

NOTES ON THE PRACTICAL APPLICATION OF NESTED SAMPLING: MULTINEST, (NON)CONVERGENCE, AND RECTIFICATION

ALEXANDER J. DITTMANN^{1,*}

¹Department of Astronomy and Joint Space-Science Institute, University of Maryland, College Park, MD 20742-2421, USA
Version April 29, 2024

ABSTRACT

Nested sampling is a promising tool for Bayesian statistical analysis because it simultaneously performs parameter estimation and facilitates model comparison. MultiNest is one of the most popular nested sampling implementations, and has been applied to a wide variety of problems in the physical sciences. However, MultiNest results are frequently unreliable, and accompanying convergence tests are a *necessary* component of any analysis. Using simple, analytically tractable test problems, I illustrate how MultiNest (1) can produce systematically biased estimates of the Bayesian evidence, which are more significantly biased for problems of higher dimensionality; (2) can derive posterior estimates with errors on the order of $\sim 100\%$; (3) is more likely to *underestimate* the width of a credible interval than to overestimate it — to a minor degree for smooth problems, but much more so when sampling noisy likelihoods. Nevertheless, I show how MultiNest can be used to jump-start Markov chain Monte Carlo sampling or more rigorous nested sampling techniques, potentially accelerating more trustworthy measurements of posterior distributions and Bayesian evidences, and overcoming the challenge of Markov chain Monte Carlo initialization.

Subject headings: Bayesian statistics; Nested sampling; Markov chain Monte Carlo

1. INTRODUCTION

Fitting models to data and comparing the ability of different models to describe observations are integral to scientific practice. Nested sampling (Skilling 2004) is a technique that can estimate both posteriors and Bayesian evidences, which has led to its widespread popularity in the physical sciences (see Ashton et al. 2022; Buchner 2023, for recent reviews).

MultiNest (Feroz & Hobson 2008; Feroz et al. 2009) is arguably the most popular nested sampling implementation. It is relatively fast (see, for example, Figure 7 of Handley et al. (2015) or Figure 4 of Albert (2020)), and it can provide accurate results for simple problems with few parameters. MultiNest has been applied to X-ray astronomy (e.g., Buchner et al. 2014; Miller et al. 2019; Riley et al. 2019), the study of exoplanets (e.g. Kipping 2013; Morton et al. 2016; Tsirias et al. 2018), and cosmology (e.g. Planck Collaboration et al. 2014, 2016, 2020), in addition to many other fields.

However, numerous studies have found that MultiNest can perform poorly and unreliably, both in terms of inferring posterior distributions (e.g., Miller et al. 2019; Ih & Kempton 2021; Miller et al. 2021) and measuring Bayesian evidences (e.g., Albert 2020; Nelson et al. 2020; Lemos et al. 2023). One of the challenges faced when using MultiNest is deciding on values for its free parameters. The most important of these parameters are the number of live points (N_{live} , which determines how densely parameter space is sampled) and the sampling efficiency (ϵ , which controls how conservatively the algorithm explores parameter space). One can also tune the algorithm's stopping criterion through a tolerance parameter for the evidence (tol), but that is usually of little consequence to the algorithm's performance.¹

Unfortunately, the values for these parameters suggested by the MultiNest developers are often inadequate (see Section 4). For example, the README file included with the MultiNest source code suggests using $\epsilon = 0.8$ for parameter inference and $\epsilon = 0.3$ for evidence measurement.² The primary MultiNest paper states that “we found 400 and 50 active points to be adequate for evidence evaluation and parameter estimation respectively,” and suggests sampling efficiency values of 1 and 0.3 for parameter estimation and evidence evaluation (Feroz et al. 2009, Section 8). As shown in Section 4, these suggestions about sampling posteriors are actually reasonable for simple Gaussian distributions (as long as one can accept a $\sim 10\%$ error for each posterior distribution). However, even for ten dimensional Gaussian distributions these suggested parameter values are unable to accurately measure the Bayesian evidence.³

Many studies have investigated potential deficiencies in nested sampling algorithms, developed algorithmic improvements, and suggested methods for testing convergence. To give a few examples, Nelson et al. (2020); Albert (2020); Lemos et al. (2023) have noted systematic, and sometimes dimension- and sampling efficiency-dependent, biases in the evidence estimates produced by MultiNest. Buchner (2016a) developed a test that illustrated MultiNest's tendency to sample from an erroneously small prior volume, and developed an alternative algorithm for unbiased, albeit slower, sampling. Higson et al. (2019a) proposed some strategies to check convergence by cross-checking multiple analyses, and advocated

that the remainder of the evidence integral is negligible, and the algorithm terminates. Setting a smaller tolerance value continues iteration until less and less of the evidence integral remains.

² <https://github.com/farhanferoz/MultiNest/blob/cc616c17ed440fbff8d35937997c86256ed4d744/README.md>

³ It is worth noting that the authors of MultiNest have advocated that “it is always advisable to increase N_{live} , and reduce $[\epsilon]$ to check if the posteriors and evidence values are stable,” (Feroz et al. 2019) although many papers have used MultiNest without reporting such convergence tests.

*dittmann@umd.edu

¹ Unless one's goal is maximum-likelihood estimation: As MultiNest marches higher and higher toward the peak of the likelihood function, eventually the remaining prior volume is so minuscule

that when using MultiNest one should decrease ϵ and increase N_{live} to check convergence.

My aims here are to systematically study MultiNest’s convergence in terms of both the sampling efficiency and number of live points for problems of varying dimensionality, and to present strategies that can be applied to improve upon completed nested sampling analyses. Section 2 provides a brief overview of MultiNest and nested sampling, after which Section 3 describes some of my methodology. The results of my tests using MultiNest to sample smooth distributions are presented in Section 4, and my exploration of MultiNest’s performance on a noisy target distribution is presented in Section 5. Section 6 demonstrates a couple of methods that can be used to derive better posterior and evidence estimates, using an initial MultiNest (or other nested sampling) run to accelerate the process.

The scripts used to run each of these tests are hosted on GitHub.⁴ This work was conducted using MultiNest’s python bindings (Buchner 2016b) and MultiNest version 3.12.⁵

2. THE MULTINEST ALGORITHM

Nested sampling begins by drawing a set of N_{live} points from the prior. Each iteration of the algorithm proceeds by replacing the lowest-likelihood point with a new higher-likelihood point. If the higher-likelihood point is drawn uniformly from within the isolikelihood surface of the lowest-likelihood point, then the expectation value of the prior volume encompassed by the set of live points at iteration i will be $\exp(-i/N_{\text{live}})$.⁶ In this way, each sample can be associated with a fraction of the prior volume. The product of each point’s likelihood and associated prior volume then yields its posterior weight, and it is straightforward to integrate the likelihood over the prior to approximate the evidence.

If one were to perform simple rejection sampling, attempting to draw samples from the entire prior, the odds of drawing a point with a higher likelihood than those in the set of live points would become (exponentially) vanishing small. MultiNest’s innovative approach was to, rather than sample from the full prior, first fit a hyperellipsoid (or set of hyperellipsoids, if favored by a k-means algorithm) to the set of live points, to potentially expand those ellipsoids by a factor f , and to sample from within the resulting set of ellipsoids. At each iteration, the factor f is chosen such that $f = \max[1, \exp(-i/N_{\text{live}})V(E)^{-1}\epsilon^{-1}]$, where ϵ is the sampling efficiency parameter and $V(E)$ is the volume contained within minimum bounding ellipsoids. That is to say, if the volume of the ellipsoids bounding the set of live points is smaller than the expectation value of the remaining prior volume divided by the sampling efficiency, the ellipsoids are expanded so that their volume matches that target. See Section 5 of Feroz et al. (2009) for a more thorough presentation.

The sampling efficiency factor ϵ acts as a safeguard in the algorithm, making it more likely that new points are

actually drawn from within the intended likelihood contour. Unfortunately, when using $\epsilon = 0.3$, for example, MultiNest is still wont to sample from too small a volume of parameter space (Buchner 2016a). Notably, the amount of ellipsoid expansion in any given linear dimension at constant ϵ is much smaller in higher dimensions. This is why smaller efficiencies are required to produce unbiased results in higher dimensions and why using sampling efficiencies greater than unity is much more damaging for lower-dimension problems, as shown in Section 4.

3. TEST PROBLEMS AND METHODOLOGY

I have chosen three problems to test MultiNest’s robustness when applied to smooth problems: multidimensional versions of the normal distribution, the Rosenbrock distribution, and the log-Gamma distribution. In each test the integral of the target distribution can be calculated analytically, and after normalizing the log likelihood the evidence is simply given by the prior normalization.

The one-dimensional posterior distributions in the normal and log-Gamma distribution tests are trivial to calculate, so one can easily test the ability of each sampler to correctly recover the underlying posterior distribution. I took a practical approach to estimating errors in posterior recovery: I first computed a kernel-density estimate using the posterior samples drawn in each test, and then calculated the L_1 error of that kernel density estimate and the true underlying distribution. Explicitly, I calculated in each dimension (i)

$$L_{1,i} = \int |p(x_i) - \tilde{p}(x_i)| dx_i, \quad (1)$$

where $p(x_i)$ is the true marginalized posterior distribution and $\tilde{p}(x_i)$ is a kernel density estimate derived from the posterior samples. I used the scipy (Virtanen et al. 2020) implementation of the QUADPACK routine (Piessens et al. 1983) to evaluate the integral (1), and the default scipy Gaussian KDE implementation, which used Scott’s rule to select the bandwidth (Scott 2015). When reporting the results of various tests, I have presented the distribution of L_1 errors over all dimensions and independent trials.

In addition to the overall error in a given analysis, the deviations between the true posteriors and a set of posteriors estimates, it is useful to know if an algorithm tends to overestimate or underestimate the widths of credible intervals. To test this, in some cases I have calculated the % error of the $\pm 1\sigma$ ($\approx 68\%$) credible region, defined such that a negative % error means that MultiNest underestimated the width of a given credible interval.

Because each of the likelihood functions examined here can be integrated analytically, I have calculated evidence errors according to

$$\Delta \log Z = \log Z_{\text{measured}} - \log Z_{\text{true}}, \quad (2)$$

such that positive values of $\Delta \log Z$ indicate that MultiNest overestimated the evidence.

3.1. Test Problems

3.1.1. The Normal Distribution

⁴ <https://github.com/ajdittmann/multiNestNotes>

⁵ <https://github.com/farhanferoz/MultiNest>

⁶ Notably, this estimate, which I focus on here because of its usage in MultiNest (Feroz et al. 2009), is very slightly biased (e.g., Walter 2017; Salomone et al. 2018; Buchner 2023).

The simplest distribution I have considered here is a multivariate uncorrelated normal distribution. In general, most local maxima of realistic log likelihoods can be approximated by parabolas, so normal distributions will be relevant in at least some neighborhood about each log likelihood peak. However, in practice, many realistic log likelihoods deviate appreciably. The log-likelihood function in N dimensions is given by

$$\log L(\mathbf{x}) = \sum_i -\frac{1}{2} \left(\frac{x_i^2}{\sigma_i} \right)^2 - \log \sigma_i \sqrt{2\pi}. \quad (3)$$

One would hope that any Bayesian analysis tool is capable of sampling this distribution adequately. In practice, however, unless used with sufficiently rigorous “resolution settings,” MultiNest fails to recover the correct evidence in 10- or 20-dimensional test problems. For each problem, I selected standard deviations σ_i uniformly between 0.1 and 0.5 (inclusive), with a spacing dependent on the number of dimensions.

3.1.2. An Extended Rosenbrock Distribution

The Rosenbrock function in two dimensions is a standard test problem in numerical optimization, notable for its long, curving degeneracies, and strong correlations between parameters. There have been many attempts to extend the Rosenbrock function to higher dimensions, but I focus here only on the distribution presented in [Pagani et al. \(2019\)](#), which exhibits nontrivial correlations between each pair of parameters and has a known normalization.

The log likelihood for this distribution is given by, in $N = (n_1 - 1)n_2 + 1$ dimensions,

$$\log L(\mathbf{x}) = -a(x_1 - \mu)^2 - \sum_{j=1}^{n_2} \sum_{i=2}^{n_1} b_{j,i} (x_{j,i} - x_{j,i-1}^2)^2 - C, \quad (4)$$

where the normalization constant C is given by

$$C = \log \left(\frac{\sqrt{a} \prod \sqrt{b_{j,i}}}{\pi^{N/2}} \right). \quad (5)$$

I chose $\mu = 0$, $a = 2.0$, and each $b_{j,i}$ was chosen between 1 and 10 (inclusive) with a geometrically uniform spacing depending on the total problem dimensionality.

3.1.3. The Log-Gamma Distribution

Many distributions possess heavy tails. It is well-known that such distributions pose challenges to MultiNest, although the standard test involves four separate modes and combinations of log-gamma and normal distributions (e.g., [Beaujean & Caldwell 2013](#); [Buchner 2016a](#); [Feroz et al. 2019](#)). I have instead chosen a unimodal variation of this problem without any Gaussian components. The log likelihood for this test is given by

$$\log L(\mathbf{x}) = \sum_i c y_i - \exp y_i - \log \Gamma(c), \quad (6)$$

$$y_i = (x_i - \mu)/\sigma_i, \quad (7)$$

where Γ is the Gamma function, $c = 1$, $\mu = 2/3$, and scale parameters σ_i were spaced log-uniformly between 0.01 and 0.05 (inclusive).

3.1.4. A Noisy Normal Distribution

While attempting to understand why MultiNest might perform fairly reliable posterior inference for many standard test problems but systematically underestimate credible intervals to noticeable degrees when applied to astronomical data analyses (e.g., [Miller et al. 2019](#); [Ih & Kempton 2021](#); [Miller et al. 2021](#)), I considered the possibility that some sort of “noise” (perhaps some fairly high-frequency or stochastic variation in the likelihood as a function of one or more parameters, superimposed over some smooth trend) might be present in actual analyses but not the smooth test problems described above. To test this idea, I used the following log-likelihood function in N dimensions

$$\log L(\mathbf{x}) = \sum_i -\frac{1}{2} \left(\frac{x_i^2}{\sigma_i} \right)^2 - A f(x_i), \quad (8)$$

where $f(x)$ is a linear interpolant over a sequence of pseudorandom numbers drawn uniformly between -1 and 1, and A is a noise amplitude. For simplicity I used the same pseudorandom field in each dimension, which had between 250 and 1250 samples per σ_i . To be clear, I make no claim that this sort of noise is actually present in practical analyses: this is simply one test case where MultiNest performs similarly to the aforementioned applications to real data.

3.2. Testing Protocols

3.2.1. Smooth Test Problems

For each smooth toy problem (those described in Sections 3.1.1, 3.1.2, and 3.1.3, and analyzed in Section 4), I have tested a wide variety of resolution parameters to gauge the MultiNest’s performance. Concretely, I used $N_{\text{live}} \in \{100, 300, 1000, 3000, 10000\}$, for each N_{live} I used the sampling efficiencies $\epsilon \in \{10, 3, 1, 0.3, 0.1, 0.03, 0.01, 0.003, 0.001\}$, and I tested each $(N_{\text{live}}, \epsilon)$ combination 10 independent times (changing only the pseudorandom seed) for each of the three test problems. For each problem, I have also investigated three different dimensionalities (e.g. 5D, 10D, and 20D for the multidimensional normal distribution test), totaling 4050 analyses.

I have focused on only the standard MultiNest algorithm ([Feroz et al. 2009](#), version 3.12), without using importance nested sampling ([Feroz et al. 2019](#)). Since this variation does not alter MultiNest’s parameter exploration strategy, it would not ameliorate any of the cases where MultiNest’s posterior estimates severely erred. Importance nested sampling might improve performance in some of the cases where the posterior estimates were acceptable and the evidence was simply overestimated, although it is far from panacean (e.g., [Nelson et al. 2020](#)), sometimes leading to the confident exclusion of the true evidence ([Buchner 2016a](#)).

Although each problem was unimodal, I allowed MultiNest to perform mode separation if it identified multiple modes, since in practice one rarely knows a priori how many modes are present in a given problem. I performed a handful of log-gamma tests with mode separation deactivated and observed no significant changes in the results. I set MultiNest’s evidence tolerance parameter to 10^{-5} in each analysis to avoid the possibility that param-

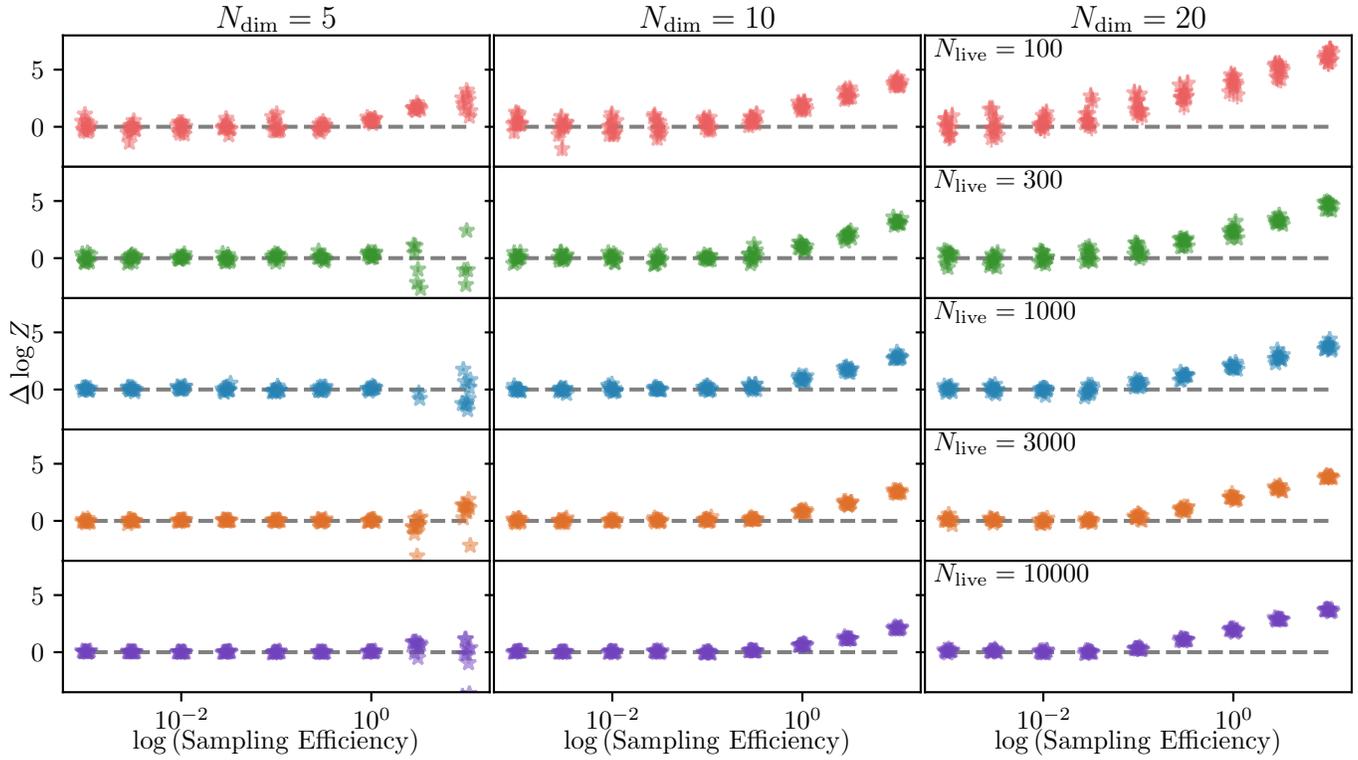


FIG. 1.— Errors in MultiNest-derived evidence values from the multivariate normal distribution tests. Each column plots results from distributions of different dimensionalities, each row plots results from tests using different numbers of live points, and results in each subplot are plotted as functions of sampling efficiency. Significant departures from the gray line, which marks an evidence error of zero, indicate cases in which MultiNest failed to measure the evidence accurately. Error bars indicate the MultiNest-estimated uncertainties, although these are often dwarfed by the actual errors.

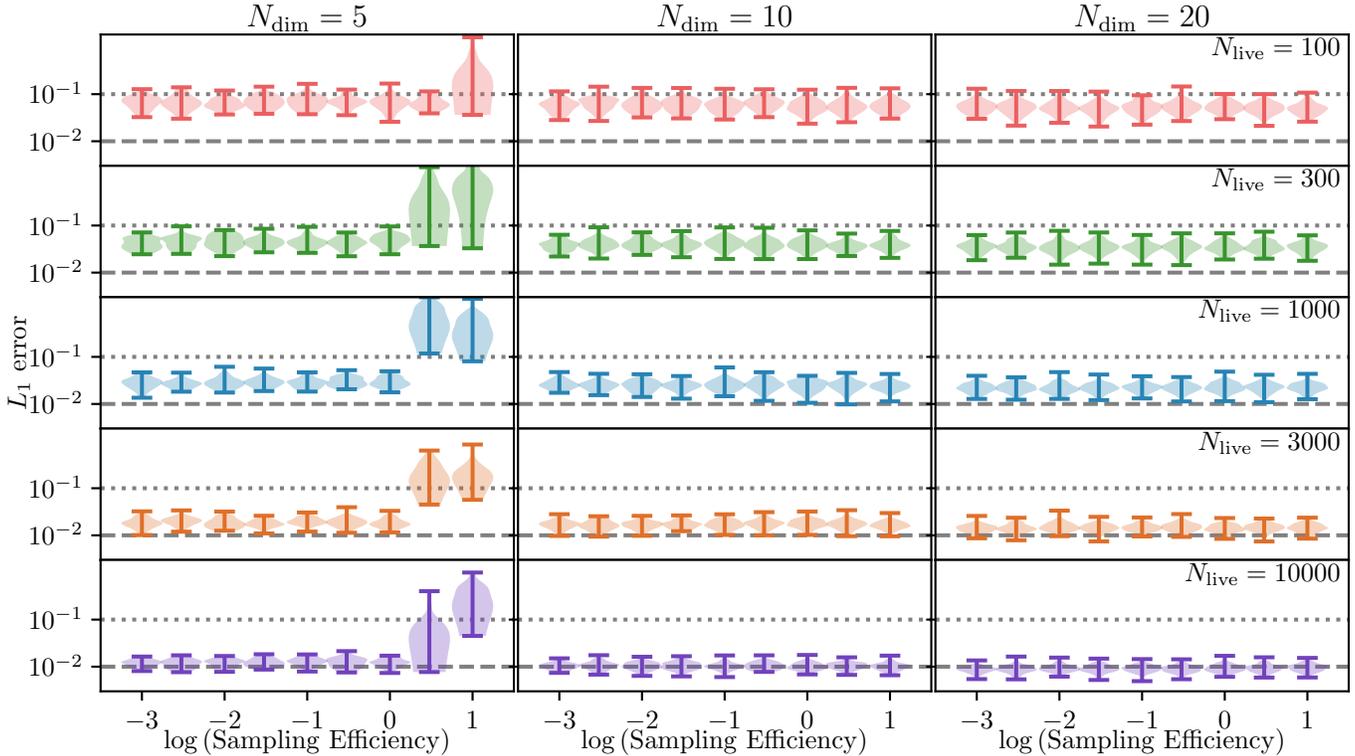


FIG. 2.— Errors in MultiNest-derived posteriors from the multivariate normal distribution tests. The layout is the same as in Figure 1. Dashed and dotted gray lines mark L_1 errors of 0.01 and 0.1 respectively.

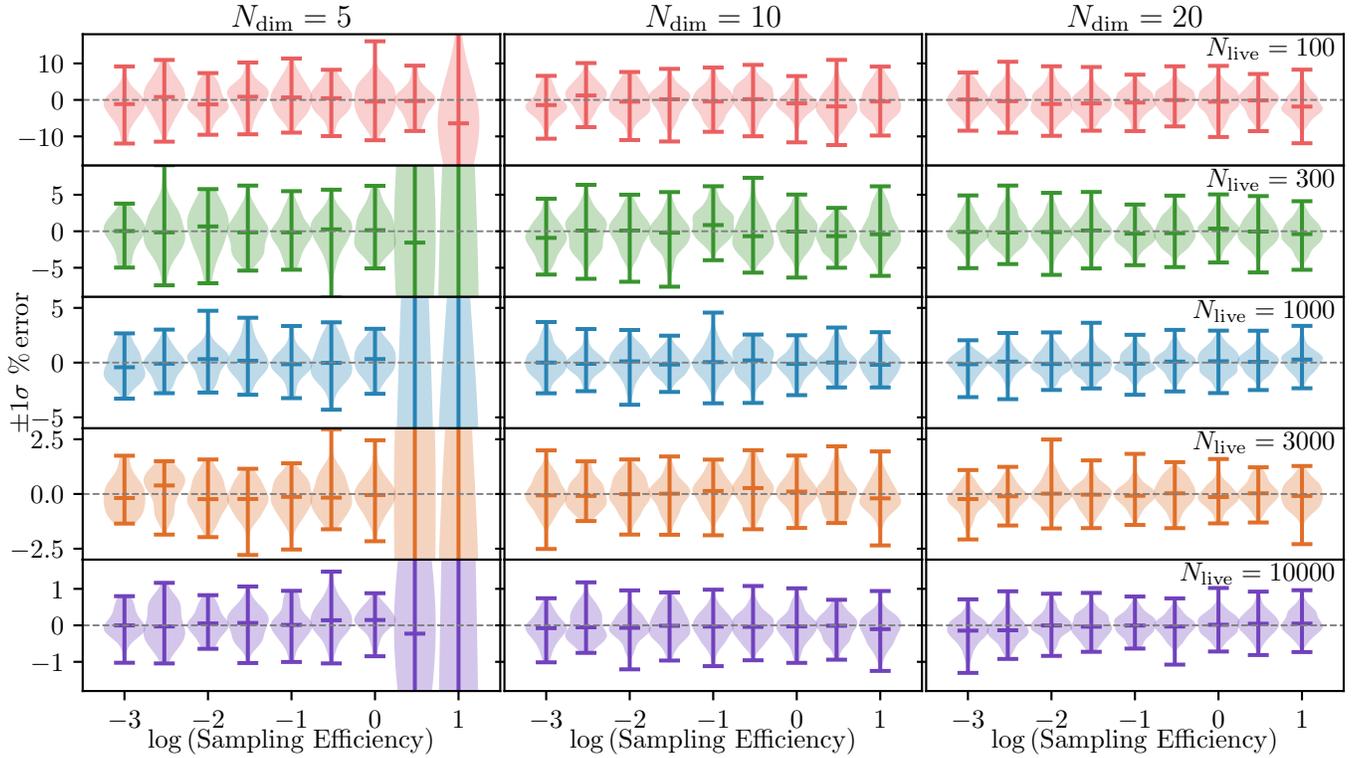


FIG. 3.— Errors in the widths of the MultiNest-derived $\pm 1\sigma$ credible regions from the multivariate normal distribution tests. The horizontal lines in each violin plot denote the minimum, median, and maximum values.

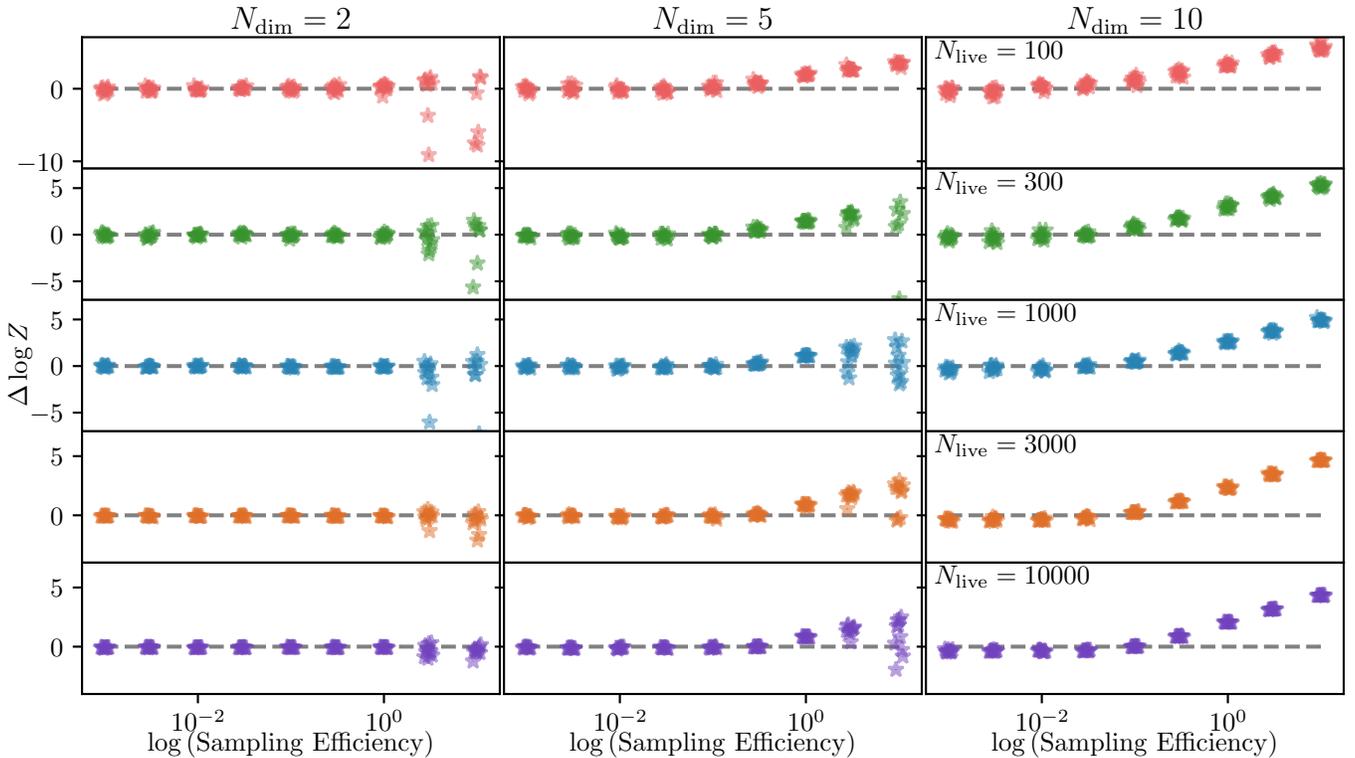


FIG. 4.— Errors in MultiNest-derived evidence values from the generalized Rosenbrock distribution tests. Each column plots results from distributions of different dimensionalities, each row plots results from tests using different numbers of live points, and results in each subplot are plotted as functions of sampling efficiency. Error bars again indicate the MultiNest-estimated uncertainties, although these are frequently orders of magnitude smaller than the actual errors.

eter space might not be thoroughly explored because the tolerance was too large, although I also repeated ~ 100 normal and log-gamma tests using a tolerance parameter of 0.25 and found no appreciable differences in the quality of the results.

3.2.2. A Noisy Test Problem

For the test introduced in Section 3.1.4, developed in an attempt to understand a possible cause for MultiNest systematically underestimating credible region widths (e.g., Miller et al. 2019; Ih & Kemp-ton 2021) in a sampling-efficiency-dependent way (e.g., Miller et al. 2021),⁷ I focused instead on MultiNest parameters closer to those used in scientific applications. Specifically, I used $N_{\text{live}} \in \{300, 1000, 3000, 10000\}$, for each N_{live} I have used the sampling efficiencies $\epsilon \in \{0.3, 0.1, 0.03, 0.01\}$, and I have tested each $(N_{\text{live}}, \epsilon)$ combination 10 independent times. For each of these, I have saved MultiNest outputs at two evidence tolerance values, $\text{tol} \in \{0.5, 0.01\}$, waiting for the first series to complete and then re-starting each with a lower tolerance. For succinctness, I have only reported in Section 5 results from a 10D test case.

4. MULTINEST IN ACTION: SMOOTH TEST PROBLEMS

4.1. Normal Distributions

- When using too large a sampling efficiency, MultiNest overestimates the evidence, more so in higher dimensions.
- For evidence measurement in lower-dimension (5) problems, $\epsilon \lesssim 0.3$ may be sufficient at $N_{\text{live}} = 10^2$, and $\epsilon \lesssim 1$ may be sufficient at $N_{\text{live}} = 10^4$.
- For higher-dimension (20) problems, $\epsilon \lesssim 0.01$ may be sufficient at $N_{\text{live}} = 10^2$, and $\epsilon \lesssim 0.03$ may be sufficient at $N_{\text{live}} = 10^4$.
- Posterior accuracy is effectively independent of ϵ (for $\epsilon \leq 1$).
- Posterior accuracy improves at close to, but slightly more slowly than, $\propto N_{\text{live}}^{-1/2}$.
- MultiNest is slightly biased towards underestimating credible region widths.

These observations are based on the results presented in Figures 1 and 2. The former figure presents the individual evidence estimates from each multivariate normal analysis, along with the uncertainties reported by MultiNest, which are plotted as error bars. In many cases the reported uncertainties are orders of magnitude smaller than the actual error of the results. The later figure plots the distribution of L_1 errors of the 1D posteriors in each analysis.

⁷ Most quantitatively, in Miller et al. (2021), MultiNest underestimated the neutron star equatorial radius posterior width by a factor of ~ 0.67 when used with $(N_{\text{live}} = 1000, \epsilon = 0.01)$, a factor of ~ 0.79 when used with $(N_{\text{live}} = 3000, \epsilon = 0.01)$, and even smaller factors when used with larger sampling efficiencies.

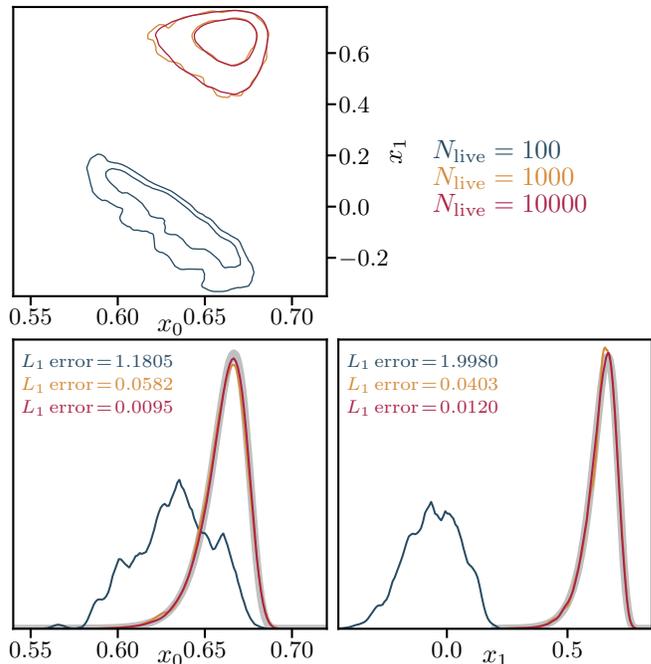


FIG. 5.— Posterior distributions for a 2D log-gamma distribution test using $\epsilon = 1$. Blue, orange, and red curves plot results derived using 10^2 , 10^3 , and 10^4 live points, respectively. The true one-dimensional marginalized posteriors are plotted using solid gray lines. These highlight a run that largely missed the mark, resulting in an L_1 error greater than 1, and two that sampled accurately to varying degrees of precision. The contours in the upper panel display the two-dimensional 1σ and 2σ minimum-area credible regions. The distributions are plotted here using averaged shifted histograms (Scott 1985).

Figure 3 displays the errors in the width of the $\pm 1\sigma$ credible intervals for each of the normal distribution analyses, collated similarly to the L_1 error measurements. Out of the 15750 posterior distributions inferred in these tests, MultiNest produced an erroneously narrow posterior distribution on 8370 instances. Although MultiNest only underestimated the width of posterior credible regions $\sim 53\%$ of the time, the odds of this many or more low estimates occurring in a sample of unbiased posterior estimates (that is, with widths overestimated as often as they are underestimated) are on the order of $\sim 10^{-13}\%$. Restricting the results to the 15477 posterior inferences that had L_1 errors less than 0.2, excluding the cases where MultiNest estimated the posteriors particularly poorly, MultiNest underestimated the width of the credible region in 8168 cases. With an unbiased sampler the odds of getting that many (or more) low rolls is a much more reasonable $\sim 2 \times 10^{-12}\%$. Although not by particularly harmful amounts for this simple problem, MultiNest appears to be systematically biased towards reporting erroneously narrow credible regions.

4.2. Abnormal Distributions I: Bananas (Generalized Rosenbrock Distributions)

- MultiNest is able to adequately sample the standard 2D Rosenbrock distribution with efficiencies of $\epsilon \leq 0.3 - 1$ depending on the number of live points.
- MultiNest struggles in higher dimensions more so than when sampling simpler normal distributions

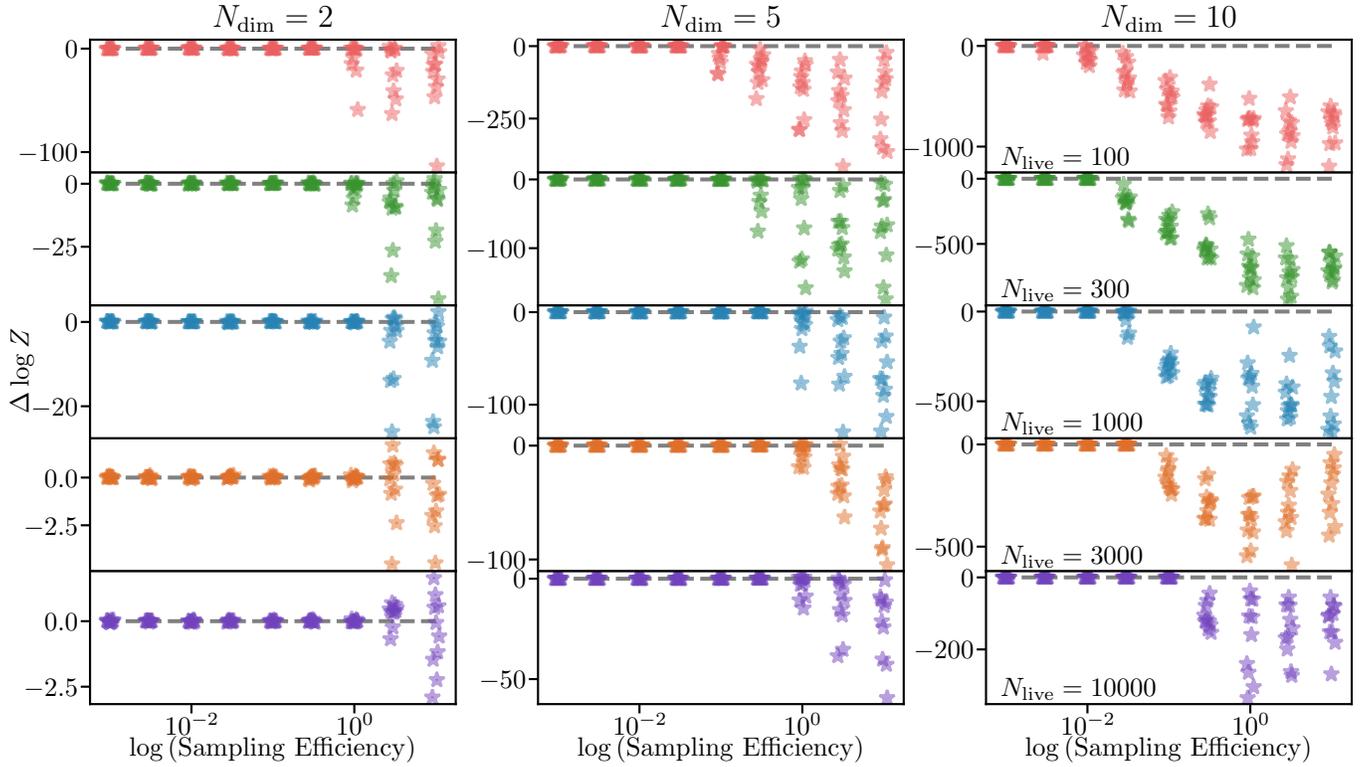


FIG. 6.— MultiNest evidence errors for the log-gamma problem, as functions of dimensionality, sampling efficiency, and the number of live points. Especially in higher numbers of dimensions, small sampling efficiencies and large number of live points are necessary for MultiNest to derive accurate evidence estimates.

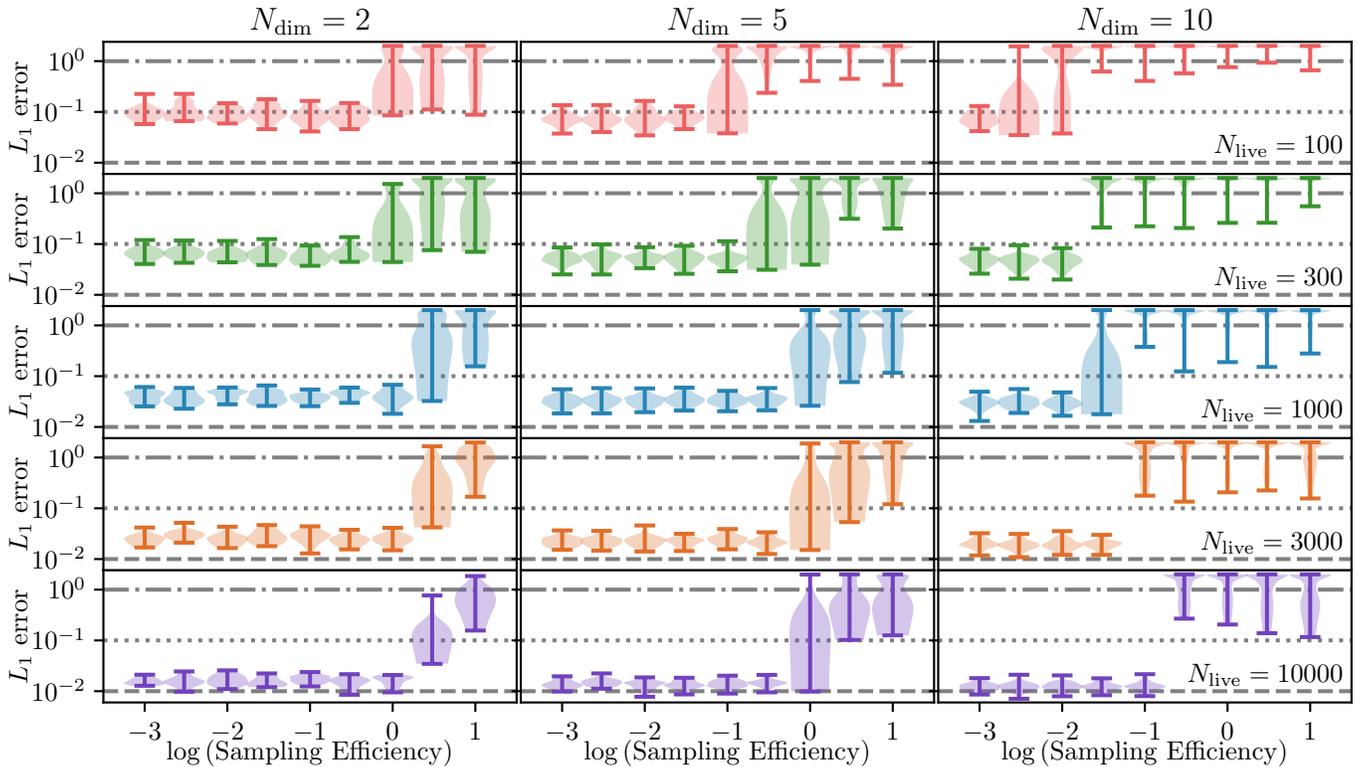


FIG. 7.— L_1 errors in MultiNest-derived posteriors from my multivariate log-gamma tests. Especially in higher numbers of dimensions, small sampling efficiencies and large number of live points are necessary for MultiNest to derive posterior estimates with that are not in complete error.

of the same dimensionality, requiring $\epsilon \leq 0.01$ at $N_{\text{live}} \sim 100$ and $\epsilon \leq 0.1$ for $N_{\text{live}} \sim 10^4$ when sampling a 10D distribution.

The generalized Rosenbrock distribution evidence estimates are presented in Figure 4. Overall, I have found that MultiNest is able to sample from generalized Rosenbrock distributions without too much additional difficulty compared to an uncorrelated multivariate normal distributions. For a given dimensionality, smaller sampling efficiency values are required to achieve converged results, in comparison to sampling simpler normal distributions.

4.3. Abnormal Distributions II: Log-Gammas

- For the 2-dimensional problem, efficiencies between 0.3 and 1 may be sufficient to derive reasonable posteriors and estimates of the evidence, depending on the number of live points (see Figure 5).
- For higher-dimension problems, efficiencies between 0.001 and 0.1 can be necessary to derive posterior and evidence estimates that are not in severe error.
- Higher-dimension problems require lower efficiencies at a fixed number of live points.
- Errors in MultiNest’s log evidence estimates can range from 100s to 1000s when applying the developer-suggested efficiency settings to this problem.
- MultiNest is slightly biased towards underestimating the widths of posteriors.

The above observations are largely drawn from Figures 6 and 7. The former figure presents the individual evidence estimates from each multivariate normal analysis; although error bars are technically included, the estimated errors are less than unity and completely dwarfed by the systematic errors present in most of the analyses. In most of the cases where MultiNest does a very poor job of estimating the evidence, it also derives very incorrect posterior estimates, reporting results with errors on the order of 100%. If this holds in general, consistent evidence measurements might suggest fairly trustworthy posterior inferences.

A more worrying aspect of Figures 6 and 7 is the breadth over which MultiNest can report $\sim 100\%$ erroneous results. Let us take the 10-dimensional problem as an example. Based on the recommendations in the MultiNest documentation, one might reasonably start with 1000 live points and an efficiency of 0.3, which would almost certainly lead to a log-evidence error on the order of ~ 500 . One may need to decrease the sampling efficiency by a factor of ~ 300 holding the number of live points fixed, or by a factor of $\sim 3 - 100$ while increasing the number of live points by factors of 10-3 to derive accurate results. However, one can not know this a priori, so diligent convergence testing should always be practiced, until the evidence estimates and posteriors converge.

Again measuring the widths of the analytical and measured $\pm 1\sigma$ credible intervals and restricting the samples

to only those with L_1 errors less than 0.2, 2110 out of 4008 estimated credible regions were too narrow. This is again quite improbable (if MultiNest underestimated the width of credible regions as often as it overestimated them, there would be a $\sim 0.04\%$ chance of this many or more of that sample being too narrow), although the evidence is not as strong given the smaller data set. However, looking at the full set of posterior estimates, I found that 3350 out of 7650 credible interval widths were underestimated. Here there is a silver lining: when MultiNest does especially poorly, credible interval widths tend to be overestimated, at least for this problem.

5. MULTINEST IN ACTION: SAMPLING NOISY LIKELIHOODS

Naturally, as the noise amplitude A in Equation (8) goes to zero, the results should approach those of Section 4.1. I tested $A \in \{0.5, 0.05, 0.005\}$, although for this test the differences between $A = 0.05$ and $A = 0.005$ were negligible and predictable. The results from these tests are collected in Figure 8, which shows the errors in credible region widths as functions of N_{live} , ϵ , A , and tol . When A is small, the results are effectively unbiased, as expected. However, when the noise amplitude is large, MultiNest’s posterior estimates can be systematically biased towards being erroneously narrow.

More specifically, at larger values of the sampling efficiency (such as $\epsilon = 0.3$, which is still less than half that recommended by the MultiNest documentation for parameter estimation), MultiNest produces systematically narrow posterior estimates. These biases do not disappear as more live points are used in the analyses; rather, as live points are added the run-to-run variation seems to decrease while the median error remains roughly constant between runs. The biases are significantly reduced at $\epsilon = 0.1$, and virtually disappear for this problem by $\epsilon \leq 0.03$. Intriguingly, for this particular problem, results can become slightly more biased as tol , the evidence tolerance used by MultiNest as a stopping condition, is decreased. Presumably, because the actual likelihood surface is nonsmooth, MultiNest’s bounding ellipsoids have a tendency to systematically underestimate the actual isolikelihood surface at a given iteration. Since the noise introduced in this test is a perturbation overtop a general trend, once MultiNest’s bounding hyperellipsoids are enlarged sufficiently at each stage they are able to derive much less biased samples. This tendency should hold for fairly generic, even smooth, likelihoods, but the this particular likelihood function seems to exacerbate the issue.

6. STRATEGIES FOR IMPROVEMENT

For problems like those analyzed in the present study, which require only a modest personal computer, a sound strategy, besides conducting rigorous convergence tests, is to use a more robust nested sampling tool. A number of other tools seem to meet the criterion of being less prone to biases than MultiNest, including POLYCHORD (Handley et al. 2015; Lemos et al. 2023), JAXNS (Albert 2020), dynesty (Speagle 2020), or UltraNest (Buchner 2021).⁸ If a given likelihood function is compatible with the JAX framework (Bradbury et al. 2018), JAXNS

⁸ Empirical evidence points to slice sampling implementations being less prone to producing biased evidence estimates than the

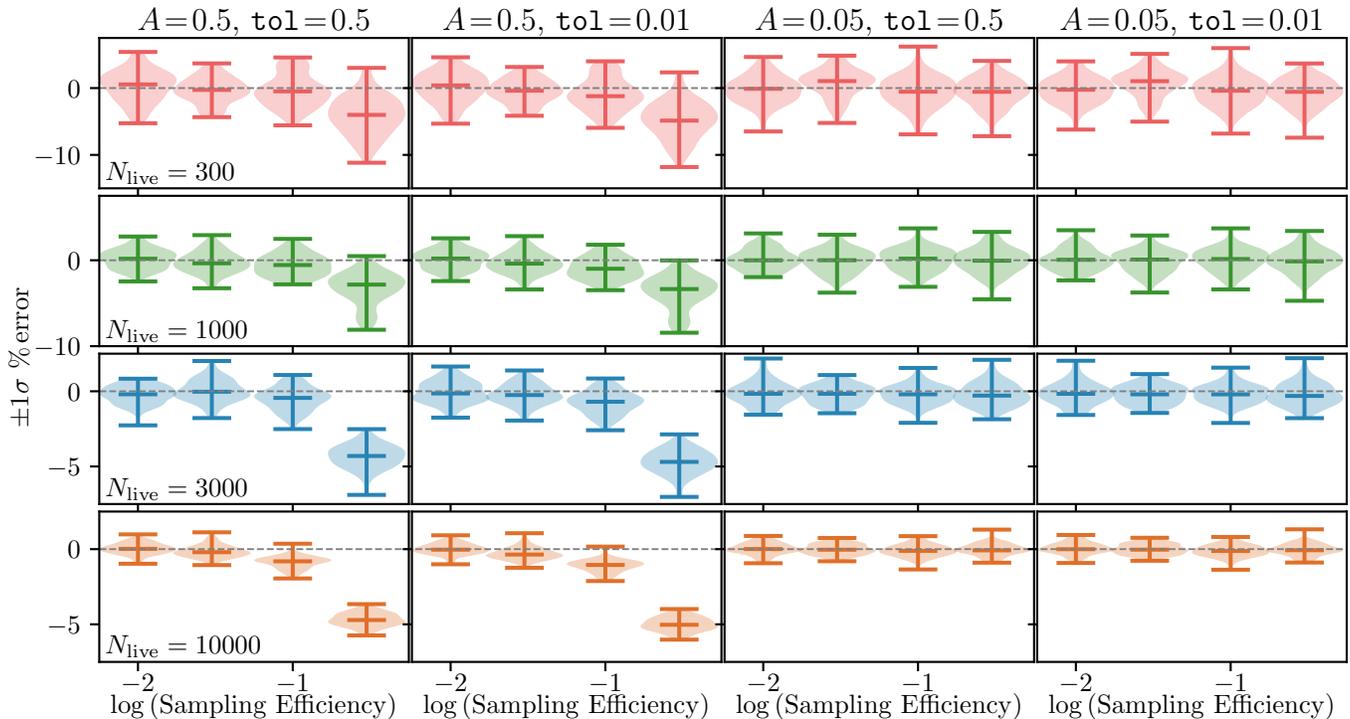


FIG. 8.— Errors in the widths of the MultiNest-derived $\pm 1\sigma$ credible regions from the noisy normal distribution test. The horizontal lines in each violin plot denote the minimum, median, and maximum values, and the gray dotted line indicates an error of zero. At larger noise amplitudes A and higher sampling efficiency values (e.g. $\epsilon = 0.3$), MultiNest produces posterior estimates that are systematically narrow (by $\sim 5 - 10\%$ for this problem). These biases remain as the number of live points is increased, but can be ameliorated by a sufficiently small value of the sampling efficiency.

might be particularly favorable, and significantly faster than the alternatives (Albert 2020).⁹ Otherwise, MultiNest can remain an order of magnitude or so faster than more robust sampling algorithms. As shown above, one might need to improve the resolution (and possibly expense) of a fiducial MultiNest run by orders of magnitude for the results to be worth trusting, but both this prospect and that of using a slower sampling strategy can be difficult pills to swallow when committing thousands or millions of core hours to a Bayesian analysis on a supercomputer.

Generally speaking, it is possible to use an initial (presumably *relatively* quick) MultiNest analysis to accelerate other methods more capable of accurately estimating evidences and posteriors. One approach is to use the initial nested sampling analysis to initialize a Markov-chain Monte Carlo (MCMC) analysis (e.g., Miller et al. 2019, 2021; Wolfe et al. 2023), where as long as the initial nested sampling analysis was able to correctly identify the most significant mode within a given parameter space the MCMC analysis can be expected to draw un-

biased samples from the posteriors from the target distribution. If the initial nested sampling analysis was able to adequately sample the posterior distribution, the MCMC sampler will be able to start drawing unbiased samples almost instantly. However, if the nested sampling algorithm does a poor job of estimating posteriors this method can require a long “burn-in” period, and standard MCMC techniques can not reliably estimate the evidence.¹⁰ Another related approach, although I have not yet tested this personally, could be to initialize a population Monte Carlo analysis (e.g., Cappé et al. 2004; Cappé et al. 2008), similar to the description in Beaujean & Caldwell (2013), which could also produce evidence estimates via importance sampling.

Another strategy is to use a preliminary MultiNest analysis to jump-start a nested sampling analysis with a slower but more robust nested sampling strategy. Qualitatively, information about the posterior can be used to reduce the time spent by a nested sampling analysis compressing from the prior to the posterior (Petrosyan & Handley 2022). This strategy can be used to refine a dubious initial analysis, or to accelerate the analysis a problem very similar to one already analyzed (for example, one adding a few new observations to an existing data set). Here, I’ve used the implementation of “warm

rejection sampling-based MultiNest (see, for example, Lemos et al. (2023) and Figure 4 of Albert (2020), although in that case other parameters must be tested to ensure convergence); other packages such as UltraNest implement more robust bounding regions for rejection sampling (e.g., Buchner 2016a, 2021). Furthermore, packages such as dynesty and dyPolyChord implement dynamic nested sampling (Higson et al. 2019b), which can target both posterior and evidence accuracy in addition to providing the capability of further refining an initial nested sampling analysis; UltraNest also allows further refinements following an initial run, and targeting both posterior and evidence accuracy.

⁹ UltraNest also provides vectorized step samplers that can take advantage of JAX-based or otherwise GPU-parallelized likelihoods.

¹⁰ Parallel-tempered MCMC samples may be used to estimate the evidence as well (e.g., Gelman & Meng 1998; Vousden et al. 2016), but estimates using only samples of the posterior typically have *infinite* variance (Newton & Raftery 1994) and are therefore useless. See Wolfe et al. (2023) for a description of how samples from a single nested sampling analysis can be used to initialize a parallel-tempered MCMC analysis.

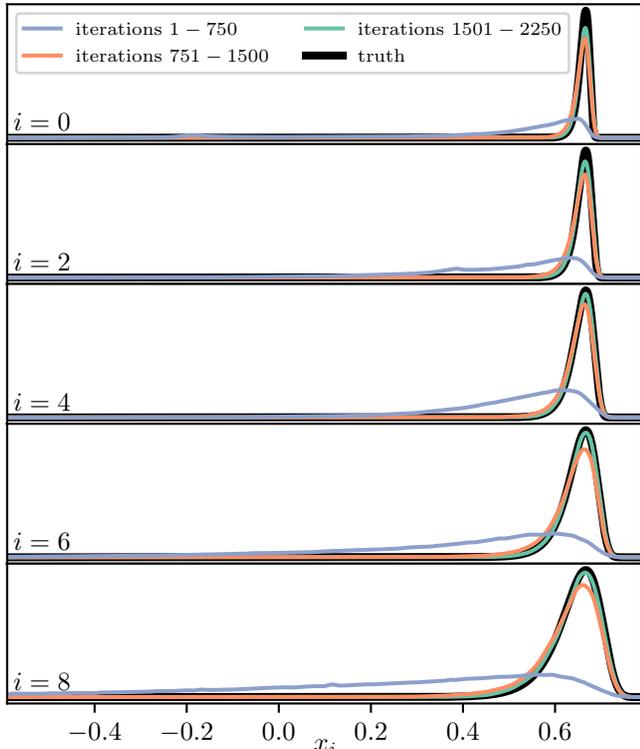


FIG. 9.— Probability distributions for the even parameters in a 10-dimensional log-gamma test over the course of an MCMC analysis. Initial walkers positions were sampled from the posteriors derived by an $N_{\text{live}} = 1000$, $\epsilon = 0.1$ MultiNest analysis. Gradually, as the MCMC algorithm iterates, the walkers move from their initial positions, far from representative of the posterior, to being consistent with the underlying distributions. The distributions are plotted here using averaged shifted histograms (Scott 1985).

starting” built into UltraNest.¹¹

6.1. MCMC

For this test, I began with arguably the most difficult problem — the 10D log-gamma distribution — and started with a MultiNest analysis that missed the mark, one of the $N_{\text{live}} = 1000$, $\epsilon = 0.1$ runs. I first performed a Gaussian kernel density estimate of the MultiNest posterior samples; at this stage I prefer to use a bandwidth an order of magnitude or two (in this case a factor of 10) smaller than the Silverman suggestion (Silverman 1986) to help prevent the kernel density estimate from extending beyond the prior bounds, if applicable. After estimating the density of the posterior, I resampled it, drawing 1024 samples, which I used to initialize the walkers in an MCMC analysis. For this I employed emcee version 3.1.4, using the default settings (Foreman-Mackey et al. 2013).

Some of the results from this test (for every other parameter) are shown in Figure 9. In this Figure, I’ve plotted the probability distributions derived from the first, second, and third groups of 750 MCMC iterations, after initializing the walkers based on draws from the MultiNest-derived posteriors. Over time, the walkers

¹¹ <https://johannesbuchner.github.io/UltraNest/example-warmstart.html>. One must take care when starting from MultiNest outputs, which have a slightly different format than UltraNest outputs.

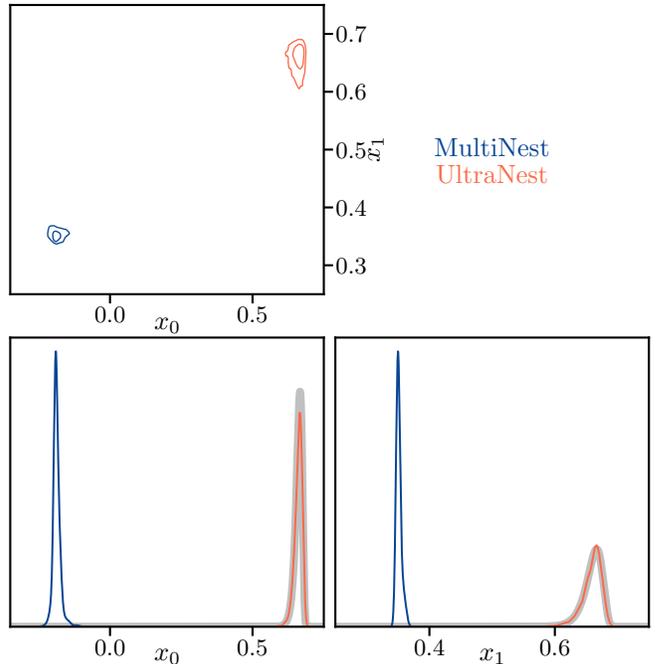


FIG. 10.— Results from a 10-dimensional log-gamma test, illustrated using a corner plot for the first two parameters. MultiNest results are shown in blue, while UltraNest results (warm-started from the MultiNest results) are shown in orange. The true 1D posteriors are shown in gray. The upper panel displays the two-dimensional 1σ and 2σ minimum-area credible regions. The distributions are plotted here using averaged shifted histograms (Scott 1985).

move from their initial positions to instead match the posterior. This sort of behavior can be expected as long as MultiNest is able to identify the most prominent mode in a multimodal parameter space. Because selecting good initial walker positions can be one of the challenges in an MCMC analysis, these two methods can complement each other nicely.

6.2. Starting Warmly

For this test, I began with the same MultiNest outputs as Section 6.1, but I instead used it to warm-start an UltraNest analysis. This technique has already been demonstrated for cases where the information used to warm-start a given nested sampling run is actually representative of the underlying posterior (e.g. Petrosyan & Handley 2022). In this sense, the present test, where the nested sampling run used to warm-start the analysis was quite far from representing the posterior, is something like a worst-case scenario, and demonstrates the robustness of this strategy when the deviations between the warm start and actual posterior are large.

Despite using far from optimal samples to inform the warm-starting procedure, UltraNest was still able to perform adequately without any tuning. For example, UltraNest derived an evidence estimate with an error of $\Delta \log Z = 0.690 \pm 0.605$. On the other hand, MultiNest had an evidence error of $\Delta \log Z = -297.245 \pm 0.259$. Additionally, as illustrated in Figure 10, UltraNest was able to derive much more accurate posterior estimates.

7. CONCLUSIONS

The results presented here imply a few general strategies for using MultiNest in a way that can be reasonably trusted. If one finds consistent evidence estimates and posterior inferences between two or more MultiNest analyses that have varied the number of live points and sampling efficiencies by factors of \sim three or more (larger factors are preferred because of the often-large variance between MultiNest using a given combination of sampling parameters), the results of the analyses can probably be trusted. In practice, lowering the sampling efficiency may be a less resource-intensive strategy, particularly if one finds a fraction of accepted proposals higher than the sampling efficiency. Alternatively, it is straightforward to use an initial MultiNest run to initialize and accelerate an MCMC analysis, or another nested sampling analysis with more rigorous convergence criteria, as described in Section 6.

To reiterate some of the results of the above tests, when used with the “recommended” settings, MultiNest:

- can produce systematically biased estimates of the Bayesian evidence (more significantly for higher-dimension problems),
- can derive completely erroneous posterior estimates,
- is more likely to *underestimate* the width of a credible interval than to overestimate it.

The degree to which MultiNest produces biased results (or whether it instead produces good results) depends sensitively on the dimensionality and structure of a given problem.

Additionally, when sampling noisy likelihoods, MultiNest tends to produce posterior estimates that are sys-

tematically narrow; this bias can be ameliorated by decreasing the sampling efficiency, but is largely independent of the number of live points used. Although the particular test problem examined here is not necessarily analogous to practical applications, this behavior mimics that observed in some actual analyses where MultiNest seemed to significantly underestimate the widths of credible regions (e.g., Miller et al. 2019; Ih & Kempton 2021; Miller et al. 2021). Because MultiNest’s posterior estimates can be *systematically* biased at larger sampling efficiency, it is crucial that any analysis relying on MultiNest-derived posteriors test convergence with respect to the sampling efficiency.

SOFTWARE

MultiNest (Feroz et al. 2009), PyMultiNest (Buchner 2016b), emcee (Foreman-Mackey et al. 2013), UltraNest (Buchner 2021), matplotlib (Hunter 2007), numpy (van der Walt et al. 2011), scipy (Virtanen et al. 2020)

ACKNOWLEDGMENTS

This work benefited from stimulating discussions with and useful feedback from Arjun Savel, Jegug Ih, Matt Nixon, and Cole Miller. I am grateful for the detailed feedback and comments provided by Johannes Buchner on a draft of this manuscript. This work was supported in part by NASA ADAP grants 80NSSC21K0649 and 80NSSC20K0288. Many of these calculations were performed on the ASTRA computing cluster, which is maintained by the Department of Astronomy at the University of Maryland. I also used NASA’s Astrophysics Data System Bibliographic Services and the ArXiv extensively.

REFERENCES

- Albert, J. G. 2020, arXiv e-prints, arXiv:2012.15286, doi: [10.48550/arXiv.2012.15286](https://doi.org/10.48550/arXiv.2012.15286)
- Ashton, G., Bernstein, N., Buchner, J., et al. 2022, Nature Reviews Methods Primers, 2, 39, doi: [10.1038/s43586-022-00121-x](https://doi.org/10.1038/s43586-022-00121-x)
- Beaujean, F., & Caldwell, A. 2013, arXiv e-prints, arXiv:1304.7808, doi: [10.48550/arXiv.1304.7808](https://doi.org/10.48550/arXiv.1304.7808)
- Bradbury, J., Frostig, R., Hawkins, P., et al. 2018, JAX: composable transformations of Python+NumPy programs, 0.3.13. <http://github.com/google/jax>
- Buchner, J. 2016a, Statistics and Computing, 26, 383, doi: [10.1007/s11222-014-9512-y](https://doi.org/10.1007/s11222-014-9512-y)
- . 2016b, PyMultiNest: Python interface for MultiNest, Astrophysics Source Code Library, record ascl:1606.005. <http://ascl.net/1606.005>
- . 2021, The Journal of Open Source Software, 6, 3001, doi: [10.21105/joss.03001](https://doi.org/10.21105/joss.03001)
- . 2023, Statistics Surveys, 17, 169, doi: [10.1214/23-SS144](https://doi.org/10.1214/23-SS144)
- Buchner, J., Georgakakis, A., Nandra, K., et al. 2014, A&A, 564, A125, doi: [10.1051/0004-6361/201322971](https://doi.org/10.1051/0004-6361/201322971)
- Cappé, O., Douc, R., Guillin, A., Marin, J.-M., & Robert, C. P. 2008, Statistics and Computing, 18, 447, doi: [10.1007/s11222-008-9059-x](https://doi.org/10.1007/s11222-008-9059-x)
- Cappé, O., Guillin, A., Marin, J., & Robert, C. P. 2004, Journal of Computational and Graphical Statistics, 13, 907, doi: [10.1198/106186004X12803](https://doi.org/10.1198/106186004X12803)
- Feroz, F., & Hobson, M. P. 2008, MNRAS, 384, 449, doi: [10.1111/j.1365-2966.2007.12353.x](https://doi.org/10.1111/j.1365-2966.2007.12353.x)
- Feroz, F., Hobson, M. P., & Bridges, M. 2009, MNRAS, 398, 1601, doi: [10.1111/j.1365-2966.2009.14548.x](https://doi.org/10.1111/j.1365-2966.2009.14548.x)
- Feroz, F., Hobson, M. P., Cameron, E., & Pettitt, A. N. 2019, The Open Journal of Astrophysics, 2, 10, doi: [10.21105/astro.1306.2144](https://doi.org/10.21105/astro.1306.2144)
- Foreman-Mackey, D., Hogg, D. W., Lang, D., & Goodman, J. 2013, PASP, 125, 306, doi: [10.1086/670067](https://doi.org/10.1086/670067)
- Gelman, A., & Meng, X.-L. 1998, Statistical Science, 13, 163. <http://www.jstor.org/stable/2676756>
- Handley, W. J., Hobson, M. P., & Lasenby, A. N. 2015, MNRAS, 450, L61, doi: [10.1093/mnras/1/slv047](https://doi.org/10.1093/mnras/1/slv047)
- Higson, E., Handley, W., Hobson, M., & Lasenby, A. 2019a, MNRAS, 483, 2044, doi: [10.1093/mnras/sty3090](https://doi.org/10.1093/mnras/sty3090)
- . 2019b, Statistics and Computing, 29, 891, doi: [10.1007/s11222-018-9844-0](https://doi.org/10.1007/s11222-018-9844-0)
- Hunter, J. D. 2007, Computing in Science Engineering, 9, 90, doi: [10.1109/MCSE.2007.55](https://doi.org/10.1109/MCSE.2007.55)
- Ih, J., & Kempton, E. M. R. 2021, AJ, 162, 237, doi: [10.3847/1538-3881/ac173b](https://doi.org/10.3847/1538-3881/ac173b)
- Kipping, D. M. 2013, MNRAS, 434, L51, doi: [10.1093/mnras/1/slt075](https://doi.org/10.1093/mnras/1/slt075)
- Lemos, P., Weaverdyck, N., Rollins, R. P., et al. 2023, MNRAS, 521, 1184, doi: [10.1093/mnras/stac2786](https://doi.org/10.1093/mnras/stac2786)
- Miller, M. C., Lamb, F. K., Dittmann, A. J., et al. 2019, ApJ, 887, L24, doi: [10.3847/2041-8213/ab50c5](https://doi.org/10.3847/2041-8213/ab50c5)
- . 2021, ApJ, 918, L28, doi: [10.3847/2041-8213/ac089b](https://doi.org/10.3847/2041-8213/ac089b)
- Morton, T. D., Bryson, S. T., Coughlin, J. L., et al. 2016, ApJ, 822, 86, doi: [10.3847/0004-637X/822/2/86](https://doi.org/10.3847/0004-637X/822/2/86)
- Nelson, B. E., Ford, E. B., Buchner, J., et al. 2020, AJ, 159, 73, doi: [10.3847/1538-3881/ab5190](https://doi.org/10.3847/1538-3881/ab5190)
- Newton, M. A., & Raftery, A. E. 1994, Journal of the Royal Statistical Society: Series B (Methodological), 56, 3, doi: <https://doi.org/10.1111/j.2517-6161.1994.tb01956.x>
- Pagani, F., Wiegand, M., & Nadarajah, S. 2019, arXiv e-prints, arXiv:1903.09556, doi: [10.48550/arXiv.1903.09556](https://doi.org/10.48550/arXiv.1903.09556)
- Petrosyan, A., & Handley, W. J. 2022, arXiv e-prints, arXiv:2212.01760, doi: [10.48550/arXiv.2212.01760](https://doi.org/10.48550/arXiv.2212.01760)
- Piessens, R., de Doncker-Kapenga, E., & Ueberhuber, C. W. 1983, Quadpack. A subroutine package for automatic integration Planck Collaboration, Ade, P. A. R., Aghanim, N., et al. 2014, A&A, 571, A22, doi: [10.1051/0004-6361/201321569](https://doi.org/10.1051/0004-6361/201321569)
- . 2016, A&A, 594, A20, doi: [10.1051/0004-6361/201525898](https://doi.org/10.1051/0004-6361/201525898)

- Planck Collaboration, Akrami, Y., Arroja, F., et al. 2020, *A&A*, 641, A10, doi: [10.1051/0004-6361/201833887](https://doi.org/10.1051/0004-6361/201833887)
- Riley, T. E., Watts, A. L., Bogdanov, S., et al. 2019, *ApJ*, 887, L21, doi: [10.3847/2041-8213/ab481c](https://doi.org/10.3847/2041-8213/ab481c)
- Salomone, R., South, L. F., Johansen, A. M., Drovandi, C., & Kroese, D. P. 2018, arXiv e-prints, arXiv:1805.03924, doi: [10.48550/arXiv.1805.03924](https://doi.org/10.48550/arXiv.1805.03924)
- Scott, D. W. 1985, *The Annals of Statistics*, 13, 1024, doi: [10.1214/aos/1176349654](https://doi.org/10.1214/aos/1176349654)
- Scott, D. W. 2015, *Multivariate Density Estimation: Theory, Practice, and Visualization*
- Silverman, B. W. 1986, *Density estimation for statistics and data analysis*
- Skilling, J. 2004, in *American Institute of Physics Conference Series*, Vol. 735, *Bayesian Inference and Maximum Entropy Methods in Science and Engineering: 24th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, ed. R. Fischer, R. Preuss, & U. V. Toussaint, 395–405, doi: [10.1063/1.1835238](https://doi.org/10.1063/1.1835238)
- Speagle, J. S. 2020, *MNRAS*, 493, 3132, doi: [10.1093/mnras/staa278](https://doi.org/10.1093/mnras/staa278)
- Tsiaras, A., Waldmann, I. P., Zingales, T., et al. 2018, *AJ*, 155, 156, doi: [10.3847/1538-3881/aaaf75](https://doi.org/10.3847/1538-3881/aaaf75)
- van der Walt, S., Colbert, S. C., & Varoquaux, G. 2011, *Computing in Science Engineering*, 13, 22, doi: [10.1109/MCSE.2011.37](https://doi.org/10.1109/MCSE.2011.37)
- Virtanen, P., Gommers, R., Oliphant, T. E., et al. 2020, *Nature Methods*, 17, 261, doi: [10.1038/s41592-019-0686-2](https://doi.org/10.1038/s41592-019-0686-2)
- Vousden, W. D., Farr, W. M., & Mandel, I. 2016, *MNRAS*, 455, 1919, doi: [10.1093/mnras/stv2422](https://doi.org/10.1093/mnras/stv2422)
- Walter, C. 2017, *Statistics and Computing*, 27, 219, doi: [10.1007/s11222-015-9617-y](https://doi.org/10.1007/s11222-015-9617-y)
- Wolfe, N. E., Talbot, C., & Golomb, J. 2023, *Phys. Rev. D*, 107, 104056, doi: [10.1103/PhysRevD.107.104056](https://doi.org/10.1103/PhysRevD.107.104056)

This paper was built using the Open Journal of Astrophysics L^AT_EX template. The OJA is a journal which provides fast and easy peer review for new papers in the **astro-ph**

section of the arXiv, making the reviewing process simpler for authors and referees alike. Learn more at <http://astro.theoj.org>.