [30]. We anticipate this to be a useful approach beyond GW science to verify inference results of deep learning methods.

Results.—We first validate DINGO-IS on two real GW events (GW150915 and GW151012) by comparing against LALINFERENCE-MCMC [13], an established GW inference code. In particular, for GW151012, we find slight disagreement between DINGO and LALINFERENCE (Fig. 1 and Tab. 1). Compared to [15], we use the more complicated waveform model IMRPhenomXPHM [31] and a larger prior, so small DINGO inaccuracies are not surprising. The importance sampled DINGO result, however, is in excellent agreement with LALINFERENCE. For the log evidences, we find general agreement between the DINGO and BILBY-DYNESTY [14, 32, 33] estimates.

We also conduct a large scale study using two waveform models (IMRPhenomXPHM [31] and SEOBNRv4PHM [34]) analyzing 42 events from O3. Running stochastic samplers with SEOBNRv4PHM requires several months of computation per event, so a study of this scale is only feasible due to the superior speed of DINGO ($\approx$ 20 seconds per event) and DINGO-IS ($\approx$ 10 h per event for SEOBNRv4PHM; $<$ 1 h for IMRPhenomXPHM). Across all events, we find a median sample

efficiency of $\epsilon = 10.9\%$ for IMRPhenomXPHM and $\epsilon = 4.4\%$ for SEOBNRv4PHM. This is two orders of magnitudes larger than the sample efficiencies achieved by stochastic samplers, which are on the order of 0.1% (see Tab. 1). For most events, we find good agreement between DINGO and DINGO-IS, indicating high-quality inference results. Moreover, many events where DINGO performs poorly are known OOD events: nine events are known to suffer data quality issues (specifically, glitch artifacts), and for most of these, DINGO-IS has very low ϵ. Indeed, deep learning methods are known to perform poorly when the data does not match the training distribution. A low sample efficiency thus arises because the deep-learning extrapolation away from the training distribution differs from the posterior as defined by the specified prior times likelihood in this region. We rely on this discrepancy to identify potential OOD data. Likewise, adversarial attacks [35, 36] are also flagged with extremely low sample efficiency ($\epsilon \approx 0.01\%$) [23]. This showcases the use of importance sampling in conjunction with deep learning methods to flag potential failure cases such as OOD data.

Experimental details.—Here and in section 2 we use the setup from [15]. DINGO models are trained for ≈ 1 week on an A100 GPU and 32 CPUs. For DINGO-IS, we use 64 CPUs for the likelihood evaluations. We use `PyTorch` [37], `nflows` [38] and the Adam optimizer [39], which are all freely available under MIT or BSD license.

4 Conclusion

In this work, we proposed two approaches to deal with OOD data for flow-based GW parameter estimation, making DINGO into a more reliable and versatile parameter estimation tool. First, we introduced a probabilistic framework to model variations of the detector noise PSDs. We empirically demonstrated that this greatly enhances the generalization capabilities of DINGO to drifting detector noise distributions—enabling real-time analysis with a network trained at the beginning of an observing run. Second, we augmented DINGO with importance sampling (DINGO-IS). This provides useful diagnostics for detecting OOD data, and at the same time corrects potentially inaccurate inference results. DINGO-IS has a substantially larger sample efficiency than stochastic samplers and is fully parallelizable, resulting in great speed advantages. It also provides Bayesian evidence estimates with ten times greater precision, enabling detailed model comparison. Going forward, we expect these techniques to become key components for the integration of DINGO into production GW parameter estimation pipelines.

Broader impact statement

Our methods are primarily targeted at scientific applications, and we do not foresee direct applications which are ethically problematic. In the context of GW analysis, we hope this study helps to establish efficient deep learning methods in production pipelines. This could reduce the required amount of compute compared to standard inference methods, in particular when the rate of detections increases with more sensitive detectors in the future.

References

[1] J. Aasi et al. Advanced LIGO. *Class. Quant. Grav.*, 32:074001, 2015. `arXiv:1411.4547`, `doi:10.1088/0264-9381/32/7/074001`.

[2] F. Acernese et al. Advanced Virgo: a second-generation interferometric gravitational wave detector. *Class. Quant. Grav.*, 32(2):024001, 2015. `arXiv:1408.3978`, `doi:10.1088/0264-9381/32/2/024001`.

[3] Kentaro Somiya. Detector configuration of KAGRA: The Japanese cryogenic gravitational-wave detector. *Class. Quant. Grav.*, 29:124007, 2012. `arXiv:1111.7185`, `doi:10.1088/0264-9381/29/12/124007`.

[4] Yoichi Aso, Yuta Michimura, Kentaro Somiya, Masaki Ando, Osamu Miyakawa, Takanori Sekiguchi, Daisuke Tatsumi, and Hiroaki Yamamoto. Interferometer design of the KAGRA gravitational wave detector. *Phys. Rev. D*, 88(4):043007, 2013. `arXiv:1306.6747`, `doi:10.1103/PhysRevD.88.043007`.

[5] T. Akutsu et al. Overview of KAGRA: Detector design and construction history. *PTEP*, 2021(5):05A101, 2021. arXiv:2005.05574, doi:10.1093/ptep/ptaa125.

[6] 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. *Phys. Rev. X*, 9(3):031040, 2019. arXiv:1811.12907, doi:10.1103/PhysRevX.9.031040.

[7] R. Abbott et al. GWTC-2: Compact Binary Coalescences Observed by LIGO and Virgo During the First Half of the Third Observing Run. *Phys. Rev. X*, 11:021053, 2021. arXiv:2010.14527, doi:10.1103/PhysRevX.11.021053.

[8] R. Abbott et al. GWTC-3: Compact Binary Coalescences Observed by LIGO and Virgo During the Second Part of the Third Observing Run. 11 2021. arXiv:2111.03606.

[9] B. P. Abbott et al. GW170817: Measurements of neutron star radii and equation of state. *Phys. Rev. Lett.*, 121(16):161101, 2018. arXiv:1805.11581, doi:10.1103/PhysRevLett.121.161101.

[10] R. Abbott et al. Tests of General Relativity with GWTC-3. 12 2021. arXiv:2112.06861.

[11] R. Abbott et al. The population of merging compact binaries inferred using gravitational waves through GWTC-3. 11 2021. arXiv:2111.03634.

[12] R. Abbott et al. Constraints on the cosmic expansion history from GWTC-3. 11 2021. arXiv:2111.03604.

[13] J. Veitch et al. Parameter estimation for compact binaries with ground-based gravitational-wave observations using the LALInference software library. *Phys. Rev.*, D91(4):042003, 2015. arXiv:1409.7215, doi:10.1103/PhysRevD.91.042003.

[14] Gregory Ashton et al. BILBY: A user-friendly Bayesian inference library for gravitational-wave astronomy. *Astrophys. J. Suppl.*, 241(2):27, 2019. arXiv:1811.02042, doi:10.3847/1538-4365/ab06fc.

[15] Maximilian Dax, Stephen R. Green, Jonathan Gair, Jakob H. Macke, Alessandra Buonanno, and Bernhard Schölkopf. Real-Time Gravitational Wave Science with Neural Posterior Estimation. *Phys. Rev. Lett.*, 127(24):241103, 2021. arXiv:2106.12594, doi:10.1103/PhysRevLett.127.241103.

[16] Stephen R. Green, Christine Simpson, and Jonathan Gair. Gravitational-wave parameter estimation with autoregressive neural network flows. *Phys. Rev. D*, 102(10):104057, 2020. arXiv:2002.07656, doi:10.1103/PhysRevD.102.104057.

[17] Plamen G. Krastev, Kiranjyot Gill, V. Ashley Villar, and Edo Berger. Detection and Parameter Estimation of Gravitational Waves from Binary Neutron-Star Mergers in Real LIGO Data using Deep Learning. *Phys. Lett. B*, 815:136161, 2021. arXiv:2012.13101, doi:10.1016/j.physletb.2021.136161.

[18] Hunter Gabbard, Chris Messenger, Ik Siong Heng, Francesco Tonolini, and Roderick Murray-Smith. Bayesian parameter estimation using conditional variational autoencoders for gravitational-wave astronomy. *Nature Phys.*, 18(1):112–117, 2022. arXiv:1909.06296, doi:10.1038/s41567-021-01425-7.

[19] Maximilian Dax, Stephen R. Green, Jonathan Gair, Michael Deistler, Bernhard Schölkopf, and Jakob H. Macke. Group equivariant neural posterior estimation. 2022. arXiv:2111.13139.

[20] Benjamin P. Abbott et al. Sensitivity of the Advanced LIGO detectors at the beginning of gravitational wave astronomy. *Phys. Rev. D*, 93(11):112004, 2016. [Addendum: Phys.Rev.D 97, 059901 (2018)]. arXiv:1604.00439, doi:10.1103/PhysRevD.93.112004.

[21] Mark Hannam, Charlie Hoy, Jonathan E. Thompson, Stephen Fairhurst, and Vivien Raymond. Measurement of general-relativistic precession in a black-hole binary. 12 2021. arXiv:2112.11300.

[22] Jonas Wildberger, Maximilian Dax, Stephen R. Green, Jonathan Gair, Michael Pürrer, Jakob H. Macke, Alessandra Buonanno, and Bernhard Schölkopf. Adapting to noise distribution shifts in flow-based gravitational-wave inference. 11 2022. arXiv:2211.08801.

[23] Maximilian Dax, Stephen R. Green, Jonathan Gair, Michael Pürrer, Jonas Wildberger, Jakob H. Macke, Alessandra Buonanno, and Bernhard Schölkopf. Neural Importance Sampling for Rapid and Reliable Gravitational-Wave Inference. 2022. arXiv:2210.05686.

[24] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International Conference on Machine Learning*, pages 1530–1538, 2015. arXiv:1505.05770.

[25] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems*, pages 2338–2347, 2017. arXiv:1705.07057.

[26] Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. In *Advances in Neural Information Processing Systems*, pages 7509–7520, 2019. arXiv:1906.04032.

[27] Tyson Littenberg and Neil Cornish. Bayesline: Bayesian inference for spectral estimation of gravitational wave detector noise. *Physical Review D*, 91, 10 2014. doi:10.1103/PhysRevD.91.084034.

[28] George Papamakarios and Iain Murray. Fast ε-free inference of simulation models with bayesian conditional density estimation, 2016. arXiv:1605.06376.

[29] Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. In *Advances in neural information processing systems*, pages 4743–4751, 2016. arXiv:1606.04934.

[30] Thomas Müller, Brian McWilliams, Fabrice Rousselle, Markus Gross, and Jan Novák. Neural importance sampling. *ACM Transactions on Graphics (TOG)*, 38(5):1–19, 2019.

[31] Geraint Pratten et al. Computationally efficient models for the dominant and subdominant harmonic modes of precessing binary black holes. *Phys. Rev. D*, 103(10):104056, 2021. arXiv:2004.06503, doi:10.1103/PhysRevD.103.104056.

[32] 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. *Mon. Not. Roy. Astron. Soc.*, 499(3):3295–3319, 2020. arXiv:2006.00714, doi:10.1093/mnras/staa2850.

[33] Joshua S Speagle. dynesty: a dynamic nested sampling package for estimating bayesian posteriors and evidences. *Monthly Notices of the Royal Astronomical Society*, 493(3):3132–3158, Feb 2020. URL: http://dx.doi.org/10.1093/mnras/staa278, arXiv:1904.02180, doi:10.1093/mnras/staa278.

[34] Serguei Ossokine et al. Multipolar Effective-One-Body Waveforms for Precessing Binary Black Holes: Construction and Validation. *Phys. Rev. D*, 102(4):044055, 2020. arXiv:2004.09442, doi:10.1103/PhysRevD.102.044055.

[35] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014. arXiv:1312.6199.

[36] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015. arXiv:1412.6572.

[37] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In

H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[38] Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. nflows: normalizing flows in PyTorch, November 2020. doi:10.5281/zenodo.4296287.

[39] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. 2014. arXiv:1412.6980.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Section 2 and 3
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] Our code is currently reviewed internally by the LVK collaboration. We will make it publically available, once the review process is concluded.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See end of section 3, which also refers to [15].
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] Because of the computational cost of running the same experiment multiple time.s
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See end of section 3.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See end of section 3.
   (b) Did you mention the license of the assets? [Yes] See end of section 3.
   (c) Did you include any new assets either in the supplemental material or as a URL? [No] All assets will be included in the Python package that will soon be released. See question 3 (a).
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]