NESSAI: Improved nested sampling with normalising flows

Michael J. Williams [1] John Veitch [1] Chris Messenger [1]

[1]SUPA, School of Physics and Astronomy, University of Glasgow, Glasgow G12 8QQ, United Kingdom

University of Glasgow

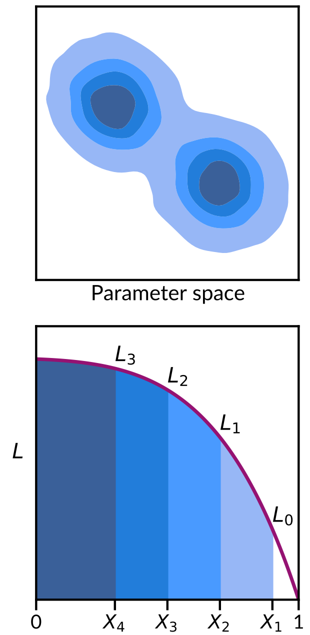
Bayesian Inference

In gravitational-wave inference, the physical parameters $\boldsymbol{\theta}$ that describe an astrophysical source, such as masses and spins, are inferred from the observed gravitational wave data d. This inference is carried out in Bayesian framework centred around Bayes' theorem [1]

$$p(\boldsymbol{\theta}|d, H) = \frac{p(d|\boldsymbol{\theta}, H)p(\boldsymbol{\theta}|H)}{p(d|H)},$$

where $p(d|\boldsymbol{\theta}$ is the likelihood, $p(\boldsymbol{\theta}|H)$ is the prior, $p(d|H)$ is the Bayesian evidence, often denoted Z, and $p(\boldsymbol{\theta}|d, H)$ is the posterior distribution. However, we typically cannot directly compute the posterior distribution and instead use stochastic algorithms such as Markov Chain Monte Carlo and Nested Sampling to sample the distribution [2].

Nested sampling



Nested sampling is a stochastic sampling algorithm proposed by John Skilling [3] in which the Bayesian evidence is rewritten as a one-dimensional integral in terms of the prior volume X

$$Z = \int_0^1 L(X)\mathrm{d}X.$$

This allows the integral to be approximated by considering an ordered sequence of decreasing points in prior volume, evaluating the likelihood and approximating the integral.

Figure 1: Visual representation of the core nested sampling idea, adapted from [3]. **Top:** an arbitrary likelihood surface with four likelihood contours. **Bottom:** the one-dimensional representation of the same parameter space in terms of the prior volume X, the curve is strictly monotonic and the area under the curve (the evidence) can therefore be approximated using, for example, the trapezoidal rule.

Challenges with nested sampling

The main bottleneck in a nested sampling algorithm is drawing new samples as the algorithm progresses. New samples must be:

- Independent and identically distributed (i.i.d),
- Distributed according to the prior,
- Have a greater likelihood than the point being replaced.

Current approaches typically rely upon either using random walks to propose new points or constructing multiple bounding distributions which can then be sampled from directly [4]. Gravitational-wave inference often requires combining both these methods to produce reliable results [5].

NESSAI: Nested Sampling with Artificial Intelligence

NESSAI (/ˈnɛsi/) [6] has been developed to address the aforementioned challenge of proposing new points. The standard proposal process is replaced with a machine learning based alternative which allows us to directly draw new samples from the current likelihood contour without using a random walk.

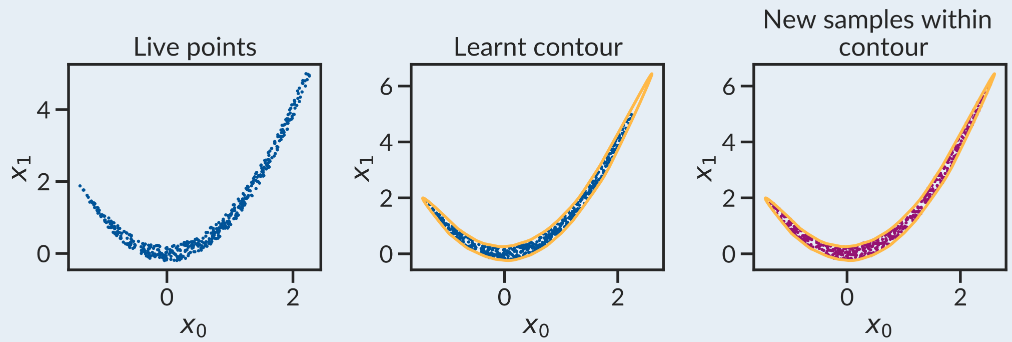


Figure 2: Visualisation of the core idea used in NESSAI. At a given iteration, the current set of live points are used to approximate the current likelihood contour using a normalising flow. New i.i.d samples can then be drawn from within the current likelihood contour.

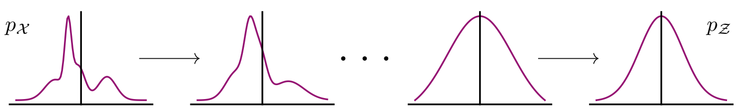
NESSAI: Improved nested sampling with normalising flows

Michael J. Williams [1] John Veitch [1] Chris Messenger [1]

[1]SUPA, School of Physics and Astronomy, University of Glasgow, Glasgow G12 8QQ, United Kingdom

Normalising flows

Normalising flows [7] are a type of generative machine learning algorithm and learn to map a complex distribution $p_{\mathcal{X}}$ to a simple latent distribution $p_{\mathcal{Z}}$ via a series of transforms.



This mapping is constructed to be invertible and to have a tractable Jacobian determinant. This allows an explicit expression for the distribution learnt by the normalising flow to be written down

$$p_{\mathcal{X}}(x) = p_{\mathcal{Z}}(f(x)) \left| \det \left(\frac{\partial f(x)}{\partial x^T} \right) \right| . \tag{1}$$

This contrasts with other generative machine learning algorithms, such as Variational Autoencoders and Generative Adversarial Networks, where the PDF of the learnt distribution is intractable.

Using normalising flows to construct contours

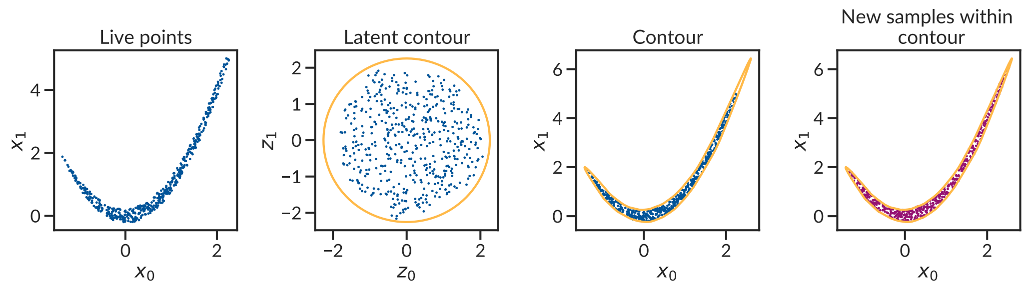
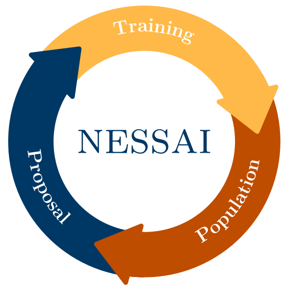


Figure 3: In NESSAI a normalising flow is trained on the current set of live points such that the trained flow maps the live points to an n-dimensional Gaussian latent space where a contour of equal likelihood is an $(n-1)$-sphere. Any point that is then mapped from the sampling space to the latent space will have a corresponding likelihood contour in the latent space and since the flow is invertible, we can map this contour back to the sampling space. To draw new samples we then sample from the n-dimensional Gaussian, apply the inverse mapping and perform rejection sampling to ensure the samples are distributed according to the prior.

Core algorithm



The core algorithm in NESSAI can be broken down in to three main stages:

1. Training: The normalising flow is trained on the current set of live points by minimising the KL-divergence.
2. Population: The current worst point is used to construct the likelihood contour and a pool of new candidate live points is populated.
3. Proposal: New live points are drawn from the pool and accepted if their log-likelihood is greater than the current worst point. This proceeds until the the pool is empty.

Challenges with NESSAI

There are some inherent challenges in NESSAI related to the choice of normalising flow:

- We must balance complexity and training time,
- There is a very limited amount of training data.

We find a normalising flow based on *affine coupling tranforms* [8] the best balance between complexity and cost of training. However, we note that certain features can be problematic with this type of flow and therefore introduce a series of reparameterisations which result in a distribution that is better suited to the flow.

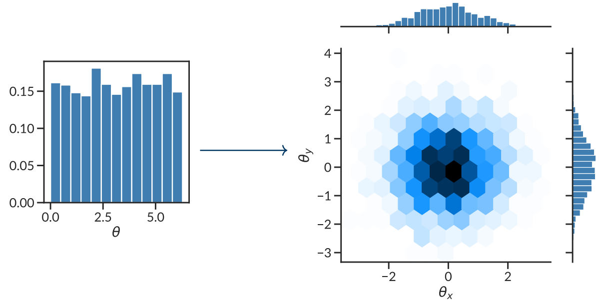


Figure 4: Example reparameterisation for an angular parameter θ that is periodic on $[0, 2\pi]$. The angle is transformed to Cartesian coordinates by introducing an auxiliary radial parameter that is drawn from a chi-distribution with two degrees of freedom. The resulting distribution is Gaussian and naturally includes the periodicity.

NESSAI: Improved nested sampling with normalising flows

Michael J. Williams [1] John Veitch [1] Chris Messenger [1]

[1]SUPA, School of Physics and Astronomy, University of Glasgow, Glasgow G12 8QQ, United Kingdom

University of Glasgow

Validating NESSAI

We validate NESSAI by analysing 128 simulated gravitational waves from binary black hole mergers and performing a series of tests, including comparing our results to those obtained with DYNESTY [9], another nested sampler that is often used for gravitational wave inference [10, 5].

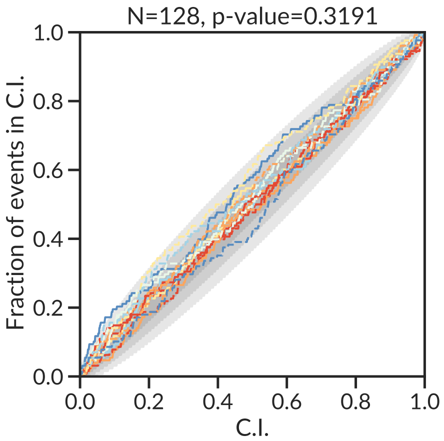


Figure 5: Probability-probability (P-P) plots produced using NESSAI. These test whether the correct fraction of injected events are recovered for a given confidence interval given the posterior distribution for each event.

Comparing NESSAI to DYNESTY

We also compare results obtained with NESSAI to those obtained with DYNESTY. We start by comparing the log-evidences obtained with each sampler and find good agreement (see below). We also use the results from DYNESTY to evaluate the number of likelihood evaluations and the total time required to reach convergence. We find that on average NESSAI requires 2.07 times fewer likelihood evaluations.

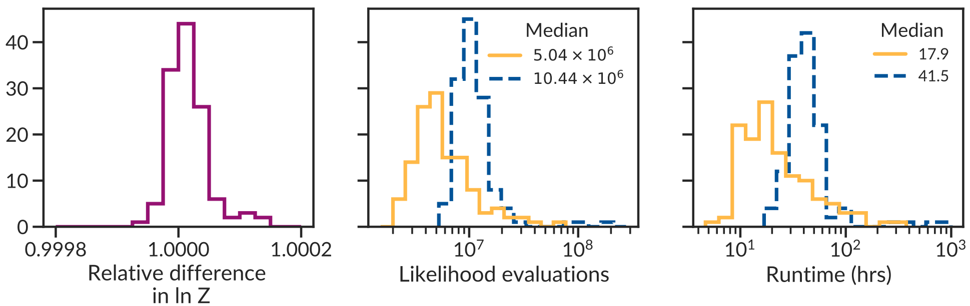


Figure 6: Relative difference in log-evidence, number of likelihood evaluations and runtime in hours for all 128 CBC injections when sampling with NESSAI (solid) and DYNESTY (dashed).

Example results

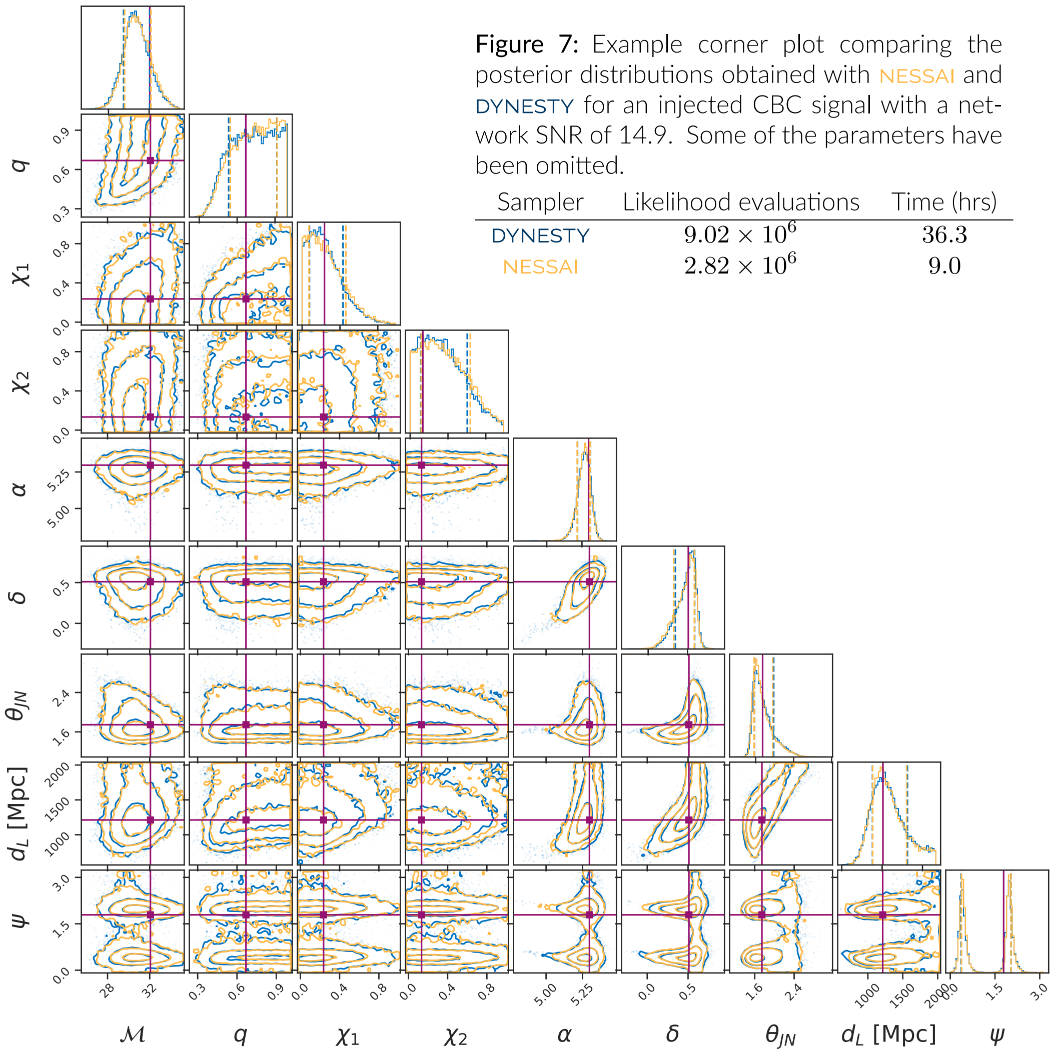


Figure 7: Example corner plot comparing the posterior distributions obtained with NESSAI and DYNESTY for an injected CBC signal with a network SNR of 14.9. Some of the parameters have been omitted.

Sampler	Likelihood evaluations	Time (hrs)
DYNESTY	9.02×10^6	36.3
NESSAI	2.82×10^6	9.0

NESSAI: Improved nested sampling with normalising flows

Michael J. Williams [1] John Veitch [1] Chris Messenger [1]

[1]SUPA, School of Physics and Astronomy, University of Glasgow, Glasgow G12 8QQ, United Kingdom

University of Glasgow

Parellelisation with NESSAI

An inherent benefit of the core algorithm in NESSAI is that there is a natural point at which the likelihood evaluation can be parallelised: after the population state when the pool of new live point has been populated. We implement this in NESSAI and show below how this can reduce the overall time to reach convergence. However, since the cost of training and population do not decrease, there is a lower limit to minimum time.

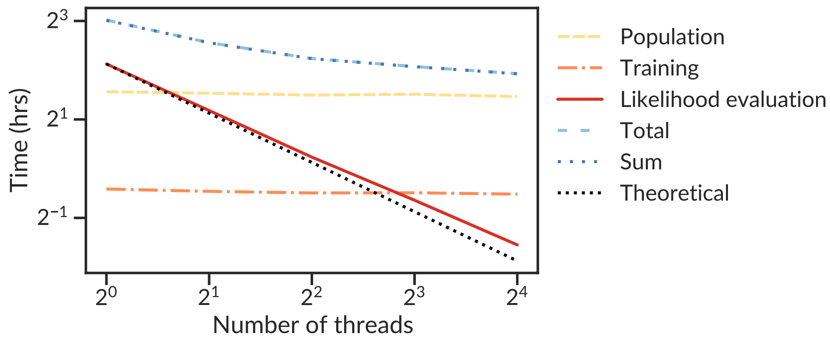
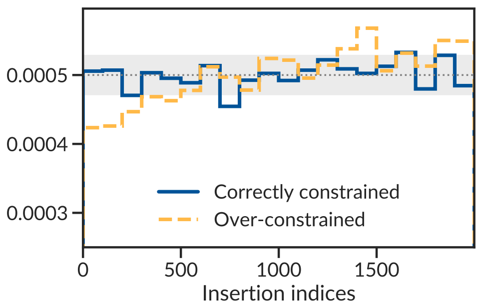


Figure 8: Comparison of total time spent on each part of the core algorithm in NESSAI as a function of the number of threads used for evaluating the likelihood.

Diagnostic plots

NESSAI also includes a range of plots to help diagnose problems with convergence. These allow the user to identify issues during sampling and adjust the samplers settings accordingly without the need to have other results for comparison.

Figure 9: Plot of the insertion indices [11] for two runs, this allows us to check that new live points have been inserted according to the ordered statistics we would expect, which should result in a uniform distribution.



Using NESSAI

Have a model that is slow to evaluate and taking a long time to sample? Then you should give **NESSAI** a try, it could save you a lot of time!

Why should I use **NESSAI**?

- NESSAI can speed up inference
- NESSAI can easily parallelise the likelihood evaluation
- NESSAI is not limited to applications in gravitational-wave inference
- NESSAI contains diagnostics that allows you to identify issues during sampling without repeating the analysis

How can I use **NESSAI**?

NESSAI is available to install via pip:

```
$ pip install nessai
```

It is also supported in BILBY [10]. Click the following button to try NESSAI out without any installation in your browser:

launch binder

For more details about NESSAI, see our paper [6], documentation and GitHub repository.

References

[1] B. P. Abbott et al. "The basic physics of the binary black hole merger GW150914". In: *Annalen Phys.* 529.1-2 (2017), p. 1600209.

[2] 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 (2015), p. 042003.

[3] J. Skilling. "Nested sampling for general Bayesian computation". In: *Bayesian Analysis* 1.4 (2006), pp. 833–859.

[4] J. Buchner. "Nested Sampling Methods". In: *arXiv e-prints*, arXiv:2101.09675 (2021), arXiv:2101.09675.

[5] 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.

[6] M. J. Williams et al. "Nested sampling with normalizing flows for gravitational-wave inference". In: *Phys. Rev. D* 103 (10 2021), p. 103006.

[7] G. Papamakarios et al. "Normalizing Flows for Probabilistic Modeling and Inference". In: *Journal of Machine Learning Research* 22.57 (2021), pp. 1–64.

[8] L. Dinh et al. "Density estimation using Real NVP". In: *arXiv e-prints*, arXiv:1605.08803 (2016), arXiv:1605.08803.

[9] J. S. Speagle. "DYNESTY: a dynamic nested sampling package for estimating Bayesian posteriors and evidences". In: *Monthly Notices of the Royal Astronomical Society* 493.3 (2020), pp. 3132–3158.

[10] G. Ashton et al. "BILBY: A User-friendly Bayesian Inference Library for Gravitational-wave Astronomy". In: *Astrophysical Journal Supplement Series* 241.2, 27 (2019), p. 27.

[11] A. Fowlie et al. "Nested sampling cross-checks using order statistics". In: *Monthly Notices of the Royal Astronomical Society* 497.4 (2020), pp. 5256–5263.