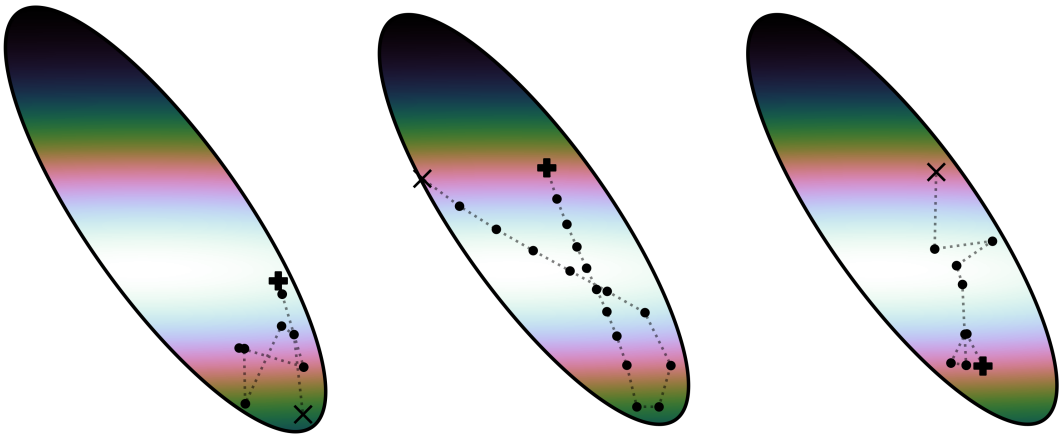


Constrained space MCMC methods for nested sampling Bayesian computations

Master's thesis in Physics and Astronomy

JACOB OLANDER

Department of Physics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2020

Thesis for the degree of Master of Science

Constrained space MCMC methods for nested sampling Bayesian computations

JACOB OLANDER

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Physics
Division of Subatomic, High Energy and Plasma Physics
Theoretical Subatomic Physics
Chalmers University of Technology
Gothenburg, Sweden 2020

Constrained space MCMC methods for nested sampling Bayesian computations
JACOB OLANDER

Supervisor and examiner: Christian Forssén, Department of Physics

Department of Physics
Division of Subatomic, High Energy and Plasma Physics
Theoretical Subatomic Physics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Visualizations of likelihood-constrained MCMC random walks for prior sampling in two dimensions. Three different methods are shown: constrained Metropolis (left), Galilean Monte Carlo (middle) and the constrained stretch move (right). The likelihood constraint is illustrated by the ellipsoid-shaped bound and the prior by the underlying density color map.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2020

Constrained space MCMC methods for nested sampling Bayesian computations
JACOB OLANDER
Department of Physics
Chalmers University of Technology

Abstract

Natural phenomena can in general be described using several different scientific models, which creates a need for systematic model selection. Bayesian model comparison assigns relative probabilities to a set of possible models using the model evidence (marginal likelihood), obtained by an integral that in general needs to be evaluated numerically. Nested sampling is a conceptual framework that efficiently estimates the model evidence and, additionally, provides samples from the model parameter posterior distribution used in Bayesian parameter estimation. A vital step of nested sampling is the likelihood-constrained sampling of the model parameter prior distribution, a task that has proven particularly difficult and that is subject to ongoing research. In this thesis we implement, evaluate and compare three methods for constrained sampling in conjunction with a nested sampling framework. The methods are variants of Markov chain Monte Carlo algorithms: Metropolis, Galilean Monte Carlo and the affine-invariant stretch move, respectively. The latter is applied in the context of nested sampling for the first time in this work. The performances of the methods are assessed by their application to a reference problem that has a known analytical solution. The problem is inspired by effective field theories in subatomic physics where the model parameters take the form of coefficients that are of natural size. We conclude that the efficiency and computational accuracy of nested sampling is strongly dependent on the choice of sampling method and the settings of its associated hyperparameters. In certain cases, especially for high-dimensional parameter spaces, the implementations of this work are seen to achieve better computational accuracy than `MultiNest`, a state-of-the-art nested sampling implementation extensively used in astronomy and cosmology. Generally for nested sampling, we observe that it is possible to obtain an inaccurate result without receiving any clear warning signs indicating that this is the case. However, we demonstrate that the validity of the computational results can be assessed by monitoring the sampling process.

Keywords: Bayesian inference, parameter estimation, model comparison, evidence, nested sampling, MCMC, Metropolis, Galilean Monte Carlo, affine-invariant sampling

Acknowledgements

I wish to express my deepest gratitude to my supervisor Christian Forssén for invaluable guidance and for sharing his endless knowledge and wisdom. Furthermore, I would like to thank all members of the Theoretical Subatomic Physics group for interesting discussion and fruitful input during the course of my project. Lastly, I want to recognize the tireless support from my parents, without which this thesis could never have been completed.

Jacob Olander, Gothenburg, June 2020

Contents

List of Figures **xi**

List of Tables **xv**

1 Introduction **1**
1.1 Purpose and scope . . . 2
1.2 Outline . . . 2

2 Bayesian data analysis and numerical methods **3**
2.1 Bayesian inference . . . 3
2.1.1 EFT toy example . . . 4
2.1.2 Parameter estimation . . . 5
2.1.3 Model comparison . . . 5
2.1.4 EFT toy example revisited . . . 6
2.2 Markov chain Monte Carlo methods . . . 8
2.2.1 Metropolis-Hastings . . . 9
2.2.2 Hamiltonian Monte Carlo . . . 10
2.2.3 Sampling with affine invariance: the stretch move . . . 11

3 Nested sampling **13**
3.1 Background . . . 13
3.1.1 Prior mass fraction and sorted likelihood function . . . 13
3.2 The algorithm . . . 16
3.2.1 Statistical properties . . . 16
3.2.2 Obtaining the evidence and posterior . . . 17
3.2.3 Pseudocode . . . 19
3.2.4 Termination . . . 19
3.2.5 Uncertainty estimation . . . 20
3.3 Generating new points . . . 20
3.3.1 Constrained Metropolis . . . 21
3.3.2 Galilean Monte Carlo . . . 21
3.3.3 Constrained stretch move . . . 23
3.3.4 Ellipsoidal nested sampling . . . 23

4 Constrained prior sampling implementations **25**
4.1 Choice of coordinates: the unit hypercube . . . 25
4.2 Constrained Metropolis implementation . . . 25
4.3 Galilean Monte Carlo implementation . . . 27
4.4 Constrained stretch move implementation . . . 29

4.5 Monitoring the sampling progress 30
4.6 Hyperparameters 33
4.6.1 Sensitivity to the scale parameter value 33
4.6.2 Choosing a sufficient number of steps 36

5 Convergence **37**
5.1 The curse of dimensionality 37
5.2 Divergence in higher dimensions 38
5.2.1 Debunking the uncertainty estimate 42

6 Conclusion **45**

References **49**

A Posterior probability distributions **I**

List of Figures

2.1 The true function $g(x)$ from which synthetic data is generated in the EFT toy example. Data consists of 10 points evenly distributed in $0 < x \leq 1/\pi$ with a relative error $c = 5\%$ according to Equation (2.7). 7

2.2 Marginal distributions of the analytically derived posterior in the EFT example for a model with $n = 3$ parameters. Two-dimensional distributions for pairs of parameters are located below the diagonal on which the one-dimensional distributions are displayed. Point estimates are given for each parameter in terms of their mean values. Error estimates are given in terms of standard deviations. 8

2.3 The model evidence computed for models $\mathcal{M}_n$ defined by their parameter space dimensionality n. It is clear that the evidence is maximized for $n = 3$ which means that $\mathcal{M}_3$ is the model favoured by the data. The evidence for $n = 1$ is ~ 0 and is therefore not included in the figure. 9

3.1 Illustrations of the relationship between the likelihoods $\mathcal{L}(\boldsymbol{\theta})$ and $L(\xi)$, the prior mass fraction ξ and the model evidence Z. 15

3.2 Evolution of active points in a nested sampling iteration. In (a) the worst point is identified and removed. The limit ξ^* is updated in (b) to the position the point was removed from. A new point is generated in (c), nested within the current limit. 16

3.3 Sequence of discarded points obtained by estimating the prior mass fractions ξ_k statistically. The data is artificially produced. 18

3.4 Illustrations of the basic principles of Galilean Monte Carlo. 23

4.1 Example of a Metropolis random walk in a likelihood-constrained region of a two-dimensional parameter space. The walk is captured at a nested sampling iteration with $\xi \approx e^{-\mathcal{H}}$. The green plus and the red cross indicates the start of the walk and the end of the walk, respectively. The black dots indicate intermediate steps. The average number of steps for this walk is $\langle N_\mathrm{s} \rangle = 20$ and the scale parameter is $s = 0.5$. 27

4.2 Example of a GMC trajectory in a likelihood-constrained region of a two-dimensional parameter space. The trajectory is captured at a nested sampling iteration with $\xi \approx e^{-\mathcal{H}}$. The green plus and the red cross indicates the start of the trajectory and the end of the walk, respectively. The black dots indicate intermediate time steps. The reflections do not occur exactly at the boundary but rather at proxy surfaces just outside the boundary (not in figure) as described in Section 3.3.2. 29

4.3 Example of a stretch move random walk in a likelihood-constrained region of a two-dimensional parameter space. The walk is captured at a nested sampling iteration with $\xi \approx e^{-\mathcal{H}}$. The green plus and the red cross indicates the start of the walk and the end of the walk, respectively. The black dots indicate intermediate steps. 30

4.4 Progress of three key quantities over the course of a nested sampling run for the three different methods. The sampling is performed in $n = 3$ dimensions using $\langle N_\mathrm{s} \rangle = 40$ with sampling parameters $N = 1000$ and $f_\mathrm{ln} = 0.01$. The top panel shows the likelihood L for the worst point of each iteration, the middle panel shows the posterior weights w, used to compute the evidence, and the bottom panel shows the moving median of the acceptance rate r_a for each iteration. The methods distinguish themselves clearly in the variation of the acceptance rate. 32

4.5 The (log) evidence error (top row), the bulk median acceptance rate (middle row) and the number of likelihood evaluations (bottom row) in $n = 3$ dimensions for the three methods applied in the nested sampling framework to the toy problem in Section 2.1.1 for different values of the scale parameters s, τ and a. Sampling parameters were $N = 1000$ active points, $f_\mathrm{ln} = 0.01$ tolerance and $\langle N_\mathrm{s} \rangle = 40$ exploration steps per iteration. 35

4.6 Error in the computed (log) evidence with different choices of the average number of steps in each nested sampling iteration. Sampling parameters are $N = 1000$ active points and $f_\mathrm{ln} = 0.01$ and the scale parameters are as indicated by the legend. The bands are the standard deviations of five identical runs with different random seeds. The general trend is that the error approaches zero and fluctuates less for larger $\langle N_\mathrm{s} \rangle$. 36

5.1 Illustration of the exponential increase of adjacent points with the dimensionality n of a Euclidean space. In (a): one, (b): two and (c): three dimensions, a grid point (filled circle) has 2, 8 and 26 neighbouring points (unfilled circles), respectively. In arbitrary dimensions, n, the corresponding number is $3^n - 1$. 38

5.2 The evidence error (top row), the bulk median acceptance rate (middle row) and the number of likelihood evaluations (bottom row) for varying parameter space dimensionality. Each point is the average of five identical runs and the bands are corresponding standard deviations. The left and right columns display two different choices of hyperparameters, respectively, as specified in the figure. Sampling parameters are $N = 1000$ and $f_\mathrm{ln} = 0.01$. The overall trend is that performance worsens and computations become less accurate for higher dimensions, as expected. 41

5.3 Tracked progress for the three methods, displayed in terms of the likelihood L, the posterior weights w and the acceptance rate r_a in $n = 24$ dimensions using $\langle N_\mathrm{s} \rangle = 40$. In contrast to the $n = 3$ case in Figure 4.4, the methods are seen to disagree on where the bulk of the posterior is located in terms of ξ, resulting in significant differences in the computed evidences. The vertical lines indicate the computed information $\mathcal{H}$. Sampling parameters are $N = 1000$ and $f_\mathrm{ln} = 0.01$. 42

A.1 Marginal posterior distributions obtained using the constrained Metropolis method with $s = 0.5$ and $\langle N_s \rangle = 40$. The distributions are compared to the analytical solution (dash-dotted). Vertical dashed lines represent the sample median and 16^{th} and 84^{th} percentiles, respectively. The Taylor coefficients (Equation (2.17)) are indicated by the square markers and vertical solid lines. The histograms contain ~ 5000 samples in each case. $N = 1000$ and $f_{\text{ln}} = 0.01$. II

A.2 Marginal posterior distributions obtained using the GMC method with $\tau = 0.1$ and $\langle N_s \rangle = 40$. The distributions are compared to the analytical solution (dash-dotted). Vertical dashed lines represent the sample median and 16^{th} and 84^{th} percentiles, respectively. The Taylor coefficients (Equation (2.17)) are indicated by the square markers and vertical solid lines. The histograms contain ~ 5000 samples in each case. $N = 1000$ and $f_{\text{ln}} = 0.01$. III

A.3 Marginal posterior distributions obtained using the constrained stretch move method with $a = 2.0$ and $\langle N_s \rangle = 40$. The distributions are compared to the analytical solution (dash-dotted). Vertical dashed lines represent the sample median and 16^{th} and 84^{th} percentiles, respectively. The Taylor coefficients (Equation (2.17)) are indicated by the square markers and vertical solid lines. The histograms contain ~ 5000 samples in each case. $N = 1000$ and $f_{\text{ln}} = 0.01$. III

A.4 Marginal posterior distributions obtained using `PyMultiNest`. The distributions are compared to the analytical solution (dash-dotted). Vertical dashed lines represent the sample median and 16^{th} and 84^{th} percentiles, respectively. The Taylor coefficients (Equation (2.17)) are indicated by the square markers and vertical solid lines. The histograms contain ~ 5000 samples in each case. $N = 1000$ and $f_{\text{ln}} = 0.01$. IV

List of Tables

4.1 Nested sampling results for the three methods applied to the same problem in $n = 3$ dimensions. Sampling parameters are $N = 1000$ active points and $f_{\ln} = 0.01$ tolerance. The true evidence value for this model is $\ln Z_{\text{true}} = 8.09$ as was analytically obtained in Figure 2.3. The number of likelihood evaluations, $N_{\mathcal{L}}$, does in the GMC case, include the number of evaluations of the gradient. The fraction of gradient evaluations is given in parenthesis. 32

5.1 The computed (log) evidence $\ln Z$ along with the uncertainty estimate $\pm\sqrt{\mathcal{H}/N}$ and the number of likelihood evaluations $N_{\mathcal{L}}$ required as functions of the parameter space dimensionality n. The analytical values $\ln Z_{\text{true}}$ are given for reference. Hyperparameter settings are the same as in the left column of Figure 5.2. Every value is the average obtained from five identical runs. 43

1
Introduction

Scientific models are typically associated with a set of model parameters used to make explicit theoretical predictions. A theoretical framework may in fact contain one or a set of competing models $\{\mathcal{M}_n\}$, each with their own collection of model parameters whose values need to be calibrated against experimental observation. A suitable example in subatomic physics regards effective field theories (EFT), e.g. chiral effective field theory (χEFT) [1], which are described by observable coefficients that parametrize the theory at the scale at which it is valid [2]. In this type of scenario we would like to make well-informed statements about which model and associated model parameter values to prefer, given a set of experimental data. To this end we employ Bayesian statistical methods [3, 4] in order to perform inductive inference, including *parameter estimation* and *model comparison*. For parameter estimation, the central object is the *posterior* probability distribution for the parameters, which in general needs to be represented by a set of random samples. The key quantity for model comparison is the Bayesian model *evidence* (or marginal likelihood), obtained by integration over the space of model parameters. By evaluating the evidence, competing models in the set can be assigned a relative probability, indicating which model is favoured by the data.

Bayesian inference generally requires demanding numerical computations in terms of high-dimensional evidence integrals and sampling of complex posterior probability distributions. For this purpose one mainly resorts to Markov chain Monte Carlo (MCMC) methods [5] which explore the model parameter space in order to find regions of high probability. However, although MCMC methods, such as the Metropolis-Hastings algorithm [6, 7], theoretically could be used to estimate the evidence integral, in practice they fail to do so with acceptable efficiency.

A less established Monte Carlo method compared to MCMC is *nested sampling*, introduced by Skilling [8, 9] and further described by Sivia and Skilling [4]. Nested sampling was specifically developed to efficiently provide an estimate of the model evidence. As a by-product it also generates a set of samples from the posterior. The algorithm has been successfully applied in parameter estimation and model comparison applications in a broad variety of fields, from astronomy, cosmology and particle physics [10–15] to biomathematics [16–18]. The basic idea of nested sampling is to transform the high-dimensional evidence integral to a one-dimensional integral over unit range by considering secluded and closed regions of parameter space, *nested* within each other. A most crucial step in the nested sampling procedure is to generate independent samples from within these constrained regions and a variety of methods for performing this step have been proposed and implemented [10, 19–23]. Producing high-quality constrained samples has proven to be a difficult task, to say the least, and is subject to ongoing research. This work aims at contributing to this research by implementing and evaluating three different constrained

space sampling methods which are integrated into a nested sampling framework. The implementations are in themselves MCMC based although their surrounding nested sampling environment is not. Two of the methods are inspired by previous work on the topic whereas the third method is introduced to the context of nested sampling for the first time in this work. The implemented methods are applied to an EFT-inspired example problem and evaluated based on their respective performance.

1.1 Purpose and scope

The purpose of this thesis is to increase the understanding of the nested sampling algorithm in general and constrained space sampling in particular. This is achieved by proposing the specific designs, implementing and evaluating three different methods for constrained sampling in conjunction with nested sampling. Focus will mainly be on the performance based on the specifics of the three methods and not on the design of the nested sampling algorithm in general. In contrast to sampling of unconstrained probability distributions, the process of constrained sampling is poorly understood. A specific goal of this work is to present modified versions of ordinary MCMC methods, adjusted to suit the requirements introduced by nested sampling.

1.2 Outline

In order to understand the context in which nested sampling is used it is necessary to grasp the concepts of the Bayesian statistical analysis framework. This background is provided in Chapter 2 and includes the basics of Bayesian inference as well as a practical example on parameter estimation and model comparison. Furthermore, the chapter is ended by a brief description of a few MCMC sampling methods which are conventionally used in the context of Bayesian inference. The underlying ideas and a full description of the nested sampling algorithm are given in Chapter 3. The chapter additionally introduces the principles of the constrained sampling methods implemented in this work and describes their place in relation to nested sampling. The specifics of the design choices for the implementations are described in Chapter 4 which further presents diagnostic results used to optimize the methods. The first method being presented is a constrained version of the Metropolis algorithm, the second is a version of Galielean Monte Carlo and the third is a constrained version of an affine-invariant algorithm referred to as the stretch move. The latter method is to our knowledge unique to this work. In Chapter 5 the methods are observed when pushed over their tipping points and broken down by applying them to models with increasingly larger parameter space dimensionalities. Here we also discuss uncertainty estimates and make a comparison to the well-used state-of-the-art nested sampling software `MultiNest` [10, 11, 24], which the methods of this work are seen to outperform in some cases. The thesis is concluded in Chapter 6 by summarizing the main findings and by proposing starting points for further related work.

2

Bayesian data analysis and numerical methods

Bayesian statistics and its applications is a vast subject and is probably relevant to most scientific research or any field of work where making predictions from data and previous experience is important. Here follows a brief description of the aspects of Bayesian inference relevant to this work. For an extensive review of the subject, see e.g. MacKay [3] or Sivia and Skilling [4].

2.1 Bayesian inference

One of the powers of the Bayesian description of probability is that it allows for a relation to be established between a conditional probability $\text{prob}(X|Y)$ and its reverse $\text{prob}(Y|X)$ where X and Y are propositions that could be outcomes of random processes (although that property is not required). This reversal is desirable in scenarios e.g. where $\text{prob}(X|Y)$ is the sought after quantity but is difficult to directly write down whereas $\text{prob}(Y|X)$ is easier to interpret and more naturally expressed. The product rule for joint probabilities states that

$$\text{prob}(X, Y) = \text{prob}(X|Y)\text{prob}(Y), \tag{2.1}$$

which in words reads: the probability that X *and* Y occurs equals the probability that X occurs *given* that Y has occurred multiplied by the probability that Y occurs. Joint probabilities are symmetric such that $\text{prob}(X, Y) = \text{prob}(Y, X)$ which by the product rule in Equation (2.1) gives Bayes' theorem

$$\text{prob}(X|Y) = \frac{\text{prob}(Y|X)\text{prob}(X)}{\text{prob}(Y)}, \tag{2.2}$$

which yields the relation between the conditional probabilities $\text{prob}(X|Y)$ and $\text{prob}(Y|X)$. The marginal probabilities $\text{prob}(X)$ and $\text{prob}(Y)$ can be expressed by marginalization of the joint probability $\text{prob}(X, Y) \overset{(2.1)}{=} \text{prob}(Y|X)\text{prob}(X)$ through the sum rule according to

$$\text{prob}(Y) = \begin{cases} \int \text{prob}(Y|X)\text{prob}(X)\text{d}X, & \text{for continuous } X \\ \sum_X \text{prob}(Y|X)\text{prob}(X), & \text{for discrete } X \end{cases} \tag{2.3}$$

and correspondingly for $\text{prob}(X)$.

In the context of this work, X takes the form of an n-dimensional *parameter vector* $\boldsymbol{\theta} = (\theta_0, ..., \theta_{n-1})$ used to make predictions in a given *model* $\mathcal{M}$. Y is in this context a set of *data* D obtained from an experiment or simulation yielding measurements of some

quantity sought to be predicted by $\mathcal{M}$. In other words, the goal is to find the probability (distribution) of the parameters $\boldsymbol{\theta}$ given data D. Bayes' theorem then reads

$$\text{prob}(\boldsymbol{\theta}|D, I) = \frac{\text{prob}(D|\boldsymbol{\theta}, I)\text{prob}(\boldsymbol{\theta}|I)}{\text{prob}(D|I)} \tag{2.4}$$

where I is any other relevant information including e.g. the choice of $\mathcal{M}$. The probabilities entering Equation (2.4) are

$$\text{the } \textit{posterior} \text{ probability distribution of the parameters} \quad \text{prob}(\boldsymbol{\theta}|D, I) \equiv \mathcal{P}(\boldsymbol{\theta}), \tag{2.5a}$$

$$\text{the } \textit{likelihood} \text{ of the data} \quad \text{prob}(D|\boldsymbol{\theta}, I) \equiv \mathcal{L}(\boldsymbol{\theta}), \tag{2.5b}$$

$$\text{the } \textit{prior} \text{ probability distribution of the parameters} \quad \text{prob}(\boldsymbol{\theta}|I) \equiv \pi(\boldsymbol{\theta}), \tag{2.5c}$$

$$\text{the Bayesian } \textit{evidence} \text{ (or marginal likelihood)} \quad \text{prob}(D|I) \equiv Z. \tag{2.5d}$$

2.1.1 EFT toy example

In order to put Bayesian inference into context, we here review a simple EFT-inspired example from Schindler et al. [25] and Wesolowski et al. [26]. In this example, a function $g(x)$ is taken to represent the true behavior of some observable in the underlying theory and x is an EFT expansion parameter, e.g. a ratio of momenta representing a low-energy physics scale and a high-energy breakdown scale. The function is chosen such that the coefficients of its Taylor expansion at $x = 0$ within $|x| < 1$ ought to be of natural order:

$$g(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + ... \tag{2.6}$$

In this way the observable coefficients a_i are related to order-by-order predictions of the EFT. More specifically, including the first n terms would correspond to the prediction of the EFT truncated at order n. By physical arguments the observable coefficients are expected to be of natural size [2]. To simulate observations from an experiment, synthetic data points d are generated by adding a Gaussian relative noise to $g(x)$ at points of measurement x_j, $j = 1, ..., N_d$:

$$d_j = g(x_j)(1 + c\eta_j) \quad \text{with experimental error} \quad \sigma_j = cd_j \tag{2.7}$$

where $\eta_j \sim \mathcal{N}(0, 1)$[1] is a standard normal random sample and c is a relative error (5% in this example). To make accurate predictions, the observable coefficients need to be calibrated against this data which makes them ideal subjects for Bayesian parameter estimation (see Section 2.1.2). In this example, our model $\mathcal{M}$ is a polynomial of order $n - 1$,

$$g_{\mathcal{M}}(x; \boldsymbol{\theta}) = \sum_{i=0}^{n-1} \theta_i x^i, \tag{2.8}$$

modelling the behavior of the true function $g(x)$. This is where the parameter vector $\boldsymbol{\theta}$ of the model enters, upon which the predictions of $\mathcal{M}$ depends. A non-Bayesian approach usually aims to find a point estimate of the parameters, such as the least-squares minimization $\arg\min_{\boldsymbol{\theta}} \chi^2(\boldsymbol{\theta})$ where

$$\chi^2(\boldsymbol{\theta}) = \sum_{j=1}^{N_d} \left(\frac{d_j - g_{\mathcal{M}}(x_j; \boldsymbol{\theta})}{\sigma_j} \right)^2. \tag{2.9}$$

[1] $\mathcal{N}(\mu, \sigma^2)$ denotes a normal distribution with mean μ and variance σ^2.

This approach might be sufficient in some applications but often fails to provide any information on the distribution of the parameters and thereby their uncertainties. Neither does it take into account any prior knowledge of the parameters (such as naturalness in the case of the EFT expansion). We will now turn to the general problem of parameter estimation and model comparison. However, we will revisit this toy example in Section 2.1.4.

2.1.2 Parameter estimation

The model parameters $\boldsymbol{\theta}$ are in the Bayesian formalism estimated using the posterior $\mathcal{P}(\boldsymbol{\theta})$ determined by Bayes' theorem Equation (2.4). In parameter estimation applications it is possible to ignore the evidence Z as it is independent of $\boldsymbol{\theta}$ and merely amounts to a normalization factor; it is however a central quantity in the context of Bayesian model comparison (see Section 2.1.3). The posterior

$$\mathcal{P}(\boldsymbol{\theta}) \propto \mathcal{L}(\boldsymbol{\theta})\pi(\boldsymbol{\theta}) \tag{2.10}$$

is consequently determined solely by the likelihood and prior and is obtained either by numerical methods, such as MCMC sampling (see Section 2.2), or as an analytical solution (if possible). Subsequently, it is possible to derive distributions for any subset $\theta_j, ..., \theta_k$ of parameters by marginalization

$$\text{prob}(\theta_j, ..., \theta_k | D, I) = \int \mathcal{P}(\boldsymbol{\theta}) \prod_{i \neq j,...,k} \mathrm{d}\theta_i, \tag{2.11}$$

where integration is taken over the appropriate ranges in $\boldsymbol{\theta}$. Marginalization is useful for e.g. discarding of nuisance parameters but also convenient for visualization which (unfortunately) is bounded by two or three dimensions. Examples of such two- and one-dimensional marginal probability density functions (pdfs) are shown in Section 2.1.4. Point estimates of any function $f(\boldsymbol{\theta})$ with respect to the posterior are obtained with the expectation value

$$\mathbb{E}[f] = \int f(\boldsymbol{\theta})\mathcal{P}(\boldsymbol{\theta})\mathrm{d}^n\boldsymbol{\theta} \tag{2.12}$$

where the special cases $f(\boldsymbol{\theta}) = \theta_i$ and $f(\boldsymbol{\theta}) = (\theta_i - \mathbb{E}[\theta_i])(\theta_j - \mathbb{E}[\theta_j])$ give the parameter means and covariances respectively. It is important to stress, however, that the posterior contains more information than mere point estimates such as mean or mode values. The posterior has in general a complex structure, such as multiple modes, not well described by a single number. Only in the special case when the posterior is (approximately) Gaussian can it be fully described by its mean vector and covariance matrix.

2.1.3 Model comparison

Model comparison can be illustrated by the problem of determining which of two possible models $\mathcal{M}_1$ and $\mathcal{M}_2$ (e.g. polynomials of different orders in the example of Section 2.1.1) to prefer. In the Bayesian formalism this problem is addressed by comparing the model posteriors $\text{prob}(\mathcal{M}_1|D, I)$ and $\text{prob}(\mathcal{M}_2|D, I)$[2]. These posteriors are conditioned on the same data D and do not include any parameter dependence as they are more general than to describe a certain fit for a given prediction. As opposed to parameter estimation the evidence Z plays a crucial role in model comparison. It is given by normalization of the posterior

[2]Note that as $\mathcal{M}_1$ and $\mathcal{M}_2$ are here made explicit they are not included in I.

$$Z = \int \text{prob}(D|\boldsymbol{\theta}, \mathcal{M}, I)\text{prob}(\boldsymbol{\theta}|\mathcal{M}, I)\text{d}^n\boldsymbol{\theta} = \int \mathcal{L}(\boldsymbol{\theta})\pi(\boldsymbol{\theta})\text{d}^n\boldsymbol{\theta}. \tag{2.13}$$

The model comparison is then performed by constructing the ratio of the model posterior pdfs using Bayes' theorem according to

$$\frac{\text{prob}(\mathcal{M}_1|D, I)}{\text{prob}(\mathcal{M}_2|D, I)} = \frac{\text{prob}(D|\mathcal{M}_1, I)\text{prob}(\mathcal{M}_1|I)}{\text{prob}(D|\mathcal{M}_2, I)\text{prob}(\mathcal{M}_2|I)} \tag{2.14}$$

where the common factor $\text{prob}(D|I)$ has cancelled. If there is no initial reason to prefer one model over the other, the ratio of model priors will evaluate to $\text{prob}(\mathcal{M}_1|I)/\text{prob}(\mathcal{M}_2|I) = 1$. The remaining ratio is called the Bayes' factor and should be determined in order to compare the models as

$$\frac{\text{prob}(\mathcal{M}_1|D, I)}{\text{prob}(\mathcal{M}_2|D, I)} \longrightarrow \frac{\text{prob}(D|\mathcal{M}_1, I)}{\text{prob}(D|\mathcal{M}_2, I)}. \tag{2.15}$$

Using the sum (2.3) and product (2.1) rules the Bayes' factor can be computed by integration as

$$\frac{\text{prob}(D|\mathcal{M}_1, I)}{\text{prob}(D|\mathcal{M}_2, I)} = \frac{\int \text{prob}(D|\boldsymbol{\theta}, \mathcal{M}_1, I)\text{prob}(\boldsymbol{\theta}|\mathcal{M}_1, I)\text{d}^n\boldsymbol{\theta}}{\int \text{prob}(D|\boldsymbol{\theta}, \mathcal{M}_2, I)\text{prob}(\boldsymbol{\theta}|\mathcal{M}_2, I)\text{d}^n\boldsymbol{\theta}} \overset{(2.13)}{=} \frac{Z_1}{Z_2} \tag{2.16}$$

meaning that the evidences, $Z_{1,2}$, are used to quantitatively assess the relative performance of different models.

2.1.4 EFT toy example revisited

In this section the representative problem introduced in Section 2.1.1 of estimating observable coefficients in an EFT will be approached using Bayesian inference. A function $g(x)$ with the desired property of natural Taylor coefficients a_i, described in the example, is

$$\begin{aligned} g(x) = \left(\frac{1}{2} + \tan\left(\frac{\pi}{2}x\right)\right)^2 &= a_0 + a_1 x + a_2 x^2 + a_3 x^3 + ... \\ &= \frac{1}{4} + \frac{\pi}{2}x + \frac{\pi^2}{4}x^2 + \frac{\pi^3}{24}x^3 + ... \\ &\approx 0.25 + 1.57x + 2.47x^2 + 1.29x^3 + ... \end{aligned} \tag{2.17}$$

which here will be used to generate synthetic data. The true function (2.17) along with data and associated errors generated according to Equation (2.7) can be seen in Figure 2.1. The data consists of 10 points evenly distributed in $0 < x \leq 1/\pi$ and the relative error is $c = 5\%$. As a reminder, the goal is to predict the coefficients a_i (and thereby also $g(x)$) given this data by studying the parameters $\boldsymbol{\theta}$ of the model function $g_{\mathcal{M}}(x; \boldsymbol{\theta})$ defined in Equation (2.8). This is done by constructing a posterior pdf for the parameters via a likelihood and a prior according to Bayes' theorem (2.4)).

Using the principle of maximum entropy (MaxEnt) [27] it is possible to derive [4] a likelihood for the data. Assuming uncorrelated data, the resulting pdf is

$$\text{prob}(D|\boldsymbol{\theta}, I) = \prod_{j=1}^{N_d} \left(\frac{1}{\sqrt{2\pi\sigma_j^2}}\right) \exp\left(-\frac{\chi^2}{2}\right) \tag{2.18}$$

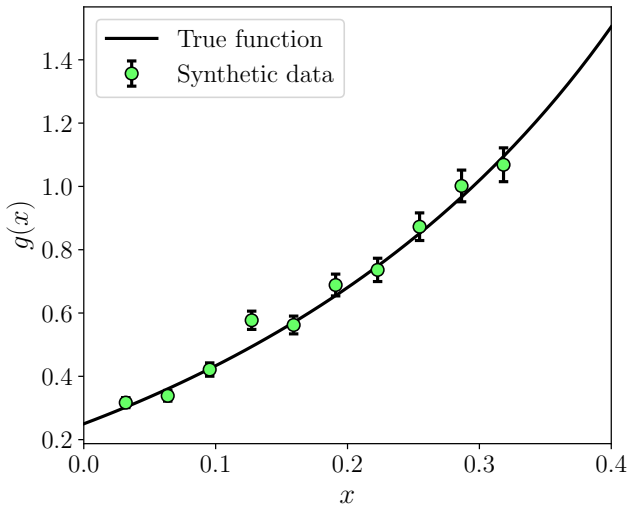


Figure 2.1: The true function $g(x)$ from which synthetic data is generated in the EFT toy example. Data consists of 10 points evenly distributed in $0 < x \leq 1/\pi$ with a relative error $c = 5\%$ according to Equation (2.7).

which is an N_d-dimensional Gaussian in the data and where χ^2 is defined in Equation (2.9). It is important to note that this is a pdf for the data D, *conditioned* on the parameters $\boldsymbol{\theta}$, not a pdf for the parameters themselves. If a uniform prior, i.e. $\text{prob}(\boldsymbol{\theta}|I) = \text{const.}$, is chosen in combination with the likelihood (2.18), the maximum of the posterior would be exactly at the least-squares point estimate discussed in Section 2.1.1 (given that the prior range includes this point). To account for the naturalness of the coefficients, however, Wesolowski et al. [26] introduces a *naturalness prior* given by the symmetric Gaussian $\mathcal{N}(\boldsymbol{\theta}; \mathbf{0}, \bar{\theta}^2 \mathbb{1}_n)$, i.e.

$$\text{prob}(\boldsymbol{\theta}|I) = \prod_{i=0}^{n-1} \text{prob}(\theta_i|I) = \prod_{i=0}^{n-1} \frac{1}{\sqrt{2\pi\bar{\theta}^2}} \exp\left(-\frac{\theta_i^2}{2\bar{\theta}^2}\right) \tag{2.19}$$

where $\bar{\theta} = 5$ is the width of the distribution. In this way the parameters are favoured by the prior to the interval ± 5 (roughly) and thereby stay close to natural size. The posterior obtained as the product of the MaxEnt-likelihood (2.18) and the naturalness prior (2.19) is visualized in Figure 2.2 for a model with $n = 3$. This is done by presenting its marginal pdfs for different parameter subsets. The posterior is in this case a Gaussian, as can be derived analytically from the relatively simple forms of the likelihood and the prior. In general, analytical expressions are impossible to find and one has to resort to numerical sampling methods such as MCMC, which will be discussed in Section 2.2. Along with the distributions, Figure 2.2 contains point and error estimates for the parameters given by their means and standard deviations respectively. The point estimates should be compared to the actual coefficients in Equation (2.17).

The posterior illustrated in Figure 2.2 is obtained given a model with $n = 3$ parameters, i.e. a polynomial of order $n - 1$ according to Equation (2.8). This specific choice might however not be the model that has the most evidence given the data. We can nevertheless try to inform ourselves by employing the principles of model comparison described in Section 2.1.3. It is in this example possible to compute the evidence $\text{prob}(D|\mathcal{M}_n, I)$ for a

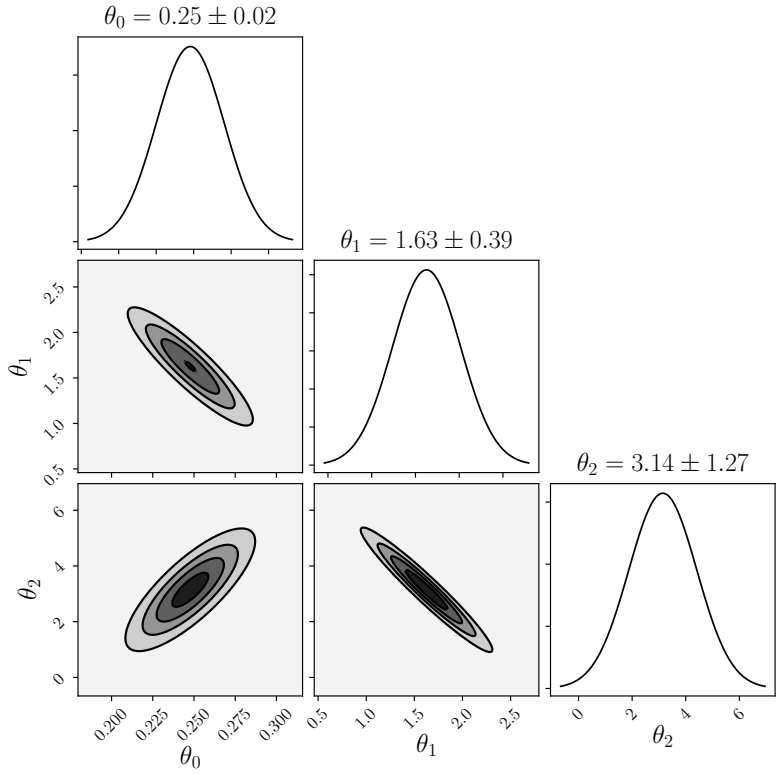


Figure 2.2: Marginal distributions of the analytically derived posterior in the EFT example for a model with $n = 3$ parameters. Two-dimensional distributions for pairs of parameters are located below the diagonal on which the one-dimensional distributions are displayed. Point estimates are given for each parameter in terms of their mean values. Error estimates are given in terms of standard deviations.

model $\mathcal{M}_n$ with n parameters analytically and the result is shown in Figure 2.3. We see clearly that $\mathcal{M}_3$ is in fact the model with the maximum evidence. The fact that the model evidence reaches a plateau for $n \geq 5$ means that not much information is neither gained nor lost by adding or removing a parameter in this region. The evidence integral (2.13) is, however, in general not possible to compute analytically and numerical methods are required. The work of this thesis concerns different versions of such a method, namely nested sampling, which will be introduced in Chapter 3. For this, we first have to acquaint ourselves with a few MCMC sampling methods.

2.2 Markov chain Monte Carlo methods

MCMC methods [5] comprise a set of sampling algorithms widely used in multiple applications, not least in Bayesian inference. They are designed to approximate a target distribution $p(\boldsymbol{\theta})$ by the construction of Markov chains whose equilibrium distributions are that of the target. This is done by letting *walkers* explore parameter space looking for regions of significant probability. Markov chains are created by ensuring that the next position $\boldsymbol{\theta}^{'}$ of each walker only depends on its current position $\boldsymbol{\theta}$ and a transition probability $T(\boldsymbol{\theta}^{'}, \boldsymbol{\theta})$. In contrast to other Monte Carlo methods, MCMC thus produces correlated samples which is important to take into account when e.g. the sample mean and variance are computed. The Markov chain limit theorem [28] provides principles for

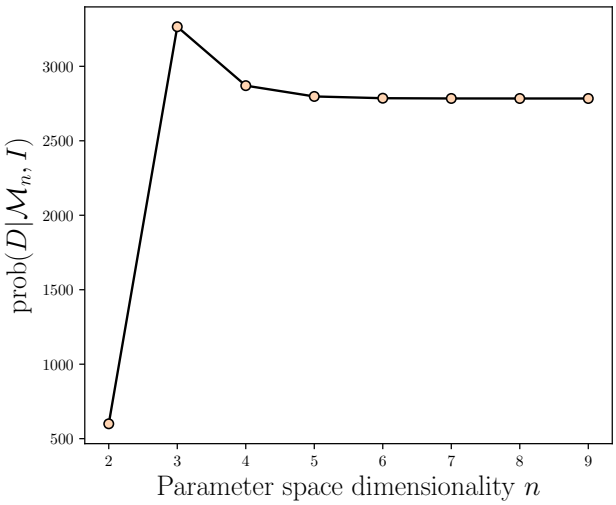


Figure 2.3: The model evidence computed for models $\mathcal{M}_n$ defined by their parameter space dimensionality n. It is clear that the evidence is maximized for $n = 3$ which means that $\mathcal{M}_3$ is the model favoured by the data. The evidence for $n = 1$ is ~ 0 and is therefore not included in the figure.

handling correlated samples and justification for computations performed using MCMC samples. We say that a Markov chain is *ergodic* if its temporal average, i.e. over time steps t, in the long run approaches its ensemble average over all possible states, i.e. if all possible states are eventually explored. This is a key property for MCMC methods for generating samples from the target distribution. Furthermore, a Markov chain is *reversible* if it satisfies *detailed balance*, meaning that the transition probability from $\boldsymbol{\theta}$ to $\boldsymbol{\theta}^{'}$ is the same as from $\boldsymbol{\theta}^{'}$ to $\boldsymbol{\theta}$, i.e. $T(\boldsymbol{\theta}, \boldsymbol{\theta}^{'}) = T(\boldsymbol{\theta}^{'}, \boldsymbol{\theta})$. Detailed balance implies that the Markov chain is *stationary* which means that the distribution of a sample $\boldsymbol{\theta}^{(t)}$ is independent of the time t. Stationarity gives the Markov chain the right properties concerning its equilibrium distribution. We will now proceed by describing a few specific MCMC algorithms used to sample distributions.

2.2.1 Metropolis-Hastings

The Metropolis-Hastings algorithm [6, 7] is an MCMC method for producing samples from a target $p(\boldsymbol{\theta})$. This is done by exploring parameter space by random walks, producing Markov chains. At each step t in a chain, a new position $\boldsymbol{\theta}^{'}$ is proposed, drawn from a proposal distribution $Q(\boldsymbol{\theta}^{'}|\boldsymbol{\theta}^{(t)})$ where $\boldsymbol{\theta}^{(t)}$ is the current position. The proposition is accepted with probability

$$\alpha = \min\left(1, \frac{p(\boldsymbol{\theta}^{'})Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{'})}{p(\boldsymbol{\theta}^{(t)})Q(\boldsymbol{\theta}^{'}|\boldsymbol{\theta}^{(t)})}\right) \tag{2.20}$$

and $\boldsymbol{\theta}^{(t+1)} \leftarrow \boldsymbol{\theta}^{'}$ is set, else $\boldsymbol{\theta}^{(t+1)} \leftarrow \boldsymbol{\theta}^{(t)}$. In the original Metropolis implementation the proposal distribution is taken to be symmetric, meaning $Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{'}) = Q(\boldsymbol{\theta}^{'}|\boldsymbol{\theta}^{(t)})$, and is regularly chosen to be a symmetric Gaussian $\mathcal{N}(\boldsymbol{\theta}^{'}; \boldsymbol{\theta}^{(t)}, \sigma_Q^2 \mathbb{1}_n)$ where the scale σ_Q needs to be set appropriately. The acceptance probability therefore simplifies to $\alpha = \min\left(1, p(\boldsymbol{\theta}^{'})/p(\boldsymbol{\theta}^{(t)})\right)$. In general, the proposal distribution does not have to be

isotropic and can for instance be a Gaussian with an arbitrary $n \times n$ covariance matrix Σ_Q. In this case, there are $\frac{1}{2}n(n+1)$ parameters that need to be set depending on the problem at hand, complicating the user experience. The Metropolis procedure is described in Algorithm 2.1. A caveat of the Metropolis sampling algorithm is that it, like MCMC methods in general, inherently produces correlated samples. This property needs to be properly considered in applications and will be revisited in Section 3.3.1.

Algorithm 2.1: Metropolis procedure.

Input: Target $p(\boldsymbol{\theta})$ and proposal distribution $Q(\boldsymbol{\theta}'|\boldsymbol{\theta})$
Output: Target samples $\boldsymbol{\theta}^{(t)}$
Initialize $\boldsymbol{\theta}^{(0)}$
for $t = 0, ..., N_\mathrm{s} - 1$ **do**
  $\boldsymbol{\theta}' \leftarrow$ sample from $Q(\boldsymbol{\theta}'|\boldsymbol{\theta}^{(t)})$
  $\alpha \leftarrow \min\left(1, \frac{p(\boldsymbol{\theta}')}{p(\boldsymbol{\theta}^{(t)})}\right)$
  $u \leftarrow$ sample from $U(0,1)$
  if $u \leq \alpha$ **then**
    $\boldsymbol{\theta}^{(t+1)} \leftarrow \boldsymbol{\theta}'$
  else
    $\boldsymbol{\theta}^{(t+1)} \leftarrow \boldsymbol{\theta}^{(t)}$

2.2.2 Hamiltonian Monte Carlo

Hamiltonian Monte Carlo (HMC) [3, 5], introduced by Duane et al. (1987) [29], is an MCMC method for generating high-quality samples from challenging target distributions $p(\mathbf{x})$[3], particularly for larger number of dimensions n. The idea is taken from Hamiltonian classical mechanics by considering the logarithm of the target distribution as an energy potential $-V(\mathbf{x})$ in parameter space, i.e.

$$p(\mathbf{x}) \propto \exp(-V(\mathbf{x})). \tag{2.21}$$

The concept of momentum $\mathbf{p}$ is included by considering the phase space extended pdf $\mathrm{prob}(\mathbf{x}, \mathbf{p}) = \mathrm{prob}(\mathbf{x})\mathrm{prob}(\mathbf{p}) = p(\mathbf{x})q(\mathbf{p})$ where we have introduced

$$q(\mathbf{p}) \propto \exp\left(-\frac{|\mathbf{p}|^2}{2}\right), \tag{2.22}$$

which is the exponent of the kinetic energy. The Hamiltonian of a classical system is of the form $H = \frac{1}{2}|\mathbf{p}|^2 + V(\mathbf{x})$, which implies that the full $2n$-dimensional phase space distribution can be written as

$$\mathrm{prob}(\mathbf{x}, \mathbf{p}) \propto \exp(-H). \tag{2.23}$$

Any point (or sample) $(\mathbf{x}, \mathbf{p})$ can thus be evolved in time t for any given period, yielding a new point $(\mathbf{x}', \mathbf{p}')$, by solving Hamilton's equations[4]:

[3] We shall here use $\mathbf{x}$ instead of $\boldsymbol{\theta}$ for the parameters to emphasize the parallel to position in classical mechanics.

[4] These equations need in general to be solved numerically, which is a topic that we will not elaborate on here.

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \frac{\partial H}{\partial \mathbf{p}} = \mathbf{p} \tag{2.24a}$$

$$\frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} = -\frac{\partial H}{\partial \mathbf{x}} = -\nabla V(\mathbf{x}), \tag{2.24b}$$

serving the purpose of the transition probability $T(\mathbf{x}', \mathbf{p}'; \mathbf{x}, \mathbf{p})$. If the momentum $\mathbf{p}$ is randomly initiated at each new position $\mathbf{x}$, these points will form a Markov chain satisfying ergodicity [29]. Detailed balance is satisfied by the time reversal symmetry of Hamilton's equations (2.24). The desired target $p(\mathbf{x})$ is obtained by discarding the $\mathbf{p}$-samples, only keeping the $\mathbf{x}$-part of phase space. These samples should be of high quality as the trajectories described by the dynamics of Equation (2.24) are favoured to move towards regions of low potential energy, i.e. high probability. This is especially useful in large-dimensional spaces out of which these high-quality regions usually make up a small fraction, making them hard to locate. The HMC algorithm also covers distance in parameter space faster than random walks such as Metropolis-Hastings, effectively decreasing the correlation of samples. The most noticeable drawback of HMC is that it requires evaluation of the gradient $-\nabla V(\mathbf{x}) = \nabla \ln p(\mathbf{x})$ at every step of a trajectory. This is in general a very computationally expensive and thus time consuming task. HMC furthermore contains a number of hyperparameters, such as the number of time steps and the size of the steps, that need to be appropriately set for your current problem.

2.2.3 Sampling with affine invariance: the stretch move

Model parameters can, in general, be strongly correlated, making their joint probability distributions highly asymmetric. This means that the characteristic length scales and relevant directions can differ largely between parameter dimensions making efficient exploration of parameter space more challenging. Building upon work by Christen [30], Goodman and Weare [31] proposed an MCMC sampling method that is independent of scale differences between dimensions. Formally, the algorithm is invariant under *affine transformations*, which in n dimensions are of the form

$$y_i = \beta_{i0}\theta_0 + ... + \beta_{in-1}\theta_{n-1} = \sum_{j=0}^{n-1} \beta_{ij}\theta_j, \quad i = 0, ..., n-1, \tag{2.25}$$

where θ_j are the parameters of the problem, β_{ij} are scale coefficients and y_i are the corresponding transformed parameters. Affine invariance of the algorithm implies that parameter dimensions are effectively scaled to the same size, simplifying the shape of the distribution making parameter space easier to explore. To achieve this, the algorithm utilizes a special kind of random walk, informally referred to as the *stretch move*. This method simultaneously evolves an ensemble of K walkers $S = \{\Theta_k\}_{k=1}^K$ where the proposal distribution for walker Θ_k depends on the complementary ensemble $S_{[k]} = \{\Theta_j,\ \forall j \neq k\}$.

At step t, a new position Y is proposed for walker Θ_k by choosing another walker Θ_j randomly from $S_{[k]}$ and setting

$$\Theta_k^{(t)} \to Y = \Theta_j + \tilde{\zeta}[\Theta_k^{(t)} - \Theta_j] \tag{2.26}$$

where $\tilde{\zeta}$ is a scale factor randomly sampled from a distribution $g(\zeta)$. For the proposal to be symmetric and satisfy detailed balance we must have $g(\zeta^{-1}) = \zeta g(\zeta)$ as well as add a Metropolis style rejection scheme where $\Theta_k^{(t+1)} \leftarrow Y$ is accepted with probability

$$\alpha = \min\left(1, \tilde{\zeta}^{n-1}\frac{p(Y)}{p(\Theta_k^{(t)})}\right), \tag{2.27}$$

where $p(\boldsymbol{\theta})$ is the target distribution. This proposal procedure is repeated for every walker in S resulting in a collective evolution.

As an explicit form for $g(\zeta)$, Goodman and Weare [31] suggests

$$g(\zeta) \propto \frac{1}{\sqrt{\zeta}}, \quad \text{for} \quad \zeta \in \left[\frac{1}{a}, a\right], \tag{2.28}$$

where a is a hyperparameter adjusting the scale of $g(\zeta)$ and which they set to 2. One main advantage of this method is that there are very few hyperparameters that need to be set by the user, there are essentially only two: step scale a and the number of steps. A state-of-the-art implementation of the stretch move procedure for MCMC sampling can be found in e.g. the Python module `emcee` [32].

The MCMC methods described in the previous sections provide a set of samples that can be used to make computations — such as the model evidence — with respect to the posterior. However, this procedure quickly becomes inefficient for large-scale problems and alternative approaches are necessary. In the next chapter we introduce such an approach: nested sampling.

3

Nested sampling

Introduced by Skilling [8], nested sampling is a Monte Carlo sampling method that naturally provides an estimate of the evidence Z. It is however important to note that nested sampling is not an MCMC method as it does not produce Markov chains distributed according to the posterior. The acquisition of posterior samples is nevertheless also a product but not the main focus of the algorithm. Here follows a review of the principles of nested sampling, for a detailed description see e.g. Sivia and Skilling [4] or Skilling [9].

3.1 Background

Let us first repeat some Bayesian key concepts from Chapter 2. Given data D and other information I, such as the choice of model, the goal is to obtain the evidence (or marginal likelihood) $Z = \text{prob}(D|I)$ but also the posterior $\mathcal{P}(\boldsymbol{\theta}) = \text{prob}(\boldsymbol{\theta}|D, I)$ from the prior $\pi(\boldsymbol{\theta}) = \text{prob}(\boldsymbol{\theta}|I)$ and the likelihood $\mathcal{L}(\boldsymbol{\theta}) = \text{prob}(D|\boldsymbol{\theta}, I)$. The relation between these quantities is given through Bayes' theorem which in this more compact notation takes the form

$$\mathcal{L}(\boldsymbol{\theta}) \times \pi(\boldsymbol{\theta}) = Z \times \mathcal{P}(\boldsymbol{\theta}). \tag{3.1}$$

Assuming that the posterior distribution is properly normalized to unity, i.e. $\int \mathcal{P}(\boldsymbol{\theta}) \mathrm{d}^n\boldsymbol{\theta} = 1$, the evidence can be written in terms of the integral

$$Z = \int \mathcal{L}(\boldsymbol{\theta})\pi(\boldsymbol{\theta}) \mathrm{d}^n\boldsymbol{\theta} \tag{3.2}$$

where the integration is taken over the parameter ranges according to the prior. The complexity of computing this integral increases exponentially with the number of dimensions as a space with n dimensions resolved to one part in R has R^n volume elements. High dimensionality furthermore obstructs the use of analytical approximations as the degree of freedom in these spaces becomes too large [4].

3.1.1 Prior mass fraction and sorted likelihood function

What makes the numerical computation of integrals such as Equation (3.2) challenging is the brute-force approach of direct summation of the exponentially large number of elements in parameter space. However, the likelihood values associated to each of these elements can be sorted in a decreasing order, resulting in a sequence $\{\mathcal{L}_k\}$. The sequence could be enumerated using a global variable encoding the information of the part of prior parameter space with likelihoods larger than $\mathcal{L}_k$. This variable is the prior mass fraction, here denoted ξ. In the case of continuous parameters $\boldsymbol{\theta}$, the prior mass fraction as a function of likelihood limit is formally defined as

$$\xi(\lambda) = \int_{\mathcal{L}(\boldsymbol{\theta})>\lambda} \pi(\boldsymbol{\theta}) \mathrm{d}^n\boldsymbol{\theta} \tag{3.3}$$

and is interpreted as the fraction of prior probability with likelihood greater than λ [4]. Assuming the prior $\pi(\boldsymbol{\theta})$ to be normalized to unity, the prior mass fraction has by definition properties: $0 \leq \xi \leq 1$, $\xi(0) = 1$ and $\xi(\mathcal{L}_{\max}) = 0$ where $0 < \mathcal{L} \leq \mathcal{L}_{\max}$. Also by definition, the mass element associated with likelihoods $\lambda \leq \mathcal{L} \leq \lambda + \mathrm{d}\lambda$ is given by $\mathrm{d}\xi = \pi(\boldsymbol{\theta})\mathrm{d}^n\boldsymbol{\theta}$. In this way the problem is transformed from n-dimensional parameter space, in terms of the parameter vector $\boldsymbol{\theta} = (\theta_0, ..., \theta_{n-1})$, to a single variable ξ. There is no information lost in this reduction of dimensions if ξ is stored at a resolution of one part in R^n and each of the n parameters θ_i is stored at a resolution of one part in R as the information required in both cases amounts to $n \log_2 R$ bits [4].

For the sake of convenience, it is rather the inverse function $L(\xi)$, defined by

$$L(\xi(\lambda)) = \lambda, \tag{3.4}$$

that we shall consider in practice. It takes on values $0 \leq L \leq \mathcal{L}_{\max}$ where the extremes are at $L(0) = \mathcal{L}_{\max}$ and $L(1) = 0$. $L(\xi)$ is a sorted version of the original likelihood $\mathcal{L}(\boldsymbol{\theta})$ and only has a single variable dependence ξ. The existence of the inverse is not trivial to prove. In fact, it has been proven by Schittenhelm and Wacker [33] that the statement (3.4) is violated for specific forms of $\mathcal{L}(\boldsymbol{\theta})$. For the likelihood functions in this work, however, the definition of the inverse $L(\xi)$ will always hold.

A cartoon representing the relation between the likelihood versions $\mathcal{L}(\boldsymbol{\theta})$ and $L(\xi)$ can be seen in Figure 3.1a. The left panel shows an example of a likelihood in two dimensions where each of the iso-likelihood contours $\mathcal{L}(\boldsymbol{\theta}) = \mathcal{L}_k$, $k = 1, ..., 4$, corresponds to a prior mass fraction ξ_k shown in the right panel. In this example, ξ_k is simply the integral of the prior over the area enclosed by the contour $\mathcal{L}_k$. The filled in areas illustrate what parts of parameter space correspond to what ξ-intervals. Note that the figure is for illustrative purposes and is not quantitatively correct.

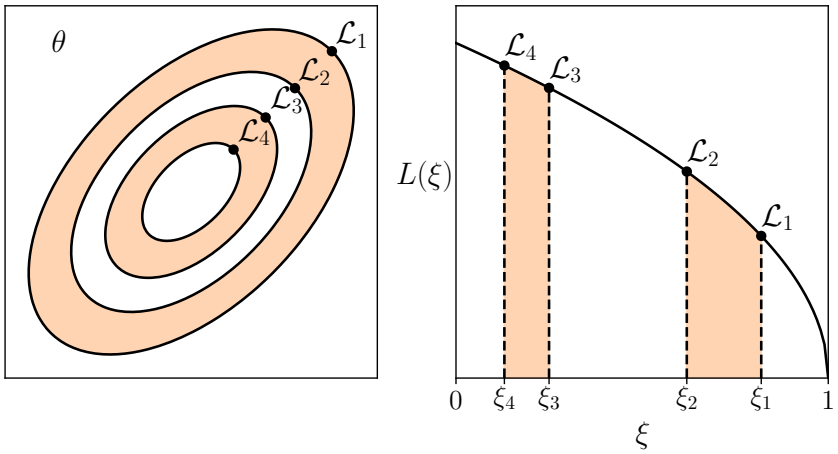
The key idea setting up the nested sampling method is to express the evidence Z in terms of the quantities L and ξ. The evidence (3.2) is a sum of likelihood values $\mathcal{L}(\boldsymbol{\theta})$ with associated prior mass weights $\pi(\boldsymbol{\theta})\mathrm{d}^n\boldsymbol{\theta}$ over volume elements $\mathrm{d}^n\boldsymbol{\theta}$. The mass elements $\mathrm{d}\xi$ originate from the same volume elements meaning the evidence is equivalently given by

$$Z = \int_0^1 L(\xi)\mathrm{d}\xi \tag{3.5}$$

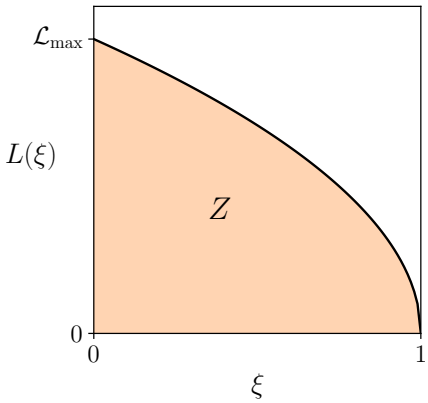
which notably is a one-dimensional integral. Geometrically, this means that Z is the area enclosed by the curve generated by $L(\xi)$ as shown in Figure 3.1b. Furthermore, $P(\xi)$ is the ξ-counterpart to the posterior $\mathcal{P}(\boldsymbol{\theta})$ and is proportional to $L(\xi)$ according to

$$P(\xi) = \frac{1}{Z} L(\xi) \tag{3.6}$$

from which a sample $\tilde{\xi}$ corresponds to a parameter sample $\tilde{\boldsymbol{\theta}}$ from $\mathcal{P}(\boldsymbol{\theta})$. This is true from the previous argument that mass elements $\mathrm{d}\xi$ and $\pi(\boldsymbol{\theta})\mathrm{d}^n\boldsymbol{\theta}$ come from the same volume element. The sorted likelihood function $L(\xi)$ is thus the primary constituent in the nested sampling machinery and can be used to yield Z as well as samples from $\mathcal{P}(\boldsymbol{\theta})$.



(a) Cartoon of the relation between iso-likelihood contours $\mathcal{L}(\boldsymbol{\theta}) = \mathcal{L}_k$ and prior mass fractions ξ_k, described by $L(\xi_k) = \mathcal{L}_k$. Each interval in ξ corresponds to an area in parameter space. Note that the figure is not meant to be quantitatively correct.



(b) Representation of the evidence Z as the area under the curve of the likelihood function $L(\xi)$.

Figure 3.1: Illustrations of the relationship between the likelihoods $\mathcal{L}(\boldsymbol{\theta})$ and $L(\xi)$, the prior mass fraction ξ and the model evidence Z.

3.2 The algorithm

The nested sampling algorithm makes use of active points exploring parameter space to reveal the structure of $L(\xi)$. N points are sampled from the parameter space in proportion to the prior and are then set to evolve under successively stricter likelihood constraints $\mathcal{L}(\boldsymbol{\theta}) > L^*$, where $L^* \geq 0$ is an evolving likelihood limit. These constraints iteratively define the iso-likelihood contours seen in Figure 3.1a. In terms of ξ this corresponds to N uniformly distributed samples evolving subject to the constraint $\xi < \xi^* = \xi(L^*)$. Each iteration hence starts with N points uniformly distributed in $(0, \xi^*)$[1] (initially $\xi^* = 1$) and the subsequent evolution, illustrated in Figure 3.2, proceeds as follows:

(a) The worst of the N active points, the one with the lowest likelihood, is identified, removed and stored for later use.
(b) Limit ξ^* (and therefore also L^*) is updated to the previous position of the removed point.
(c) A new point, *nested* within ξ^*, is generated and added to the set of active points.

This procedure is set to repeat for a pre-defined number of iterations M. The way in which item (c) above is implemented is a subject of particular importance and will be further discussed in Section 3.3. Sampling from the prior under the likelihood-constraint could be argued to be the most crucial subtopic of nested sampling and will be the main focus of this thesis. For now it is assumed to be possible to generate such new points.

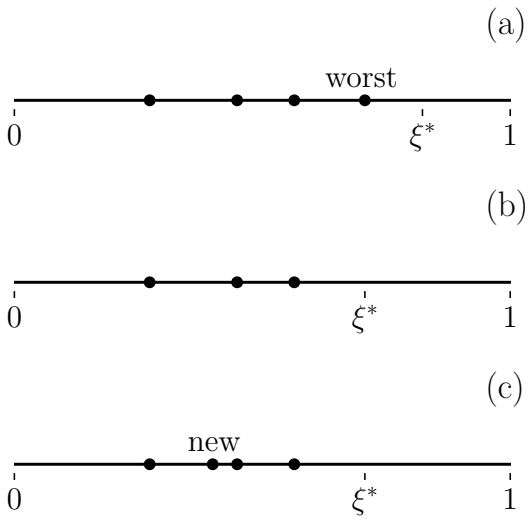


Figure 3.2: Evolution of active points in a nested sampling iteration. In (a) the worst point is identified and removed. The limit ξ^* is updated in (b) to the position the point was removed from. A new point is generated in (c), nested within the current limit.

3.2.1 Statistical properties

At this stage it is necessary to address the topic that the ξ's are not explicitly known. It is however possible to determine their statistical properties from which an estimate can be extracted. If the worst point removed from iterate k is denoted ξ_k, the shrinkage ratio

[1] A procedure for validating this assumption of uniformity has been suggested by Fowlie et al. [34].

of the active volume between iterations is

$$t_k = \xi_k/\xi_{k-1}, \quad \text{where} \quad \xi_0 = 1 \Longrightarrow t_1 = \xi_1. \tag{3.7}$$

Noting that ξ_k is the topmost of N uniformly distributed samples between 0 and $\xi^* = \xi_{k-1}$, the ratio t_k can be considered to be the topmost of N samples uniformly distributed between 0 and 1. This means that the properties of order statistics [35] can be used to show that $t_k \sim \text{Beta}(N, 1)$[2] with pdf

$$\text{prob}(t_k) = N t_k^{N-1}. \tag{3.8}$$

Given the distribution (3.8), the associated mean and variance of $\ln t_k$ are

$$\mathbb{E}[\ln t_k] = -\frac{1}{N} \quad \text{and} \quad \text{Var}[\ln t_k] = \frac{1}{N^2} \tag{3.9}$$

respectively. Using the (natural) logarithm rather than the plain value in (3.9) ensures better properties in the transition from full prior space to smaller regions where the bulk of the posterior mass resides [9]. At iterate k, ξ_k can be expressed in terms of a product of shrinkage ratios, i.e.

$$\xi_k = t_1 t_2 ... t_k = \prod_{j=1}^{k} t_j \tag{3.10}$$

leading to a geometrical progression of the active ξ-range (but linear in $\ln \xi$). The properties of the ξ's are thereby determined entirely by the properties of the t's and it is straightforward to derive the mean and variance

$$\mathbb{E}[\ln \xi_k] = \sum_{j=1}^{k} \mathbb{E}[\ln t_j] = -\frac{k}{N} \quad \text{and} \quad \text{Var}[\ln \xi_k] = \frac{k}{N^2}, \tag{3.11}$$

of the logarithm $\ln \xi_k$. Crudely adopting the mean in Equation (3.11) as an estimator one obtains an exponentially decreasing sequence of discarded prior mass fractions

$$\xi_k = e^{-k/N}, \tag{3.12}$$

illustrated in Figure 3.3. That is to say that with this scheme the active range shrinks exponentially as regions of higher likelihoods are approached. Along with each ξ_k is an associated likelihood L_k that together compose a list of pairs $\{(\xi_k, L_k)\}_{k=1}^{M}$ which is the main output of the algorithm and used to compute the sought after quantities.

3.2.2 Obtaining the evidence and posterior

The evidence Z is estimated by approximating the integral expression of Equation (3.5) by a weighted sum using the list of samples $\{(\xi_k, L_k)\}_{k=1}^{M}$ according to

$$Z = \int_0^1 L(\xi) \mathrm{d}\xi \approx \sum_{k=1}^{M} L_k \Delta\xi_k. \tag{3.13}$$

Straight-forward estimation of the mass element by $\Delta\xi_k = \xi_{k-1} - \xi_k$ implies a numerical integration error of $\mathcal{O}(N^{-1})$. Any other valid numerical integration method would also be an option, such as the trapezoidal rule $\Delta\xi_k = \frac{1}{2}(\xi_{k-1} - \xi_{k+1})$ which lowers the error

[2] Sample from Beta distribution with parameters N and 1.

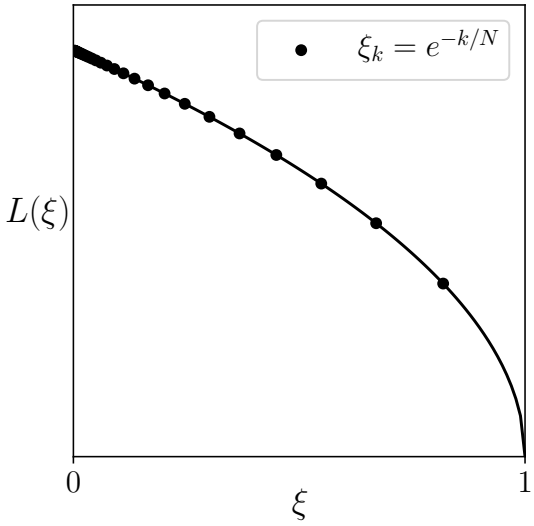


Figure 3.3: Sequence of discarded points obtained by estimating the prior mass fractions ξ_k statistically. The data is artificially produced.

to $\mathcal{O}(N^{-2})$ in most cases [9]. Improvement from using the trapezoidal rule will however be small as the principal source of error is assumed to originate from the crude estimate $\xi_k = e^{-k/N}$ that will be discussed further in Section 3.2.5.

Along with each pair (ξ_k, L_k) the algorithm also produces a parameter sample $\boldsymbol{\theta}_k$ with an associated probability weight

$$w_k = \frac{1}{Z} L_k \Delta\xi_k. \tag{3.14}$$

Together $\boldsymbol{\theta}_k$ and w_k form a list of weighted samples that can be used to estimate any quantity $\mathcal{Q}(\boldsymbol{\theta})$ with respect to the posterior via

$$\mathbb{E}[\mathcal{Q}] = \int \mathcal{Q}(\boldsymbol{\theta})\mathcal{P}(\boldsymbol{\theta})\mathrm{d}^n\boldsymbol{\theta} \approx \sum_{k=1}^{M} w_k \mathcal{Q}(\boldsymbol{\theta}_k) \tag{3.15}$$

The shape of the posterior can be extracted by binning the obtained samples $\boldsymbol{\theta}_k$, creating a histogram that can be visualized by plotting its one- and/or two-dimensional projections. As opposed to MCMC samples, the samples produced by nested sampling are not equally weighted implying that their contribution to the bin values (or "heights") should be proportional to their associated weights w_k. We also see this in the formula for the expectation value (3.15), which takes into account that the samples are unequally weighted. It is however possible to extract an equally weighted subset of the nested samples making storage more convenient. One method for extracting samples with equal weights is *staircase sampling* [4].

In summary, the nested sampling procedure described above provides natural means of both computing the evidence Z and revealing the structure of the posterior distribution $\mathcal{P}(\boldsymbol{\theta})$. An additional quantity, relevant for post-analysis, is the information

$$\mathcal{H} = \mathcal{P}(\boldsymbol{\theta}) \ln \left[\frac{\mathcal{P}(\boldsymbol{\theta})}{\pi(\boldsymbol{\theta})} \right] \mathrm{d}^n \boldsymbol{\theta} = \int_0^1 P(\xi) \ln P(\xi) \mathrm{d}\xi \approx \sum_{k=1}^{M} w_k \ln \left[\frac{L_k}{Z} \right] \tag{3.16}$$

being a logarithmic measure of the prior-to-posterior shrinkage. A useful interpretation is $\mathcal{H}$ = "information gained going from the prior to the posterior" [36]. The information $\mathcal{H}$ (3.16) is in the nested sampling context used to formulate an uncertainty estimation which we shall discuss in Section 3.2.5.

3.2.3 Pseudocode

Algorithm 3.1 portrays the basic principle of the nested sampling procedure in pseudocode format. Nested sampling is performed M times, pre-defined by the user. In practice it is preferable to automate the termination by imposing a condition for when to stop sampling, this will be further discussed next in Section 3.2.4.

Algorithm 3.1: Nested sampling procedure.

Input: Prior $\pi(\boldsymbol{\theta})$ and likelihood $\mathcal{L}(\boldsymbol{\theta})$
Output: Evidence Z and posterior samples $\boldsymbol{\theta}_k$ with weights w_k
Initialization
  Draw N active points $\boldsymbol{\theta}_1^{\mathrm{a}}, ..., \boldsymbol{\theta}_N^{\mathrm{a}}$ from the prior $\pi(\boldsymbol{\theta})$
  Set $Z \leftarrow 0$ and $\xi_0 \leftarrow 1$
for $k = 1, 2, ..., M$ **do**
  $\boldsymbol{\theta}_k \leftarrow \boldsymbol{\theta}_{\mathrm{worst}}^{\mathrm{a}}$, point in the current set $\boldsymbol{\theta}_1^{\mathrm{a}}, ..., \boldsymbol{\theta}_N^{\mathrm{a}}$ with lowest likelihood
  $L_k \leftarrow \mathcal{L}(\boldsymbol{\theta}_k)$, likelihood bound
  $\xi_k \leftarrow \exp(-k/N)$
  $\Delta\xi_k \leftarrow \xi_{k-1} - \xi_k$ or e.g. $\frac{1}{2}(\xi_{k-1} - \xi_{k+1})$
  $w_k \leftarrow L_k \Delta\xi_k$
  $Z \leftarrow Z + w_k$
  Replace $\boldsymbol{\theta}_{\mathrm{worst}}^{\mathrm{a}}$ with an independent sample from $\pi(\boldsymbol{\theta})$ obeying $\mathcal{L}(\boldsymbol{\theta}) > L_k$
Normalize weights, i.e. $w_k \leftarrow w_k / Z$

3.2.4 Termination

The main nested sampling loop can be set to terminate after a fixed number of steps M. It may however be desirable to implement a more well-motivated stopping criterion. One such approach is to stop when the remaining contribution to Z is smaller than some (small) fraction f. At iteration k, the remaining contribution is taken to be bounded from above by the maximum likelihood of the set of active points $L_{\mathrm{max}}^{\mathrm{a}} = \max(\mathcal{L}(\boldsymbol{\theta}_1^{\mathrm{a}}), ..., \mathcal{L}(\boldsymbol{\theta}_N^{\mathrm{a}}))$ multiplied by the current prior mass ξ_k. The condition then reads

$$L_{\mathrm{max}}^{\mathrm{a}} \xi_k < f Z_k \Longrightarrow \textbf{Termination} \tag{3.17}$$

where Z_k is the evidence accumulated at iteration k. If $L_{\mathrm{max}}^{\mathrm{a}}$ is close to the true maximum likelihood, this stopping criterion implies that approximately all but a fraction f of the evidence has been accounted for. The accumulation of Z usually begins to increase as regions of higher likelihoods are found and flattens out when the bulk of the posterior is reached and the decrease in range $\Delta\xi_k \propto e^{-k/N}$ starts to dominate increases in L_k. The amount of prior mass associated to the region of the posterior bulk is roughly estimated

by $\xi \approx e^{-\mathcal{H}}$ which in general could be very small. Recalling that $\xi_k = e^{-k/N}$, the number of steps taken to the bulk according to Equation (3.11) is $N\mathcal{H} \pm \sqrt{N\mathcal{H}}$, resembling the mean and standard deviation of a Poisson distribution.

In practice — where computations are performed in terms of logarithmic quantities for numerical stability — the condition is implemented in the form

$$\ln(Z_k + L^{\mathrm{a}}_{\max}\xi_k) - \ln(Z_k) = \ln\left(1 + \frac{L^{\mathrm{a}}_{\max}\xi_k}{Z_k}\right) < f_{\ln} \Longrightarrow \textbf{Termination}. \tag{3.18}$$

Equation (3.18) is equivalent to Equation (3.17) if $f_{\ln} = \ln(1 + f)$ which means that if $f \ll 1$ then $f_{\ln} \approx f$.

3.2.5 Uncertainty estimation

The major source of uncertainty is assumed to stem from the variability in the number of nested sampling steps taken to reach the bulk of the posterior mass $N\mathcal{H} \pm \sqrt{N\mathcal{H}}$. This corresponds to an uncertainty in $\ln \xi$ of $\sqrt{N\mathcal{H}}/N = \sqrt{\mathcal{H}/N}$ which through Equation (3.13) also gives an uncertainty in $\ln Z$ of $\sqrt{\mathcal{H}/N}$. The full expression for $\ln Z$ including the uncertainty estimate is thus

$$\ln Z \approx \ln\left(\sum_{k=1}^{M} L_k \Delta\xi_k\right) \pm \sqrt{\frac{\mathcal{H}}{N}}. \tag{3.19}$$

Expressing the uncertainty as an additive term to $\ln Z$ rather than as a geometrical factor $e^{\pm\sqrt{\mathcal{H}/N}}$ to Z is common practice since the distribution in $\ln Z$ should have better properties in terms of symmetry and similarity to a Gaussian. It would also be disadvantageous to translate the uncertainty to an additive deviation in Z as it would make possible for negative Z-values to be within this range. As an example, an assumed normally distributed $\ln Z = 100 \pm 10$ would be naively translated to $Z = e^{150} \pm e^{200}$, which obviously is a useless statement.

3.3 Generating new points

Until now the details of how to generate new points as the nested sampling algorithm progresses have been left out. This task is at the very core of the method and is of utmost importance for obtaining useful results. Each iteration requires a new sample $\boldsymbol{\theta}$ which needs to:

(a) be sampled in proportion to the prior $\pi(\boldsymbol{\theta})$
(b) be (approximately) independent from other samples
(c) obey the current likelihood constraint $\mathcal{L}(\boldsymbol{\theta}) > L^*$.

These requirements make the task of generating new points non-trivial and worthy of special attention. The likelihood constraint (c) stands out as it exponentially reduces the range of active prior mass to explore as sampling progresses. In many sampling scenarios one often resorts to MC methods in general and MCMC methods in particular (see Section 2.2). This approach is available also in this case but under the addition of the likelihood constraint. For the remaining part of this thesis the main topic will be MCMC methods adjusted to generate samples from the constrained prior in the context of nested sampling. Modified versions of MCMC methods for constrained prior sampling

are introduced in the following sections and are further described and implemented in Chapter 4.

3.3.1 Constrained Metropolis

The standard Metropolis-Hastings algorithm, described in Section 2.2.1, is likely the most well known among MCMC methods. As suggested by Sivia and Skilling [4] and followed up by Feroz and Hobson [11] it can be applied in the context of nested sampling by simply adding the likelihood constraint $\mathcal{L}(\boldsymbol{\theta}) > L^*$ to the acceptance conditions. At each nested sampling iteration, one of the active points $\boldsymbol{\theta}$ is picked at random as the start of a random walk. A step to a new point $\boldsymbol{\theta}'$ is generated from a symmetric proposal distribution $Q(\boldsymbol{\theta}'|\boldsymbol{\theta})$ and accepted with probability

$$\alpha = \begin{cases} \min\left(1, \frac{\pi(\boldsymbol{\theta}')}{\pi(\boldsymbol{\theta})}\right), & \text{if } \mathcal{L}(\boldsymbol{\theta}') > L^* \\ 0, & \text{otherwise.} \end{cases} \tag{3.20}$$

The proposal distribution is usually an isotropic Gaussian $Q(\boldsymbol{\theta}'|\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}'; \boldsymbol{\theta}, \sigma_Q^2 \mathbb{1}_n)$ where the scale σ_Q is a free parameter that needs to be tuned in relation to the problem. However, it is fully possible to use a Gaussian with a general covariance matrix Σ_Q. As the Metropolis algorithm produces correlated samples a sufficiently large number of steps N_s needs to be taken in order for the new sample to grow independent of its starting point. This might be a cause of concern as the active prior volume shrinks at each iteration and acceptable Metropolis steps become more rare. However, Sivia and Skilling [4] suggests that one should always use $N_\mathrm{s} \approx 20$, a recommendation that we shall evaluate later. Since the allowed region shrinks in each nested sampling iteration it would be unwise to keep the distribution scale σ_Q, or scales Σ_Q, constant as it would make it increasingly harder to propose acceptable positions. This would cause the acceptance rate to drop drastically making the exploration very inefficient. A scheme for adjusting the step scale will be suggested in Section 4.2.

3.3.2 Galilean Monte Carlo

A likelihood-constrained version of the HMC algorithm has been suggested by Betancourt [37] to produce high-quality samples from the prior in the context of nested sampling. In constrained HMC, the likelihood contour is interpreted as a hard wall, or infinite potential barrier in the terms of Hamiltonian mechanics, with a potential of the form

$$V_\mathrm{cHMC}(\mathbf{x}) = \begin{cases} -\ln \pi(\mathbf{x}), & \text{if } \mathcal{L}(\mathbf{x}) > L^* \\ \infty, & \text{otherwise.} \end{cases} \tag{3.21}$$

The time evolution of the trajectories is therefore adjusted such that they will specularly reflect if they were to encounter the $\mathcal{L}(\mathbf{x}) = L^*$ barrier. As a starting point for the Hamiltonian evolution, one of the active points $\mathbf{x}$ of the current nested sampling iteration is chosen at random, ensuring that the constraint will be satisfied also for a new point $\mathbf{x}'$ if the reflection is properly conducted. This procedure, however, requires the gradient of the prior to be evaluated in every time step. In fact, for a typical problem the prior is often at least approximately constant in regions where the likelihood varies significantly. This means that the prior gradient will be small and that it will not influence the trajectories enough to make it worth while computing.

As a simpler, more straight forward successor to constrained HMC, Skilling suggests Galilean Monte Carlo (GMC) [19, 38, 39]. Time evolution in GMC is performed without the influence of any forces, and is therefore without the need for prior gradient evaluation. For the sake of convenience and presentation we shall without loss of generality adopt coordinates in which the prior is flat, i.e. $\pi(\mathbf{x}) \equiv 1$, over the unit hypercube. Such a coordinate transformation is always possible [38] but may however be quite impractical, depending on the form of $\pi(\mathbf{x})$. This transformation is discussed in more detail in Section 4.1.

GMC trajectories will proceed as straight lines[3] as long as they are within the allowed region according to the potential

$$V_{\mathrm{GMC}}(\mathbf{x}) = \begin{cases} \text{const.}, & \text{if } \mathcal{L}(\mathbf{x}) > L^* \\ \infty, & \text{otherwise.} \end{cases} \tag{3.22}$$

Time evolution is particularly simple as a phase space point $(\mathbf{x}, \mathbf{v})$, where $\mathbf{v}$ is its velocity[4], will in one time step move to a new point

$$(\mathbf{x}, \mathbf{v}) \to (\mathbf{x}', \mathbf{v}) = (\mathbf{x} + \tau\mathbf{v}, \mathbf{v}) \quad \text{(proceed)}, \tag{3.23}$$

where τ is the size of the time step. This step is acceptable under the assumption that $\mathbf{x}'$ is still within the available region. We consequently need a mechanism to reroute and return a point back into the active region if the step according to (3.23) were to cross the likelihood-barrier. A natural way to do this is by letting the point reflect specularly from the iso-likelihood surface by using that the surface normal vector $\mathbf{n}$ is proportional to the gradient of the likelihood $\nabla\mathcal{L}$, or equivalently to the gradient of the log-likelihood $\nabla \ln \mathcal{L}$. A reflection of this sort, occurring at point $(\mathbf{x}', \mathbf{v})$, will transform $(\mathbf{x}, \mathbf{v})$ according to

$$(\mathbf{x}, \mathbf{v}) \to (\mathbf{x}'', \mathbf{v}') = (\mathbf{x} + \tau\mathbf{v} + \tau\mathbf{v}', \mathbf{v} - 2\mathbf{n}\mathbf{n}^{\mathrm{T}}\mathbf{v}) \quad \text{(reflect)}, \tag{3.24}$$

where $\mathbf{n} = \nabla \ln \mathcal{L}(\mathbf{x}')/|\nabla \ln \mathcal{L}(\mathbf{x}')|$. A schematic illustration of the GMC exploration including reflections off the hard walls is shown in Figure 3.4a. It is obviously unlikely for $\mathbf{x}'$ to lie exactly on the iso-likelihood surface but the gradient at $\mathbf{x}'$ will work as a substitute for the gradient at the ideal reflection point located somewhere between $\mathbf{x}$ and $\mathbf{x}'$. The reflected point $\mathbf{x}''$ will nevertheless often be returned to the active region making it possible for the trajectory to continue with a redirected velocity. This reflection by proxy scheme is illustrated in Figure 3.4b. In the assumably rare case that also $\mathcal{L}(\mathbf{x}'') > L^*$ is violated a possible option satisfying detailed balance is to reverse the trajectory according to

$$(\mathbf{x}, \mathbf{v}) \to (\mathbf{x}, -\mathbf{v}) \quad \text{(reverse)}. \tag{3.25}$$

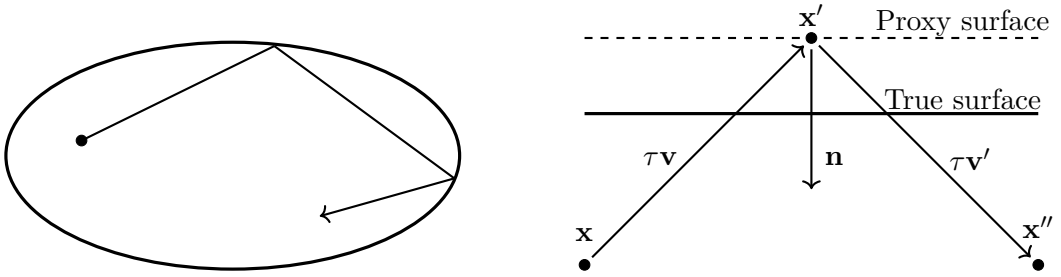
Detailed balance actually allows an additional option in the reflection's mirror point

$$(\mathbf{x}, \mathbf{v}) \to (\mathbf{x} + \tau\mathbf{v} - \tau\mathbf{v}', -\mathbf{v}') \quad \text{(mirrored reflection)}, \tag{3.26}$$

which in most cases should lie outside of the allowed region but is however still possible. For detailed balance (time-reversal symmetry) to remain satisfied under the inclusion of the mirrored reflection, either the reflection or the mirrored reflection should be acceptable if a trajectory is to redirect, but not both. If both were acceptable, the time reversed

[3] When the coordinates are transformed back to the original prior space the trajectories will follow geodesics determined by the geometry of the prior which in general are not straight lines.

[4] We will for GMC adopt the notion of velocity $\mathbf{v}$ to distinguish it from the HMC momentum $\mathbf{p}$.



(a) Bird's-eye view illustration of a typical GMC trajectory propagating in a constrained region of parameter space.

(b) Detailed illustration of the GMC reflection off the proxy iso-likelihood surface.

Figure 3.4: Illustrations of the basic principles of Galilean Monte Carlo.

trajectory would break detailed balance.

As the active region shrinks during the nested sampling iterations it is necessary to adjust the effective size of the GMC steps accordingly. If no adjustment of the step size would be carried out, fewer and fewer proposed steps will be accepted making the exploration increasingly inefficient. A GMC implementation is presented in Section 4.3 and includes a suggested scheme for adjusting the effective step size.

3.3.3 Constrained stretch move

As previously stressed, one of the key difficulties of nested sampling is to produce samples from the prior for successively smaller regions of parameter space. Furthermore, this region is typically highly asymmetric, spanning significantly different distances in different dimensions. This problem of varying and separated scales is exactly what the stretch move algorithm, outlined in Section 2.2.3, is designed to manage due to its affine invariance. We therefore propose a constrained version of the stretch move, adapted for the context of nested sampling. The stretch move is dependent on an ensemble of walkers; a requirement which is naturally provided by the active set of N points used in nested sampling. At each iteration one of the active points Θ_k is randomly selected to perform the stretch move random walk where one of the remaining points Θ_j is used to propose a new position Y according to Equation (2.27). Each step is then accepted with the probability

$$\alpha = \begin{cases} \min\left(1, \tilde{\zeta}^{n-1} \frac{\pi(Y)}{\pi(\Theta_k)}\right), & \text{if } \mathcal{L}(\Theta_k) > L^* \\ 0, & \text{otherwise} \end{cases} \tag{3.27}$$

which is of the exact same form as the corresponding constrained Metropolis accept probability defined in Equation (3.20). The constrained stretch move is implemented, evaluated and further discussed in Section 4.4.

3.3.4 Ellipsoidal nested sampling

A rather different (non-MCMC) approach to likelihood-constrained prior sampling is ellipsoidal sampling (Mukherjee et al. [12]). From the covariance matrix of the current set of active points it is possible to define an n-dimensional ellipsoid intended to approximate the

iso-likelihood surface defined by $\mathcal{L}(\boldsymbol{\theta}) = L^*$. Ellipsoidal sampling proceeds by enlarging the ellipsoid by some factor, compensating for the iso-likelihood surface not being perfectly ellipsoidal, then drawing a sample from the prior volume within this bound. The more closely the ellipsoidal approximates the iso-likelihood surface, the higher is the probability that the sample fulfills the likelihood constraint. In a hypothetical situation where the iso-likelihood is perfectly ellipsoidal, the acceptance ratio is one. Shaw et al. [40] improves the method for multi-modal distributions and gives a method for uniform sampling of an ellipsoid (extension to non-uniform priors is straightforward). Ellipsoidal sampling has proven quite successful and is used in state-of-the-art implementations such as `MultiNest` by Feroz et al. [10]. The benefits of ellipsoidal sampling are that the samples produced are independent and that this can be achieved with typically very few likelihood-evaluations. This is in contrast to MCMC methods which need several steps for samples to become (approximately) independent, where each of those steps requires the likelihood to be evaluated. On the other hand, the iso-likelihood is not always well approximated by the ellipsoid in which case the method will fail without any warning signal. No implementation of ellipsoidal sampling has been carried out in this work but it will act as a benchmark for the MCMC methods outlined above through the use of `MultiNest` via the Python code `PyMultiNest` [24].

4

Constrained prior sampling implementations

Three different MCMC schemes for generating constrained prior samples as required by the nested sampling algorithm were implemented in this work and are described in this chapter. The implementations are based on the methods introduced in Section 3.3 and will be referred to as constrained Metropolis, GMC and constrained stretch move. These three MCMC methods have all been integrated into a nested sampling framework that was developed in this work and that follows the principles presented in Chapter 3 in general and Algorithm 3.1 in particular. For our purposes we have mainly used $N = 1000$ active points and imposed a stopping criterion according to Equation (3.17) using $f_{\ln} = 0.01$.

Here follows a description of the specific implementation designs and usage, including e.g. method-specific hyperparameters. Each method has two main input parameters: the number of exploration steps per iteration and a step size scale. Demonstration and testing will be carried out by applying each method to the toy problem introduced in Section 2.1.1.

4.1 Choice of coordinates: the unit hypercube

A nested sampling convention is to work with parameter coordinates $\mathbf{u} = (u_0, ..., u_{n-1})$ in which the prior is uniform over a cube of unit volume in n-dimensional space. This convention is also adopted in this work and has already been mentioned in the context of GMC in Section 3.3.2. If the joint prior probability is independent, which means it can be written as a product of individual parameter priors[1] $\pi(\boldsymbol{\theta}) = \pi_0(\theta_0)...\pi_{n-1}(\theta_{n-1})$, the transformation relating $\boldsymbol{\theta}$ to $\mathbf{u}$ is

$$\theta_i(u_i) = \Pi_i^{-1}(u_i) \tag{4.1}$$

where $\Pi_i(\theta_i) = \int_{-\infty}^{\theta_i} \pi_i(\theta_i')\mathrm{d}\theta_i'$ is the cumulative distribution function (cdf) for θ_i. The MCMC implementations described below are formulated in terms of the coordinates $\mathbf{u}$ but the obtained results are however transformed back to $\boldsymbol{\theta}$ using Equation (4.1). A consequence of working in coordinates where the prior is flat is that any Metropolis style rejection step simplifies. The acceptance probability α (see e.g. Equations (3.20) and (3.27)) contains a prior ratio $\pi(\boldsymbol{\theta}')/\pi(\boldsymbol{\theta})$ which cancels if the prior is constant.

4.2 Constrained Metropolis implementation

The key object in the Metropolis scheme is the proposal distribution $Q(\mathbf{u}'|\mathbf{u})$ from which new positions $\mathbf{u}'$ are proposed in the random walk, depending on the current position

[1] This is the case for the prior defined in Equation (2.19).

$\mathbf{u}$. The common approach of taking Q to be a symmetric Gaussian centered at $\mathbf{u}$ with a fixed covariance matrix $\Sigma_Q = \sigma_Q^2 \mathbb{1}_n$ (see Section 2.2.1) would be quite insufficient for the requirements concerning nested sampling. Fixing the scale σ_Q of the proposal distribution would not be compatible with the exponentially shrinking domain of nested sampling. We therefore propose that information from the active set of points $\mathbf{u}_1^{\mathrm{a}}, ..., \mathbf{u}_N^{\mathrm{a}}$ be utilized to estimate the scale of the active region. Firstly, we randomly select a starting point $\mathbf{u}_{\mathrm{start}}$ from the active set. Secondly, a subset $\{\mathbf{u}_m\}_{m=1}^M$ of the remaining active points is created, also at random. The size of the subset, M, is in this work always taken to be $M = \max\left(1, \left\lfloor \frac{N}{10} \right\rfloor\right)$. Thirdly, the covariance matrix elements of the proposal distribution are set according to

$$(\Sigma_Q)_{ij} = \begin{cases} \frac{s^2}{M} \sum_{m=1}^{M} (\mathbf{u}_{\mathrm{start}} - \mathbf{u}_m)_i^2, & \text{if } i = j \\ 0, & \text{otherwise,} \end{cases} \tag{4.2}$$

where $(\mathbf{u}_{\mathrm{start}} - \mathbf{u}_m)_i^2$ denotes the squared distance along dimension i and s is an overall scale parameter. With this formula, Σ_Q is a diagonal matrix with variances on the diagonal proportional to estimates of the typical squared distances for each dimension in the active set of points and by extension in the active region. A typical value of the scale parameter is $s \sim 0.1$ in order for proposed steps to be well within the allowed region. Effects of different choices of s are studied in Section 4.6.1. To introduce additional randomness, the number of steps taken in every walk is drawn from a discrete uniform distribution $U\left(\frac{1}{2}\langle N_{\mathrm{s}}\rangle, \frac{3}{2}\langle N_{\mathrm{s}}\rangle\right)$ where the average $\langle N_{\mathrm{s}}\rangle$ is set by the user. The proposed constrained Metropolis algorithm is shown in Algorithm 4.1.

Algorithm 4.1: Constrained Metropolis procedure.

Input: Likelihood limit L^* and active points $\mathbf{u}_1^{\mathrm{a}}, ..., \mathbf{u}_N^{\mathrm{a}}$
Output: New active point $\mathbf{u}_{\mathrm{new}}$

Set $\mathbf{u}^{(0)} \leftarrow \mathbf{u}_{\mathrm{start}}$ and Σ_Q from $\mathbf{u}_1^{\mathrm{a}}, ..., \mathbf{u}_N^{\mathrm{a}}$, M and s
$N_{\mathrm{s}} \leftarrow$ sample from $U\left(\frac{1}{2}\langle N_{\mathrm{s}}\rangle, \frac{3}{2}\langle N_{\mathrm{s}}\rangle\right)$
$t \leftarrow 0$
while $t < N_{\mathrm{s}}$ **or** acceptance is zero **do**
  $\mathbf{u}' \leftarrow$ sample from $Q(\mathbf{u}'|\mathbf{u}^{(t)}) = \mathcal{N}(\mathbf{u}'; \mathbf{u}^{(t)}, \Sigma_Q)$
  if $\mathcal{L}(\boldsymbol{\theta}(\mathbf{u}')) \geq L^*$ **then**
    $\mathbf{u}^{(t+1)} \leftarrow \mathbf{u}'$
  else
    $\mathbf{u}^{(t+1)} \leftarrow \mathbf{u}^{(t)}$
  $t \leftarrow t + 1$
$\mathbf{u}_{\mathrm{new}} \leftarrow$ last position from the chain

An example of a constrained Metropolis walk can be seen in Figure 4.1 and is for the sake of visualization conducted in two-dimensions. The walk starts at the green plus-shaped marker and ends at the position of the red cross which is promoted to member of the active points. The hyperparameters settings used in this example are $\langle N_{\mathrm{s}}\rangle = 20$ and $s = 0.5$. The number of unique steps in the walk in Figure 4.1 are however fewer than $\frac{1}{2}\langle N_{\mathrm{s}}\rangle = 10$, meaning that not all steps have been accepted. The acceptance rate r_{a}, defined as the fraction of accepted steps in an MCMC walk or trajectory, is an important measure for assessment of the exploration performance and will be discussed later in this thesis (e.g. Section 4.6.1).

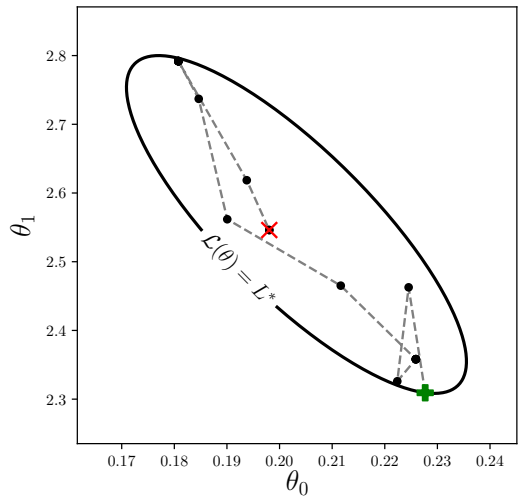


Figure 4.1: Example of a Metropolis random walk in a likelihood-constrained region of a two-dimensional parameter space. The walk is captured at a nested sampling iteration with $\xi \approx e^{-\mathcal{H}}$. The green plus and the red cross indicates the start of the walk and the end of the walk, respectively. The black dots indicate intermediate steps. The average number of steps for this walk is $\langle N_\mathrm{s} \rangle = 20$ and the scale parameter is $s = 0.5$.

4.3 Galilean Monte Carlo implementation

The GMC equivalent of the Metropolis proposal distribution Q discussed in the previous Section 4.2 is the velocity distribution $q(\mathbf{v})$ from which the initial velocity is sampled at the beginning of each iteration[2]. By the same arguments as in the Metropolis case above, this distribution needs to adapt to the shrinking active volume. We will therefore similarly suggest that $q(\mathbf{v})$ be a Gaussian $\mathcal{N}(\mathbf{v}; \mathbf{0}, \Sigma_\mathrm{vel})$ where the covariance matrix elements at each iteration are set according to

$$(\Sigma_\mathrm{vel})_{ij} = \begin{cases} \frac{1}{M} \sum_{m=1}^{M} \frac{1}{\tau_\mathrm{ref}^2} (\mathbf{u}_\mathrm{start} - \mathbf{u}_m)_i^2, & \text{if } i = j \\ 0, & \text{otherwise,} \end{cases} \tag{4.3}$$

where $\mathbf{u}_\mathrm{start}$ and $\mathbf{u}_m$ are the same as before and $\tau_\mathrm{ref} = 1$ is a reference time scale which gives the covariances units of velocity squared. The lack of a variable overall length scale parameter in Equation (4.3) compared to the corresponding s in Equation (4.2) is due to the time step formulation "$\mathbf{u} + \tau\mathbf{v}$"of GMC, where τ is used to set the overall step size. We will in Section 4.6.1 see the impact from specific choices of τ. Furthermore, the number of steps taken in the trajectory are set randomly in the same way as in Section 4.2. Consequently, the average number of steps $\langle N_\mathrm{s} \rangle$ together with the scale parameter τ constitutes the main user input for GMC. Following this scheme and the principles laid out in Section 3.3.2, the full implemented GMC procedure is shown in Algorithm 4.2. The four possible outcomes after a GMC step described in Section 3.3.2 are here denoted North (proceed), East (reflect), West (mirrored reflection) and South (reverse). It is important to stress that it is the gradient with respect to $\mathbf{u}$ of the likelihood that should be computed to obtain the normal to the reflection surface. This is related to the gradient with

[2] Analogous to the HMC case discussed in Section 2.2.2.

respect to $\boldsymbol{\theta}$ by the chain rule via the Jacobian matrix J of the transformation in Equation (4.1) according to $\nabla_{\mathbf{u}} \to J(\mathbf{u})\nabla_{\boldsymbol{\theta}}$. The Jacobian matrix element definition used here is $J_{ij} = \partial\theta_j/\partial u_i$.

Algorithm 4.2: Galilean Monte Carlo procedure.

Input: Likelihood limit L^* and active points $\mathbf{u}^{\mathrm{a}}_1, ..., \mathbf{u}^{\mathrm{a}}_N$
Output: New active point $\mathbf{u}_{\mathrm{new}}$

Set $\mathbf{u}^{(0)} \leftarrow \mathbf{u}_{\mathrm{start}}$ and Σ_{vel} from $\mathbf{u}^{\mathrm{a}}_1, ..., \mathbf{u}^{\mathrm{a}}_N$ and M
$\mathbf{v} \leftarrow$ sample from $q(\mathbf{v}) = \mathcal{N}(\mathbf{v}; \mathbf{0}, \Sigma_{\mathrm{vel}})$
$N_{\mathrm{s}} \leftarrow$ sample from $U\left(\frac{1}{2}\langle N_{\mathrm{s}}\rangle, \frac{3}{2}\langle N_{\mathrm{s}}\rangle\right)$
for $t = 0, 1, ..., N_{\mathrm{s}} - 1$ **do**
  $\mathbf{u}' \leftarrow \mathbf{u}^{(t)} + \tau\mathbf{v}$
  N $\leftarrow \mathcal{L}(\boldsymbol{\theta}(\mathbf{u}')) \geq L^*$ # Continue north
  if N **then**
    $\mathbf{u}^{(t+1)} \leftarrow \mathbf{u}'$ # Go north
  else
    $\mathbf{n} \leftarrow \frac{\nabla_{\mathbf{u}} \ln \mathcal{L}(\boldsymbol{\theta}(\mathbf{u}'))}{|\nabla_{\mathbf{u}} \ln \mathcal{L}(\boldsymbol{\theta}(\mathbf{u}'))|}$
    $\mathbf{v}' \leftarrow \mathbf{v} - 2\mathbf{n}\mathbf{n}^{\mathrm{T}}\mathbf{v}$
    # Check possible directions
    E $\leftarrow \mathcal{L}(\boldsymbol{\theta}(\mathbf{u}' + \tau\mathbf{v}')) \geq L^*$ # Reflection to east
    W $\leftarrow \mathcal{L}(\boldsymbol{\theta}(\mathbf{u}' - \tau\mathbf{v}')) \geq L^*$ # Mirrored reflection to west
    S $\leftarrow \mathcal{L}(\boldsymbol{\theta}(\mathbf{u}' - \tau\mathbf{v})) \geq L^*$ # Reversal to south
    if S **and** (E but **not** W) **then**
      $\mathbf{u}^{(t+1)} \leftarrow \mathbf{u}' + \tau\mathbf{v}'$ # Go east
      $\mathbf{v} \leftarrow \mathbf{v}'$
    else if S **and** (W but **not** E) **then**
      $\mathbf{u}^{(t+1)} \leftarrow \mathbf{u}' - \tau\mathbf{v}'$ # Go west
      $\mathbf{v} \leftarrow -\mathbf{v}'$
    else
      $\mathbf{u}^{(t+1)} \leftarrow \mathbf{u}^{(t)}$ # Aim south
      $\mathbf{v} \leftarrow -\mathbf{v}$
$\mathbf{u}_{\mathrm{new}} \leftarrow$ last point of the trajectory $\mathbf{u}^{(N_{\mathrm{s}})}$

Figure 4.2 shows an example of a GMC trajectory in $n = 2$ dimensions contained by a iso-likelihood contour. The trajectory is initiated at the the green plus-shaped marker, reflects twice off the walls and ends at the red cross. The reflections in the figure do not appear physically correct given the directions of the iso-likelihood surface. However, this is a combined effect of the different scales on the two axes and that the trajectory is presented in the original coordinates $\boldsymbol{\theta}$ whereas the simulation is performed in the flat coordinates $\mathbf{u}$. The parameters used for this example are $\langle N_{\mathrm{s}}\rangle = 20$ and $\tau = 0.1$.

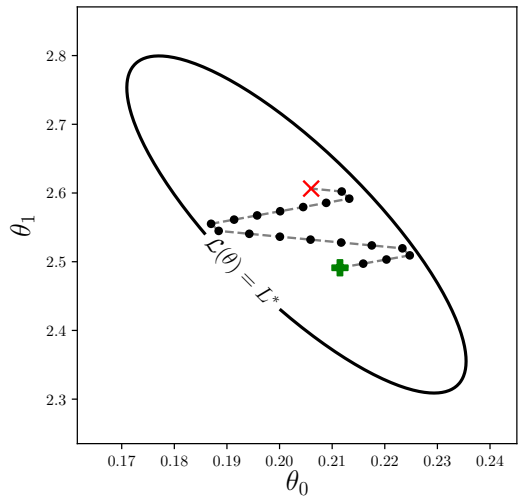


Figure 4.2: Example of a GMC trajectory in a likelihood-constrained region of a two-dimensional parameter space. The trajectory is captured at a nested sampling iteration with $\xi \approx e^{-\mathcal{H}}$. The green plus and the red cross indicates the start of the trajectory and the end of the walk, respectively. The black dots indicate intermediate time steps. The reflections do not occur exactly at the boundary but rather at proxy surfaces just outside the boundary (not in figure) as described in Section 3.3.2.

4.4 Constrained stretch move implementation

One major advantage of the combination of the stretch move and nested sampling, proposed in this thesis, is that it requires no explicit adjustment when the nested sampling process progresses and the active volume shrinks. The stretch move proposal distribution naturally adapts to the decreasing scales of parameter space and there is no need to introduce an approach similar to the covariance matrix updates defined in Equations (4.2) and (4.3). The number of stretch move steps is randomly drawn as above and the average number of steps $\langle N_\mathrm{s} \rangle$ is set by the user. Except from the number of steps, the user additionally needs to specify the scale parameter a that sets the extent of the distribution $g(\zeta)$ defined in Equation (2.28). In Section 4.6.1 we will study the sampling performance based on the choice of a. The full stretch move procedure is shown in Algorithm 4.3.

A constrained stretch move walk in $n = 2$ dimensions is shown in Figure 4.3 with hyperparameters $\langle N_\mathrm{s} \rangle = 20$ and $a = 2$. It is clear that the step sizes seem to fluctuate more compared to the Metropolis walk in Figure 4.1. This could potentially be an effect of the fact that the Metropolis procedure uses the same proposal distribution in every step whereas the stretch move proposal distribution changes in every step depending on the randomly selected point $\mathbf{u}_j$. This is, however, the results of only a single iteration from a single run and one should be careful when drawing conclusions from it.

Algorithm 4.3: Constrained stretch move procedure.

Input: Likelihood limit L^* and active points $\mathbf{u}_1^{\mathrm{a}}, ..., \mathbf{u}_N^{\mathrm{a}}$
Output: New active point $\mathbf{u}_{\mathrm{new}}$

Set $\mathbf{u}^{(0)} \leftarrow \mathbf{u}_{\mathrm{start}}$ from $\mathbf{u}_1^{\mathrm{a}}, ..., \mathbf{u}_N^{\mathrm{a}}$
$N_{\mathrm{s}} \leftarrow$ sample from $U\left(\frac{1}{2}\langle N_{\mathrm{s}}\rangle, \frac{3}{2}\langle N_{\mathrm{s}}\rangle\right)$
$t \leftarrow 0$
while $t < N_{\mathrm{s}}$ **or** acceptance is zero **do**
  Draw $\mathbf{u}_j$ randomly from the complementary set $\{\mathbf{u}_j^{\mathrm{a}}\}_{j=1}^{N} \setminus \{\mathbf{u}_{\mathrm{start}}\}$
  $\tilde{\zeta} \leftarrow$ random sample from $g(\zeta)$, Equation (2.28)
  $\mathbf{u}' \leftarrow \mathbf{u}_j + \tilde{\zeta}(\mathbf{u}^{(t)} - \mathbf{u}_j)$
  $\alpha \leftarrow \tilde{\zeta}^{n-1}$, Equation (3.27)
  $r \leftarrow$ random sample from $U(0, 1)$
  if $r \leq \alpha$ **and** $\mathcal{L}(\boldsymbol{\theta}(\mathbf{u}')) \geq L^*$ **then**
    $\mathbf{u}^{(t+1)} \leftarrow \mathbf{u}'$
  else
    $\mathbf{u}^{(t+1)} \leftarrow \mathbf{u}^{(t)}$
  $t \leftarrow t + 1$
$\mathbf{u}_{\mathrm{new}} \leftarrow$ last position from the chain

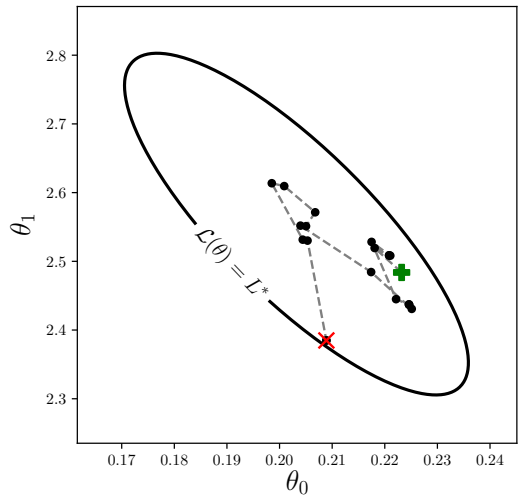


Figure 4.3: Example of a stretch move random walk in a likelihood-constrained region of a two-dimensional parameter space. The walk is captured at a nested sampling iteration with $\xi \approx e^{-\mathcal{H}}$. The green plus and the red cross indicates the start of the walk and the end of the walk, respectively. The black dots indicate intermediate steps.

4.5 Monitoring the sampling progress

The progress of the sampling over the course of a nested sampling run can be examined by tracking a few key quantities. Examples of such quantities are the likelihood L defined in Equation (3.4), the posterior weights $w = L\Delta\xi/Z$ defined in Equation (3.14) and the acceptance rate r_{a} defined as

$$r_{\mathrm{a}} = \frac{\#\text{ steps accepted}}{\#\text{ steps proposed}} \tag{4.4}$$

in each iteration. The tracking of these quantities is shown in Figure 4.4 for the three methods applied to the problem of Sections 2.1.1 and 2.1.4 in $n = 3$ dimensions using $\langle N_{\mathrm{s}} \rangle = 40$. The natural choice of progress parameter is $-\ln \xi$ which is directly proportional to the iteration number k according to Equation (3.12). We see clearly that it takes a considerable fraction of the total amount of iterations before reaching significant likelihood values (top panel). The peak of the weights (middle panel) indicates where the bulk of the posterior probability mass is located and it is obvious that this does not coincide with the point of maximum likelihood. The reason for this is that as the likelihood increases, the prior mass elements, $\Delta\xi \propto e^{-k/N}$, decrease exponentially. This tug of war implies the existence of an optimal point where w is maximized. The vertical lines (middle panel) accompanied by bands (see the inset) show the average and uncertainty $N\mathcal{H} \pm \sqrt{N\mathcal{H}}$ of the number of steps that is needed to reach the bulk, as was discussed in Section 3.2.5, and where the information $\mathcal{H}$ is computed according to Equation (3.16). These estimates seem to predict reasonably accurately where the weights have reached significant values, i.e. where the bulk of the posterior mass is. The most obvious difference between the methods is, however, the evolution of the acceptance rate r_{a} (bottom panel). The acceptance rate values typically vary rapidly between individual iterations, making it necessary to a apply a moving median[3] to read out the overall behavior. This has been done to obtain the acceptance rate curves in Figure 4.4 and the associated band edges show the 16^{th} and 84^{th} percentiles, respectively. We see from these percentiles that the typical variation between iterations is reasonably large for all three methods, especially for GMC in the second half of the run where we also observe a prominent asymmetry. Furthermore, it is natural for the GMC acceptance rate to be larger compared to the other methods as the GMC step should be smaller in order to create a trajectory rather than a random walk. The most drastic change is seen for the Metropolis acceptance rate as its median drops to approximately a third of its initial value. The drop ends roughly in the region where the posterior mass starts to accumulate which means this is where the algorithm struggles the most with finding acceptable proposal steps. Remarkably, the stretch move median acceptance rate is fairly unchanged over the course of the run, which suggests that the proposal distribution adapts well to the successively stricter likelihood constraint.

The results of the nested sampling runs described above can be seen in Table 4.1. The computed evidence values $\ln Z$ differ between the methods but do however agree fairly well with the true value $\ln Z_{\mathrm{true}} = 8.09$ obtained in Section 2.1.4 for the model with $n = 3$. The uncertainty estimates $\sqrt{\mathcal{H}/N}$ (see Section 3.2.5) are, to two significant figures, equal for the three methods. This is a general observation that has been made throughout this work and that we will return to in Section 5.2.1. The number of likelihood evaluations, $N_{\mathcal{L}}$, are seen to be almost twice as many for GMC compared to the other methods despite that the number of steps are the same. The first reason for this is that the GMC algorithm requires evaluations of the likelihood gradient, in the event of a reflection, which is here counted as an extra likelihood evaluation and included in $N_{\mathcal{L}}$. The fraction of gradient evaluations are given in parenthesis in Table 4.1. The second reason is that more than one likelihood evaluation occur in the event of a reflection in order to assess the reflection conditions according to Algorithm 4.2.

[3]The median-equivalent to a moving (or rolling) average, which is better suited for constrained and skewed data [41] as in this case for r_{a}.

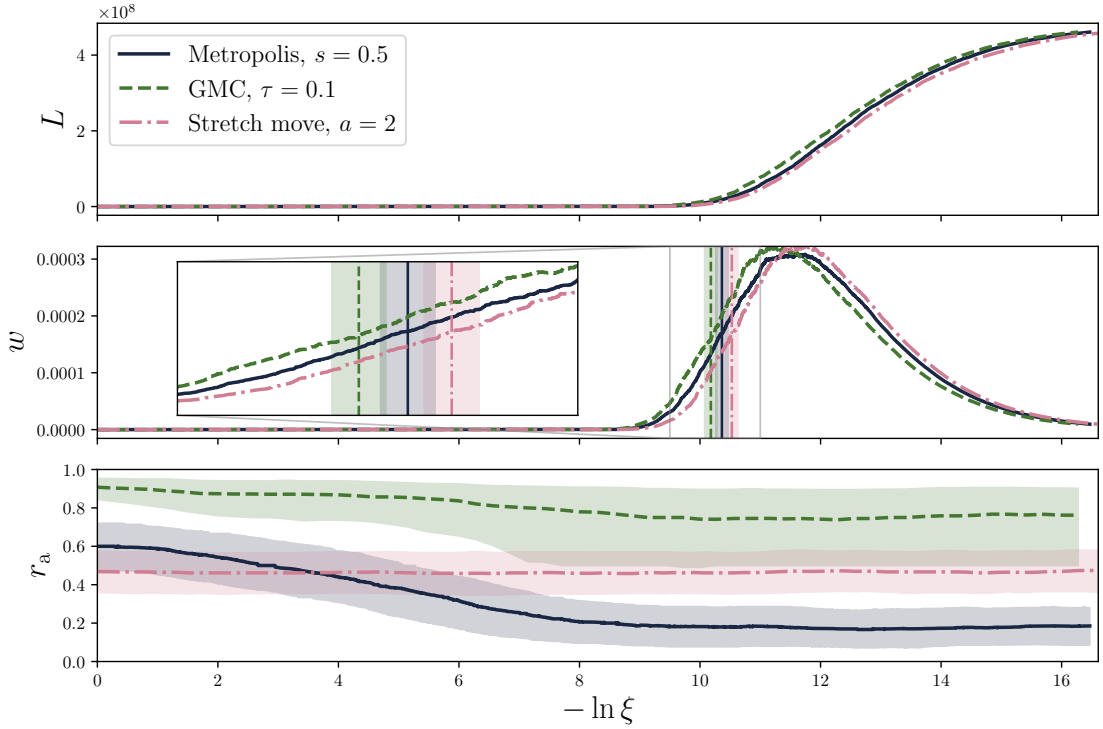


Figure 4.4: Progress of three key quantities over the course of a nested sampling run for the three different methods. The sampling is performed in $n = 3$ dimensions using $\langle N_\mathrm{s} \rangle = 40$ with sampling parameters $N = 1000$ and $f_\mathrm{ln} = 0.01$. The top panel shows the likelihood L for the worst point of each iteration, the middle panel shows the posterior weights w, used to compute the evidence, and the bottom panel shows the moving median of the acceptance rate r_a for each iteration. The methods distinguish themselves clearly in the variation of the acceptance rate.

Table 4.1: Nested sampling results for the three methods applied to the same problem in $n = 3$ dimensions. Sampling parameters are $N = 1000$ active points and $f_\mathrm{ln} = 0.01$ tolerance. The true evidence value for this model is $\ln Z_\mathrm{true} = 8.09$ as was analytically obtained in Figure 2.3. The number of likelihood evaluations, $N_\mathcal{L}$, does in the GMC case, include the number of evaluations of the gradient. The fraction of gradient evaluations is given in parenthesis.

Method	$\langle N_\mathrm{s} \rangle$	Scale parameter	$\ln Z \pm \sqrt{\frac{\mathcal{H}}{N}}$	$N_\mathcal{L}$	# Iterations
Metropolis	40	$s = 0.5$	8.13 ± 0.10	$\sim 6.65 \cdot 10^5$	$\sim 1.65 \cdot 10^4$
GMC	40	$\tau = 0.1$	8.32 ± 0.10	$\sim 1.12 \cdot 10^6$ (13%)	$\sim 1.63 \cdot 10^4$
Stretch move	40	$a = 2.0$	7.99 ± 0.10	$\sim 6.68 \cdot 10^5$	$\sim 1.66 \cdot 10^4$

4.6 Hyperparameters

The three constrained MCMC implementations described in the previous sections have in common that they have two main hyperparameters:

- a scale parameter effectively setting the overall step size, denoted s, τ and a, respectively, in the three approaches
- the average number of steps $\langle N_\mathrm{s} \rangle$.

These hyperparameters need to be set by the user and it is therefore important to have an idea of how they influence the sampling performance. We will therefore proceed by studying the effects of different specific hyperparameter choices.

4.6.1 Sensitivity to the scale parameter value

The scale parameters s, τ and a act as specifiers for the overall step size for the three MCMC exploration methods. They are, however, defined in different ways and will therefore not take on the same values. For instance, the stretch move parameter a defines an interval $[a^{-1}, a]$ from which $\tilde{\zeta}$ is drawn (see Equation (2.28)), meaning that we must have $a > 1$ for this formulation to be reasonable. The Metropolis parameter s, on the other hand, measures the step size relative to the current active volume and should therefore be $\sim$ 0.1–1.0 in order for a step not to be too large to step outside too often, but not too small to make the walk not cover enough distance. The time step τ of the GMC algorithm should typically be smaller than s such that the proxy surface approximation is acceptable. For the idea of a trajectory to be valid, at least several GMC steps should be taken in a straight line before a reflection occurs and we should therefore have $\tau \sim 0.1$.

To measure the impact of the scale parameters we apply the methods to the problem presented in Sections 2.1.1 and 2.1.4 using different scale parameter values. The top row of Figure 4.5 shows the error made when computing the (log) evidence, $\ln Z$, for a model of the form (2.8) with $n = 3$ for different choices of s, τ and a. It is important to stress that this type of error is only possible to obtain when the true value Z_true is analytically obtainable as was done in Section 2.1.4 and showed in Figure 2.3. The bands are standard deviations between five identical runs using different random seeds. The sampling was performed using $N = 1000$ active points, $f_\mathrm{ln} = 0.01$ tolerance and $\langle N_\mathrm{s} \rangle = 40$ exploration steps per iteration (on average). Note the logarithmic axes for s and τ. The Metropolis method (left column) slightly overestimates the evidence for the entire range in s, but there appears to be a minimum in the error around $s \approx 0.25$–0.5. This is in agreement with the argument above that s should not be so large that too many proposed steps are outside the allowed region but not so small that not enough distance is covered. The GMC error (middle column) is seen to be roughly constant for $\tau \approx 0.05$–0.1 before it drastically diverges at $\tau \approx 0.2$. This is probably a consequence of a too low acceptance rate causing the trajectories to insufficiently explore the available parts of parameter space. The stretch move error (right column) is in this case seen to be the closest to zero among the three methods and is also quite stable to variations of a around the value $a = 2.0$ proposed by [31] for the unconstrained case. Note, however, that the scale for a is linear as opposed to s and τ.

To quantify how efficiently each method explores parameter space we need to compute the acceptance rate r_a for each nested sampling iteration, defined in Equation (4.4). The value

of the acceptance rate will typically vary significantly between individual nested sampling iterations but will on a larger scale show clear trends, as seen in the bottom panel of Figure 4.4. As the nested sampling compression approaches the bulk of the posterior mass, the active region will have shrunken enough to make r_a tend to lower values. As argued in Sections 3.2.4 and 3.2.5 and showed in the middle panel of Figure 4.4, the bulk is approximately reached when $-\ln \xi \approx \mathcal{H}$, which roughly should be where the value of r_a is of most importance. This motivates us to introduce the *bulk median* of the acceptance rate, $\mathcal{M}_\mathcal{H}[r_\mathrm{a}]$, defined as the median of the recorded values $r_{\mathrm{a},k}$, over all nested sampling iterations k, for which $-\ln \xi_k \geq \mathcal{H}$. Formally:

$$\mathcal{M}_\mathcal{H}[r_\mathrm{a}] = \mathrm{Median}[\{r_{\mathrm{a},k},\ \forall k : -\ln \xi_k \geq \mathcal{H}\}] \tag{4.5}$$

where the information $\mathcal{H}$ is computed according to Equation (3.16). The bulk median $\mathcal{M}_\mathcal{H}[...]$ should be interpreted as a point estimate of a quantity (the acceptance rate in this case) in the region where it matters the most, i.e. where the bulk of the probability mass is located. $\mathcal{M}_\mathcal{H}[r_\mathrm{a}]$-values for the three methods are shown for different values of the scale parameters in the middle row of Figure 4.5. The bands (barely visible) are the same as for the evidence errors in the top row. There is a clear decrease in $\mathcal{M}_\mathcal{H}[r_\mathrm{a}]$ for each of the three methods as their scale parameters are increased. This is to be expected as a larger step size implies a higher probability for a proposed step to be outside the active volume. The most drastic change is for the Metropolis case (left column) which drops to near zero already at $s \approx 1.0$. Moreover, the acceptance rate should not be too large, as indicated by the evidence error in the top row which is minimized at $s \approx 0.25$–0.5, corresponding to a bulk median acceptance rate of $\mathcal{M}_\mathcal{H}[r_\mathrm{a}] \approx 0.2$–$0.4$. The stretch move implementation (right column) also shows a significant drop in acceptance rate as a increases. However the drop is more moderate compared to Metropolis. Comparing this result to the top row of Figure 4.5 we see that $\mathcal{M}_\mathcal{H}[r_\mathrm{a}] \approx 0.4$–$0.6$ for $a \approx 1.5$–3.0 seems to be a suitable range as this is where the evidence computed by the stretch move implementation fluctuates the least. For the GMC implementation (middle column) it is quite noteworthy how the character of $\mathcal{M}_\mathcal{H}[r_\mathrm{a}]$'s dependence on τ differs from the other cases. A plausible explanation for this fundamentally different behavior is that a single GMC trajectory is, as opposed to Metropolis and the stretch move, not a random walk and does not form a Markov chain. In this sense, the GMC approach is fundamentally different. Once the GMC velocity is set, the step size will not change during the course of the trajectory and the direction only changes in the event of a reflection. In contrast, the Metropolis as well as the stretch move procedures draw direction as well as size randomly for each step, causing an important distinction.

The bottom row of Figure 4.5 shows how the total number of likelihood evaluations, $N_\mathcal{L}$, varies with the scale parameters. Note that for the random walk methods, Metropolis and stretch move, $N_\mathcal{L}$ will only exceed $N_\mathrm{s} \times (\#\ \text{iterations})$ if there are iterations where none of the first N_s proposed steps are rejected. This is a consequence of the while loop-conditions in Algorithms 4.1 and 4.3, which assure that at least one step is accepted. This is the reason behind the increase in $N_\mathcal{L}$ for the Metropolis case (left column) when s is increased. Furthermore, there seems to be a sudden jump between $s = 1.0$ and $s = 2.0$, indicating the existence of a threshold in s above which almost no steps are accepted. This agrees well with the observation of the Metropolis acceptance rate (middle row, left column), which for $s = 2.0$ is close to zero. In contrast, the stretch move number of likelihood evaluations (right column) is seen to be roughly constant for the displayed range in a, indicating that there is at least one accepted step among the first N_s proposals in the

majority of the iterations. In other words, the corresponding threshold for a has not been surpassed. For the GMC approach (middle column), $N_{\mathcal{L}}$ (including the number of gradient evaluations) appears to vary oppositely to $\mathcal{M}_{\mathcal{H}}[r_{\mathrm{a}}]$, which is to be expected since the number of likelihood evaluations per iteration should be approximately

$$\frac{N_{\mathcal{L}}}{\#\text{ iterations}} \approx \underbrace{\langle N_{\mathrm{s}} \rangle}_{\#\mathcal{L}\text{-evaluations}} + 3\underbrace{(1 - \mathcal{M}_{\mathcal{H}}[r_{\mathrm{a}}])\langle N_{\mathrm{s}} \rangle}_{\#\nabla\mathcal{L}\text{-evaluations}} \propto \text{const.} - \mathcal{M}_{\mathcal{H}}[r_{\mathrm{a}}]. \tag{4.6}$$

The factor of 3 accounts for the likelihood evaluations required to assess the reflection conditions as is described in Section 3.3.2 and Algorithm 4.2. It should be stressed, however, that the gradient in general can be more expensive to evaluate than the likelihood itself, effectively leading to an additional increase in the total computation time.

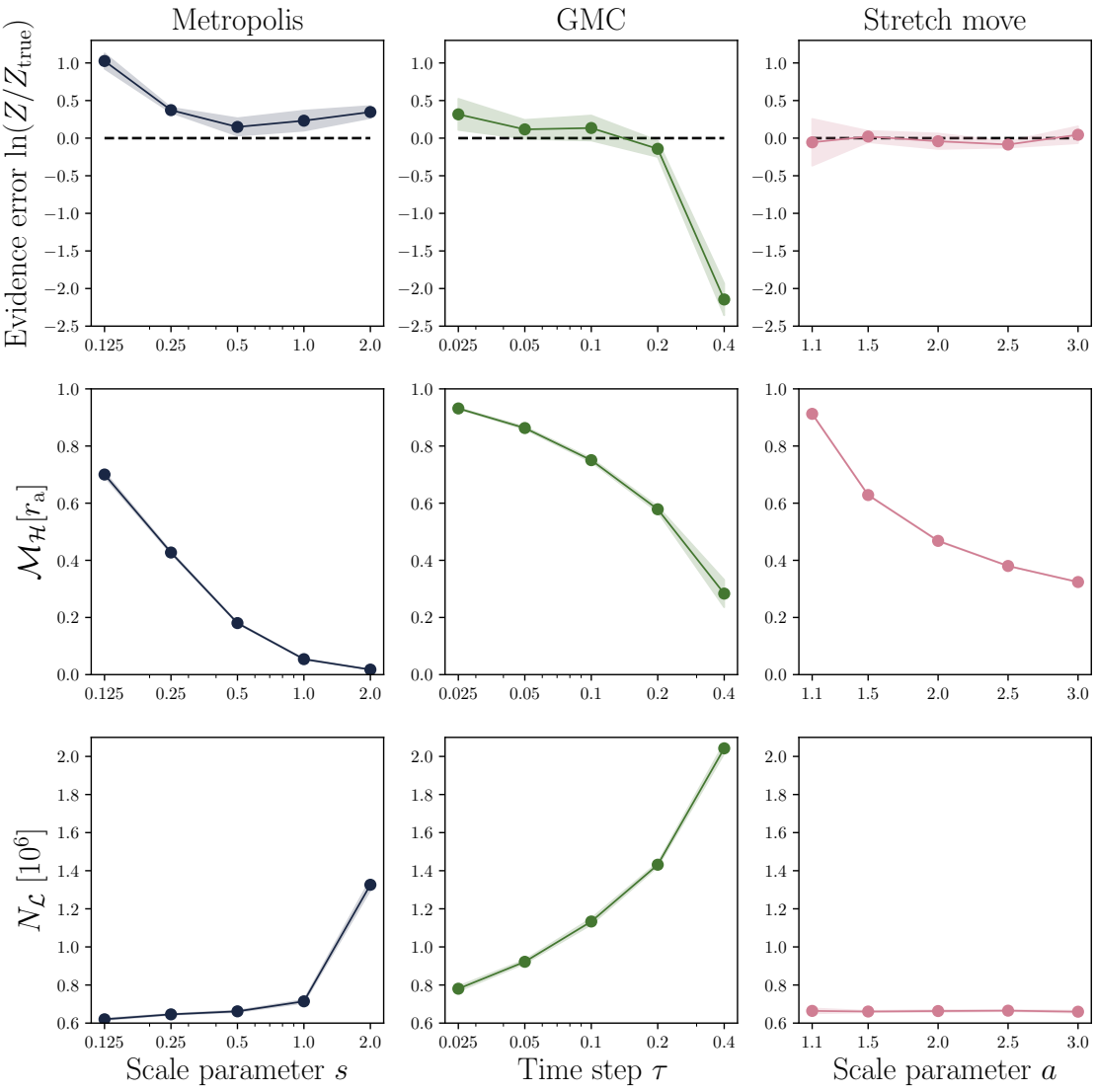


Figure 4.5: The (log) evidence error (top row), the bulk median acceptance rate (middle row) and the number of likelihood evaluations (bottom row) in $n = 3$ dimensions for the three methods applied in the nested sampling framework to the toy problem in Section 2.1.1 for different values of the scale parameters s, τ and a. Sampling parameters were $N = 1000$ active points, $f_{\ln} = 0.01$ tolerance and $\langle N_{\mathrm{s}} \rangle = 40$ exploration steps per iteration.

4.6.2 Choosing a sufficient number of steps

We observed in Section 4.6.1 that the value of the scale parameter greatly influences the sampling behavior, especially for the Metropolis and GMC implementations. We will now turn to the second common hyperparameter which is the average number of steps per nested sampling iteration $\langle N_\mathrm{s} \rangle$. The chosen number of steps is closely related to the requirement that the new point generated in every iteration ought to be (approximately) independent of the starting point. The correlation between the start and the endpoint decreases with the number of steps, making it desirable not to set $\langle N_\mathrm{s} \rangle$ too low. There is, however, a competing aspect requiring that $\langle N_\mathrm{s} \rangle$ is not too high, namely that the number of likelihood evaluations, $N_\mathcal{L}$, increases (linearly) with the number of steps making the sampling more expensive. One should therefore strive to use the minimum number of steps possible while maintaining sufficient performance.

Figure 4.6 shows the error in the computed evidence using the three methods as a function of the average number of steps per iteration $\langle N_\mathrm{s} \rangle$. Sampling is performed for the same problem as previously in $n = 3$ dimensions with $N = 1000$ active points and $f_\mathrm{ln} = 0.01$ tolerance and scale parameters are as indicated in the figure. The bands are, as before, the standard deviations of five identical runs with different random seeds. The general trend is that the error approaches zero and fluctuates less for larger $\langle N_\mathrm{s} \rangle$, in agreement with the argument above. Again we see that the stretch move is more stable towards changes in hyperparameters whereas the performance of Metropolis and GMC depend more strongly on the choice of $\langle N_\mathrm{s} \rangle$. In this particular example, this means that Metropolis and GMC require 40–80 steps, or even more, to achieve results comparable in accuracy to those that the stretch move produces with only 10–20 steps.

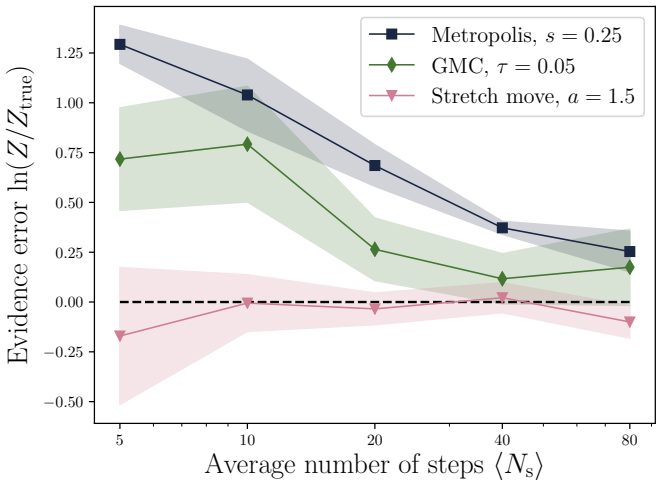


Figure 4.6: Error in the computed (log) evidence with different choices of the average number of steps in each nested sampling iteration. Sampling parameters are $N = 1000$ active points and $f_\mathrm{ln} = 0.01$ and the scale parameters are as indicated by the legend. The bands are the standard deviations of five identical runs with different random seeds. The general trend is that the error approaches zero and fluctuates less for larger $\langle N_\mathrm{s} \rangle$.

Throughout this chapter we have been testing the nested sampling implementations in fairly modestly sized spaces, mainly $n = 3$ dimensions. In the next chapter we will increase the difficulty by adding more dimensions while studying the implications.

5
Convergence

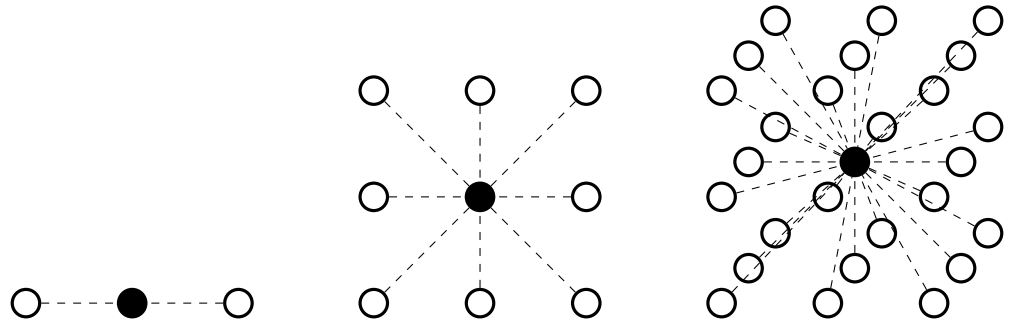
The idea of nested sampling is to allow for Bayesian computations in large-scale problems where conventional MCMC methods struggle and analytical approximations are inadequate. Thus far, the methods implemented in this work have not been evaluated in this large-dimensional regime as the focus has been on the specific implementation designs. These designs will in this chapter be put to the test and their performance in higher dimensions will be studied. When applicable the results will be compared to the corresponding results of the ellipsoidal sampling implementation `MultiNest` [10], which in this work has been run through the Python interface `PyMultiNest` [24].

5.1 The curse of dimensionality

First coined by Bellman [42], *the curse of dimensionality* has come to be an expression generally referring to various problematic phenomena associated with working in higher dimensions. Important in the context of sampling in general, and nested sampling in particular, is the exponential increase in the volume adjacent to a given region of space with the number of dimensions. Consider a Euclidean space in arbitrary dimensions, partitioned into a cubic grid. In one dimension, a grid point will be immediately surrounded by 2 neighbouring points, in two dimensions 8 points, in three dimensions 26 points and in n dimensions it will be surrounded by $3^n - 1$ points. This exponential growth is illustrated in Figure 5.1. An alternative point of view is to consider a hyperspherical coordinate system in n dimensions with one radial coordinate r and $n-1$ angular coordinates $\varphi_1, ...\varphi_{n-1}$. With an angular resolution of R points per angle range[1], there are R^{n-1} possible directions in which a new step can be proposed. For a quite moderate example with $R = 10$ and $n = 10$, we already have 10^9 possible directions, which demonstrates the vastness of high-dimensional spaces.

For the case of likelihood-constrained MCMC this means that it becomes increasingly unlikely for a proposed step to satisfy the constraint, simply because there are so many alternatives. This issue will, obviously, worsen as the active volume shrinks and the bulk of the posterior probability mass is approached. The immediate consequence is a dropping acceptance rate and, furthermore, inefficient exploration of the active region in the search for high-quality samples. The GMC procedure, described in Section 3.3.2 and implemented in Section 4.3, was in fact proposed [38] for its higher efficiency in higher dimensions as compared to e.g. Metropolis and ellipsoidal sampling. The idea is that the concept of a trajectory makes the exploration more guided and direct, reflecting away from unavailable regions whenever a proposed step is out of bounds. In the next section we will explore increasingly larger spaces, searching for the point where the algorithms break down.

[1] One of these angles takes values in $[0, 2\pi)$ and the remaining in $[0, \pi]$.



(a) $n = 1$ **(b)** $n = 2$ **(c)** $n = 3$

Figure 5.1: Illustration of the exponential increase of adjacent points with the dimensionality n of a Euclidean space. In (a): one, (b): two and (c): three dimensions, a grid point (filled circle) has 2, 8 and 26 neighbouring points (unfilled circles), respectively. In arbitrary dimensions, n, the corresponding number is $3^n - 1$.

5.2 Divergence in higher dimensions

With the considerations of Section 5.1 in mind, we here assess the performance of the three nested sampling versions in varying number of dimensions n. This is done for the toy problem presented in Sections 2.1.1 and 2.1.4 by considering different models $\mathcal{M}_n$, defined by the order $n-1$ of the polynomial model function in Equation (2.8). In other words, the nested sampling implementations are employed to perform a Bayesian model comparison, as was done analytically in Section 2.1.4. Figure 2.3 displays the evidence as a function of the number of dimensions, n, which should not be considered as a free parameter, but rather as an index for this specific model space, consisting of all possible models $\mathcal{M}_n$. In this section the numerically computed evidence for each model, $\ln Z$, is presented in terms of its difference from the analytically obtained evidence, $\ln Z_{\text{true}}$, i.e. $\ln Z - \ln Z_{\text{true}} = \ln(Z/Z_{\text{true}})$.

The results are seen in Figure 5.2, including the error of the computed evidence $\ln(Z/Z_{\text{true}})$ (top row), the bulk median acceptance rate $\mathcal{M}_{\mathcal{H}}[r_{\text{a}}]$ (middle row), defined in Equation (4.5), and the number of likelihood evaluations $N_{\mathcal{L}}$ (bottom row). The sampling parameters are $N = 1000$ and $f_{\ln} = 0.01$ and each point is, as before, the average of five identical runs with different random seeds and the bands are the corresponding standard deviations. The parameter settings in the left column are $\langle N_{\text{s}} \rangle = 40$, $s = 0.5$, $\tau = 0.1$, $a = 2.0$ and in the right column $\langle N_{\text{s}} \rangle = 80$, $s = 0.25$, $\tau = 0.05$, $a = 1.5$.

For comparison we have included the corresponding results from the ellipsoidal sampling method (see Section 3.3.4) for the evidence error (top row), obtained using the implementation `PyMultiNest`. We observe a clear method-wide drop in accuracy, as both overestimates $\ln(Z/Z_{\text{true}}) > 0$ and underestimates $\ln(Z/Z_{\text{true}}) < 0$ become apparent as the number of dimensions is increased. In the left panel, the Metropolis and ellipsoidal methods start to lose accuracy around $n = 4$–5. For the Metropolis method this can be seen to coincide with a drastic decline in the acceptance rate (middle row, left column) to $\mathcal{M}_{\mathcal{H}}[r_{\text{a}}] < 0.1$ for $n \geq 4$. For $n = 40$, the acceptance rate is essentially zero for the Metropolis method, this is further indicated by the number of likelihood evaluations (bottom row, left column) which is, quite literally, increasing off the charts. As shown in

the top-right corner inset, it grows by almost an order of magnitude as the number of dimensions goes from $n = 24$ to $n = 40$. The evidence error curves for the GMC and stretch move methods stay flat longer, up to around $n = 16$ and $n = 8$ respectively. GMC, however, is for lower n overestimating the evidence and the fact that it does so consistently indicates that it has better precision than accuracy. The stretch move is fairly accurate in its entire flat range, before it too drops. The behavior of the stretch move accuracy is to prefer as it is more trustworthy in its accurate range, even if it drops sooner than GMC.

As mentioned, the inaccuracy of the evidence computations split up in two classes depending on whether the evidence becomes overestimated, $\ln(Z/Z_{\text{true}}) > 0$ (Metropolis and `PyMultiNest`), or if it becomes underestimated, $\ln(Z/Z_{\text{true}}) < 0$ (GMC and stretch move), for larger n. The reason for this division is not clearly understood and needs further investigation. We can argue, nevertheless, that an overestimation of the evidence means that the terms $L_k\Delta\xi_k$, constituting the evidence according to Equation (3.13), are too large. This in turn indicates that significant likelihood values L_k are reached too quickly, before the prior mass elements $\Delta\xi_k \propto e^{-k/N}$ have had enough time to decay[2]. An underestimation is, by the same argument, caused by significant likelihood values being reached too slowly compared to the decay of the prior mass element. This phenomenon is emergent in Figure 5.3 which, similarly to Figure 4.4, tracks the sampling progress for the three methods, although here in $n = 24$ dimensions. It is clear that the methods do not agree; Metropolis is seen to reach the bulk first, followed by GMC and last the stretch move. This is in accordance with the evidence results displayed in Figure 5.2 (top left), where Metropolis gives the largest value and stretch move the lowest for $n = 24$. Thus, both Metropolis and ellipsoidal sampling (`PyMultiNest`) appear to inhibit a bias towards larger likelihood values. However, more research is required to understand the fundamental cause of these different behaviors.

The $\langle N_{\text{s}} \rangle = 80$ case in the right column of Figure 5.2, agrees qualitatively with the $\langle N_{\text{s}} \rangle = 40$ case in the left column. The motivation for increasing the number of steps while decreasing the scale parameters is to effectively increase the resolution since a given active volume constitutes smaller and smaller fractions of parameter space in higher dimensions. This will increase the acceptance rate but at the cost of a shorter distance covered per step, thereof the increased number of steps. There is an obvious improvement for all methods in that the evidence error curves (top right) are more accurate for a larger number of dimensions compared to the $\langle N_{\text{s}} \rangle = 40$ case (top left). Moreover, comparing the acceptance rates (middle right), the declines are more moderate, as expected since smaller step sizes mean less tendency to propose steps to points outside of the active region. From this we conclude that it is better to take a small step than no step at all. However, examining the number of likelihood evaluations $N_{\mathcal{L}}$ (bottom right), we see that we pay for the improved performance with more computational effort, which, as expected with twice as many steps, is roughly twice as high. The exception is the Metropolis curve which in the right panel does not show any drastic increase, as was observed in the left panel. We thus note that the scale $s = 0.5$ is undoubtedly too high for $n \geq 4$ whereas $s = 0.25$ seems more optimal.

While discussing the number of likelihood evaluations it ought to be mentioned that the `PyMultiNest`-evidence computations presented in Figure 5.2 all terminated after $N_{\mathcal{L}} \approx$

[2]There is also the possibility that the elements $\Delta\xi_k$ are inadequately modelled, which however should affect all methods similarly.

4.0–$6.5 \cdot 10^4$ likelihood evaluations, tabulated in Table 5.1. In contrast, the methods implemented in this work used 10–100 times as many likelihood evaluations, also shown in Table 5.1. This is a consequence of the intrinsic properties of MCMC methods, i.e. that many steps are generally required to generate an independent sample, and demonstrates an obvious advantage of the ellipsoid method. Furthermore, the number of likelihood evaluations used by `PyMultiNest` is observed to first increase up until around $n = 9$, and then decrease slowly up to $n = 40$. This highly non-intuitive behavior does in fact go against results presented in [10, Table 3.], where a clear monotonic increase in $N_{\mathcal{L}}$ was observed when applying `MultiNest` to a toy problem[3] for increasing n. This discrepancy is probably due to problem-specific details which make direct comparison inappropriate. As seen in Figure 2.3, the evidence in our toy problem has a steep increase from $n = 2$ to the maximum value at $n = 3$ and is then practically constant for $n \geq 5$. This very specific behavior differs from the dimensional dependence of the evidence in [10, Table 2.] which is strictly decreasing with n, possibly causing the differences in the number of `MultiNest`-likelihood evaluations used for different dimensions in the two problems.

Another aspect of nested sampling in successively higher dimensions is the decreasing density of active points. We have for this discussion kept the number of active points N fixed for every dimensionality. However, we may assume that this is not optimal and that one in general needs to increase the number of active points for higher dimensions. We can therefore expect to be able to improve the performance of the methods in general and the results in Figure 5.2 in particular. The computational cost will however increase as the number of steps (or iterations) needed are roughly $k \propto N$ (see Section 3.2.4).

In this section the nested sampling methods have been observed to break down, producing demonstrably erroneous evidence values. In the next section we will discuss how the true error stands in relation to the log evidence uncertainty estimate $\pm\sqrt{\mathcal{H}/N}$ discussed in Section 3.2.5.

[3] This is not the same toy problem that is considered in this work.

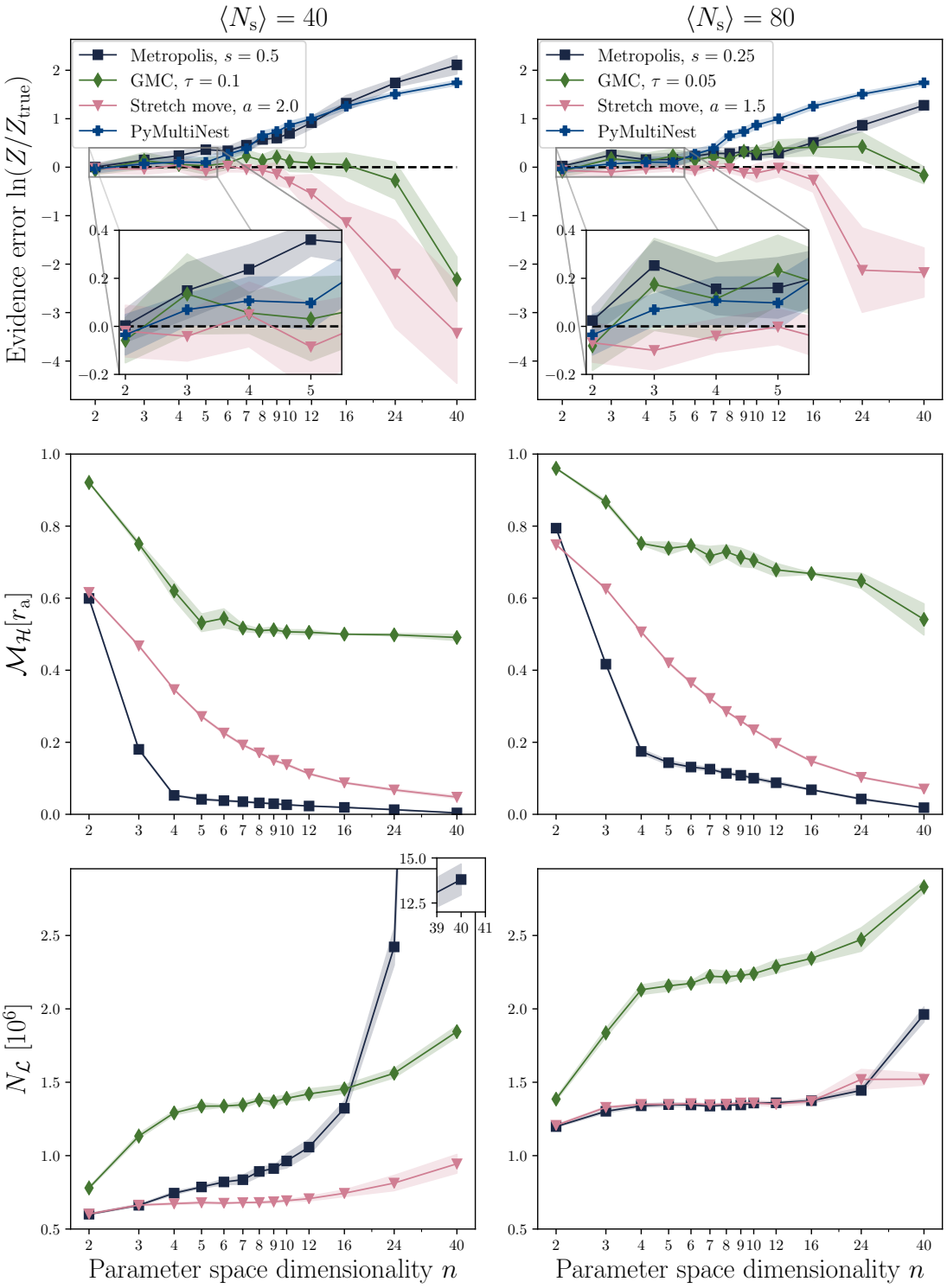


Figure 5.2: The evidence error (top row), the bulk median acceptance rate (middle row) and the number of likelihood evaluations (bottom row) for varying parameter space dimensionality. Each point is the average of five identical runs and the bands are corresponding standard deviations. The left and right columns display two different choices of hyperparameters, respectively, as specified in the figure. Sampling parameters are $N = 1000$ and $f_{\ln} = 0.01$. The overall trend is that performance worsens and computations become less accurate for higher dimensions, as expected.

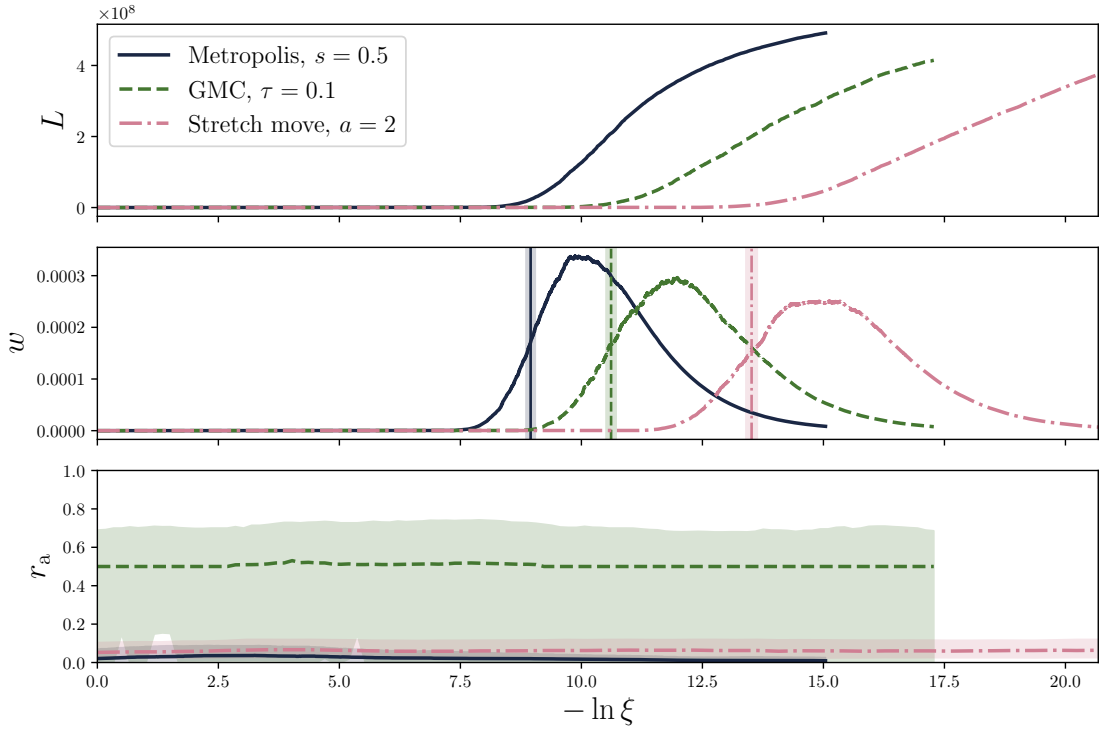


Figure 5.3: Tracked progress for the three methods, displayed in terms of the likelihood L, the posterior weights w and the acceptance rate r_a in $n = 24$ dimensions using $\langle N_\mathrm{s} \rangle = 40$. In contrast to the $n = 3$ case in Figure 4.4, the methods are seen to disagree on where the bulk of the posterior is located in terms of ξ, resulting in significant differences in the computed evidences. The vertical lines indicate the computed information $\mathcal{H}$. Sampling parameters are $N = 1000$ and $f_\mathrm{ln} = 0.01$.

5.2.1 Debunking the uncertainty estimate

The conventional nested sampling log evidence uncertainty estimate $\pm\sqrt{\mathcal{H}/N}$ was discussed in Section 3.2.5. The motivation for this particular expression is entirely based on an assessment of the uncertainty in the number of nested sampling iterations k required to reach the bulk of the posterior probability mass based on the definition $\xi_k = e^{-k/N}$. Table 5.1 lists the computed log evidence values $\ln Z$ for models with varying dimensionality n along with the uncertainty estimate for the different methods. The evidence data is the same as that which was used to generate Figure 5.2. The analytically obtained log evidence values $\ln Z_\mathrm{true}$ (see Section 2.1.4) are also included for reference. We see that the uncertainty estimate exclusively evaluates to $\sqrt{\mathcal{H}/N} \approx 0.10$, independent of method, dimension and even the actual error $\ln Z - \ln Z_\mathrm{true} = \ln(Z/Z_\mathrm{true})$. It is thus clear that in this case, the provided uncertainty estimate gives little information of the actual accuracy and gives no indication on when the algorithms begin to struggle. In a real application this implies that it might be far from sufficient to simply present the computed log evidence along with the uncertainty estimate without any further assessment of the sampling performance. The statistical variation, indicated by the bands in Figure 5.2, is however seen to be close to the uncertainty estimates ~ 0.10 displayed in Table 5.1, at least for the more accurate regime $n \lesssim 7$. From this we can conclude that the uncertainty estimate only seems to be reliable when the evidence computation works as intended, a statement of very little use in any real world application.

Table 5.1: The computed (log) evidence $\ln Z$ along with the uncertainty estimate $\pm\sqrt{\mathcal{H}/N}$ and the number of likelihood evaluations $N_{\mathcal{L}}$ required as functions of the parameter space dimensionality n. The analytical values $\ln Z_{\text{true}}$ are given for reference. Hyperparameter settings are the same as in the left column of Figure 5.2. Every value is the average obtained from five identical runs.

		Metropolis		GMC		Stretch move		`PyMultiNest`	
n	$\ln Z_{\text{true}}$	$\ln Z \pm \sqrt{\frac{\mathcal{H}}{N}}$	$N_{\mathcal{L}}$	$\ln Z \pm \sqrt{\frac{\mathcal{H}}{N}}$	$N_{\mathcal{L}}$	$\ln Z \pm \sqrt{\frac{\mathcal{H}}{N}}$	$N_{\mathcal{L}}$	$\ln Z \pm \sqrt{\frac{\mathcal{H}}{N}}$	$N_{\mathcal{L}}$
2	6.40	6.40 ± 0.10	$6.0 \cdot 10^5$	6.34 ± 0.10	$7.8 \cdot 10^5$	6.38 ± 0.10	$6.0 \cdot 10^5$	6.36 ± 0.10	$5.0 \cdot 10^4$
3	8.09	8.24 ± 0.10	$6.6 \cdot 10^5$	8.23 ± 0.10	$1.1 \cdot 10^6$	8.05 ± 0.10	$6.6 \cdot 10^5$	8.16 ± 0.10	$5.0 \cdot 10^4$
4	7.96	8.20 ± 0.10	$7.4 \cdot 10^5$	8.02 ± 0.10	$1.3 \cdot 10^6$	8.01 ± 0.10	$6.7 \cdot 10^5$	8.07 ± 0.10	$5.2 \cdot 10^4$
5	7.93	8.30 ± 0.10	$7.9 \cdot 10^5$	7.97 ± 0.10	$1.3 \cdot 10^6$	7.85 ± 0.10	$6.8 \cdot 10^5$	8.03 ± 0.10	$5.6 \cdot 10^4$
6	7.93	8.27 ± 0.10	$8.2 \cdot 10^5$	8.02 ± 0.10	$1.3 \cdot 10^6$	7.96 ± 0.10	$6.8 \cdot 10^5$	8.20 ± 0.10	$5.9 \cdot 10^4$
7	7.93	8.39 ± 0.10	$8.4 \cdot 10^5$	8.16 ± 0.10	$1.3 \cdot 10^6$	7.89 ± 0.10	$6.8 \cdot 10^5$	8.31 ± 0.10	$6.3 \cdot 10^4$
8	7.93	8.50 ± 0.10	$8.9 \cdot 10^5$	8.07 ± 0.10	$1.4 \cdot 10^6$	7.87 ± 0.10	$6.8 \cdot 10^5$	8.58 ± 0.10	$6.2 \cdot 10^4$
9	7.93	8.53 ± 0.10	$9.1 \cdot 10^5$	8.14 ± 0.10	$1.4 \cdot 10^6$	7.79 ± 0.10	$6.9 \cdot 10^5$	8.67 ± 0.10	$6.5 \cdot 10^4$
10	7.93	8.63 ± 0.10	$9.6 \cdot 10^5$	8.05 ± 0.10	$1.4 \cdot 10^6$	7.63 ± 0.10	$6.9 \cdot 10^5$	8.80 ± 0.10	$6.3 \cdot 10^4$
12	7.93	8.85 ± 0.10	$1.1 \cdot 10^6$	8.02 ± 0.10	$1.4 \cdot 10^6$	7.38 ± 0.10	$7.1 \cdot 10^5$	8.93 ± 0.10	$6.2 \cdot 10^4$
16	7.93	9.25 ± 0.10	$1.3 \cdot 10^6$	7.98 ± 0.10	$1.5 \cdot 10^6$	6.79 ± 0.11	$7.4 \cdot 10^5$	9.19 ± 0.10	$5.7 \cdot 10^4$
24	7.93	9.67 ± 0.09	$2.4 \cdot 10^6$	7.66 ± 0.10	$1.6 \cdot 10^6$	5.73 ± 0.11	$8.1 \cdot 10^5$	9.44 ± 0.10	$4.9 \cdot 10^4$
40	7.93	10.04 ± 0.09	$1.4 \cdot 10^7$	5.62 ± 0.11	$1.8 \cdot 10^6$	4.51 ± 0.11	$8.4 \cdot 10^5$	9.67 ± 0.10	$4.1 \cdot 10^4$

There has been no attempt to investigate the fundamental reasons for the behavior of the nested sampling uncertainty estimate in this work. It is nevertheless a central topic and crucial for the application of nested sampling and should therefore be thoroughly studied. From the results presented in this work we can at least argue, however, that the acceptance rate r_{a} is a key quantity for assessing the performance of the nested sampling computations. We see, e.g. in Figure 4.5 and Figure 5.2, that the inaccuracy of the evidence computations generally worsens when the bulk median acceptance rate $\mathcal{M}_{\mathcal{H}}[r_{\text{a}}]$ drops too low. Thus, by monitoring the acceptance rates we get a rough indication of the quality of the computations and might thereby determine whether the results are worth considering or if they should be disposed of as invalid. Very roughly, for this particular problem we should be alerted if we for the Metropolis or stretch move methods have $\mathcal{M}_{\mathcal{H}}[r_{\text{a}}] \lesssim 0.2$ and if we for the GMC method have $\mathcal{M}_{\mathcal{H}}[r_{\text{a}}] \lesssim 0.6$.

6
Conclusion

Bayesian inference is a powerful tool for any activity that involves making predictions from data and assessing the uncertainties of these predictions. This obviously includes any scientific research, in which the common scenario is to have one or several competing models describing certain quantities or phenomena. Each model is typically associated with a set of model parameters whose appropriate values generally need to be inferred from data. Model comparison and parameter estimation are key ingredients in the Bayesian recipe and provides systematic treatment of models and model parameters in these scenarios by constructing probability distributions. A typical example is effective field theories in subatomic physics. These include so called observable coefficients that are ideal subjects for Bayesian parameter estimation. In addition, the study of effective field theories at different truncation orders and with different choices for the degrees of freedom can easily be formulated as a model comparison problem. However, Bayesian inference procedures are generally dependent on expensive numerical efforts such as the computation of the model evidence integral over the, commonly high-dimensional, parameter space. It is for this purpose important to develop and evaluate numerical methods which are accurate, reliable and versatile.

Summary

Nested sampling is a method which naturally provides an estimate of the model evidence used in Bayesian model comparison while it at the same time generates samples from the posterior probability distribution for the parameters. The main challenge of nested sampling is identified as the sampling of the constrained prior probability distribution. It is also the main topic of this thesis where we have described, implemented and compared three different methods for generating samples from a constrained probability distribution in a nested sampling framework. Each method utilizes their own version of an MCMC random walk to explore the prior parameter space subject to successively stricter bounds from iso-likelihood surfaces. We have applied the implemented methods on an EFT-inspired toy problem and evaluated their respective performances. The performance is observed to vary between the methods and shows considerable sensitivity to the choice of hyperparameters. All methods are nevertheless able to estimate the model evidence within a reasonable accuracy, at least for modestly sized parameter spaces. Some of the implementations even outperform existing state-of-the-art software in terms of accuracy when applied to the same problem.

The first and simplest of the implemented constrained sampling methods was the constrained Metropolis procedure. It was based on the well-known standard Metropolis algorithm but modified to operate under the likelihood constraint. Furthermore, it was deemed necessary to introduce a scheme for adjusting the scale of the Metropolis proposal

distribution in unison with the evolving constraint. This scheme utilized the current set of nested active points to update the covariance matrix of a Gaussian proposal distribution. Adaption to changes in scale is obviously a recurrent necessity for the methods implemented in this work in particular and is a key aspect in the context of nested sampling in general.

The second implementation was based on Galilean Monte Carlo — a method which was originally introduced specifically with the intention that it be used in conjunction with nested sampling. Whereas the Metropolis exploration is relatively slow and diffusive, the idea of GMC is to explore parameter space more efficiently by simulating trajectories according to classical mechanics, especially in higher dimensions. The implemented GMC version was, similarly to the Metropolis method, modified with an update scheme for the initial velocity proposal distribution based on the current set of active points.

The third method implemented in this work is a version of affine-invariant sampling, referred to as the stretch move, modified to work under the nested sampling restrictions. This method is unique to this work, it has to our knowledge not been implemented elsewhere. The basic idea is however quite simple; the scale of the nested active region changes sequentially, making it natural to use a sampling algorithm which is affine- and therefore also scale-invariant. The constrained stretch move is arguably the most accurate and robust constrained sampling method evaluated in this work.

The methods have two general types of hyperparameters: a nominal number of exploration steps $\langle N_\mathrm{s} \rangle$ and a scale parameter. The behaviors of the methods depending on the choices of hyperparameters were studied and it was concluded that the values generally needed to be finely tuned to achieve optimal performance. The accuracy of the computed evidence was generally seen to improve for larger $\langle N_\mathrm{s} \rangle$ — an expected observation since a larger number of steps is assumed to result in lower correlation, and thereby higher quality, of new samples. However, more steps comes at the expense of an increasing computational cost, primarily from the increasing number of likelihood evaluations. For each method, larger scale parameter values were observed to cause a decrease in the acceptance rate. This effect is particularly evident for the Metropolis and GMC methods for which the choice of scale settings was observed to significantly alter the evidence accuracy. The acceptance rate for a run was quantified by introducing the bulk median, used to measure any quantity in the region that is of most significance: in and around the bulk of the posterior mass. The stretch move method generally showed greater stability towards changes in hyperparameters values, both $\langle N_\mathrm{s} \rangle$ and scale, which implies that less effort is needed from the user in terms of finding the optimal values for a specific problem.

By applying the nested-sampling implementations in a model-comparison context within the EFT toy problem, the evidence accuracy and general behavior was examined for models with varying parameter space dimensionalities n. Here the methods were compared to the state-of-the-art ellipsoidal nested sampling implementation `MultiNest` [10]. For sufficiently large n, all methods, including `MultiNest`, begin to struggle in maintaining the evidence accuracy achieved for lower n. This is a result of the exponential growth of space with increasing number of dimensions, referred to as the curse of dimensionality. However, we demonstrated that it is possible to maintain accuracy in higher dimensions by decreasing the scale parameter value, while the number of steps at the same time is increased. It is further expected that accuracy can be maintained by increasing the number of active

points N, an approach which was not explored in this work. Both increasing the number of steps and increasing the number of active points naturally comes with additional computational cost, an unavoidable consequence for sampling high-dimensional spaces. In agreement with its design intentions, the GMC method has in this work been able to maintain its accuracy for the largest dimensionalities, despite not being the most accurate for lower dimensions. Thus, the GMC method shows the most potential for sampling high-dimensional spaces if care is taken to adjust and tune it to provide better accuracy overall.

A general concern highlighted in this work regards the nested sampling log evidence uncertainty estimate $\pm\sqrt{\mathcal{H}/N}$ (see Section 3.2.5) and the situations where the algorithms fail to compute the evidence. When a computation fails, e.g. because of a poor choice of hyperparameters and/or too high dimensionality, there is no indication in this uncertainty measure that this has occurred. In fact, we have observed the uncertainty estimate to be practically unaffected by the actual sampling performance and that it only gives reasonable estimates of the statistical variations when the computation is accurate and reliable. This was the case for the methods implemented in this work, and also for `MultiNest`. Consequently, an advice of caution concerning nested sampling in general is that it might not be sufficient to simply state a computed evidence value along with the uncertainty estimate without having acquired any further insight into the inner workings of the sampling. We have not made any attempt at further investigation of the nested sampling uncertainty estimate. However, we have argued that for the methods implemented in this work it is possible to obtain an indication of the sampling performance by monitoring the acceptance rate during a run. Different hyperparameter settings can be compared by using the bulk median. The acceptance rate should in general neither be too high nor too low, even though the latter is more likely to be a concern in the context of nested sampling.

When failing to compute the evidence the methods split up in two distinct behaviors: the Metropolis and `MultiNest` methods consistently overestimated the evidence and the GMC and stretch-move methods consistently underestimated it. By monitoring the sampling progress, the over- or underestimation were observed to relate to whether higher likelihood values L_k were reached more quickly or more slowly, respectively, in relation to the modelled decay of the prior mass elements $\Delta\xi_k \propto e^{-k/N}$. Determining the fundamental cause for these distinct behaviors needs further investigation.

Further work

To quantify the quality of MCMC samples it is common practice to compute the auto-correlation function (ACF) and its associated auto-correlation time [5]. The auto-correlation time quantifies the number of steps needed to reach an uncorrelated sample starting from any given point. However, while employing MCMC methods in the nested sampling context it is not straightforward to obtain these quantities since one would obtain an ACF for every iteration. A scheme is therefore needed to compress the information from every obtained ACF into a single function. Moreover, the ACFs would need to be averaged over several Markov chains in every iteration to reduce statistical noise, causing a significant addition to the computational cost. It would nonetheless be desirable to implement such a scheme in order to obtain information on how the auto-correlation differs between the methods and varies with hyperparameter settings. A suggestion for this scheme is to first compute the auto-correlation time averaged over several chains in each nested sampling iteration. Then, one would obtain the auto-correlation time as a function of the prior

mass ξ, which would allow to compute the corresponding bulk median, equivalent to what was done for the acceptance rate r_{a} in this work. Acquiring this information would give us better understanding of the different methods and the influence of the hyperparameters.

In this work we proposed procedures for adapting the Metropolis and GMC proposal distributions to the shrinking active region by adjusting the corresponding covariance matrices based on the set of active points. These procedures have not been fully evaluated and validated, which could be done, however, by studying the evolution of the proposal distributions over the course of a nested sampling run. Doing so, one could (for simple likelihood functions) compare the extent of the proposal distributions with the size of the current active region and thereby determine how the proposed procedure could be further optimized. This analysis may also be done for the stretch move method, although the proposal distribution has a different format.

As discussed, the nested sampling performance is expected to improve by increasing the number of active points N, however at an increased computational cost. Increasing N implies a higher resolution of parameter space and decreases the numerical integration error. These effects have neither been quantified nor confirmed in this work and should be considered in future studies.

Throughout this work the implemented nested sampling methods have been evaluated using a toy problem with a simple, Gaussian likelihood function, enabling comparisons to analytically derived results. This is advantageous as it allows for exact assessment of the computational accuracy and deep insight into the sampling process. However, the toy problem is not fully representative of real applications where the problem could be of arbitrary complexity. This includes likelihoods which are non-Gaussian, multimodal, curved or degenerate with respect to the parameters. The difficulty of testing sampling algorithms is that results will always be problem-specific and conclusions might therefore differ between applications. In order to gain more general understanding of the implemented methods they should be applied to a broader variety of problems. This would provide insight into what optimization and adjustments that could be done to increase the robustness and usefulness for general nested-sampling applications.

References

[1] R. Machleidt and D. Entem, "Chiral effective field theory and nuclear forces", Physics Reports **503**, 1–75 (2011).

[2] D. B. Kaplan, *Five lectures on effective field theory*, 2005, arXiv:`nucl-th/0510023 [nucl-th]`.

[3] D. J. C. MacKay, *Information Theory, Inference, and Learning Algorithms* (Cambridge Univ. Press, 2003).

[4] D. S. Sivia and J. Skilling, *Data analysis : a Bayesian tutorial.* Oxford science publications (Oxford University Press, 2006).

[5] S. Brooks et al., *Handbook of Markov Chain Monte Carlo* (Chapman & Hall/CRC., May 2011).

[6] N. Metropolis et al., "Equation of State Calculations by Fast Computing Machines", The Journal of Chemical Physics **21**, 1087–1092 (1953).

[7] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications", Biometrika **57**, 97–109 (1970).

[8] J. Skilling, "Nested Sampling", AIP Conference Proceedings **735**, 395–405 (2004).

[9] J. Skilling, "Nested sampling for general Bayesian computation", Bayesian Analysis **1**, 833–859 (2006).

[10] F. Feroz, M. P. Hobson, and M. Bridges, "MultiNest: an efficient and robust Bayesian inference tool for cosmology and particle physics", Monthly Notices of the Royal Astronomical Society **398**, 1601–1614 (2009).

[11] F. Feroz and M. P. Hobson, "Multimodal nested sampling: an efficient and robust alternative to Markov Chain Monte Carlo methods for astronomical data analyses", Monthly Notices of the Royal Astronomical Society **384**, 449–463 (2008).

[12] P. Mukherjee, D. Parkinson, and A. R. Liddle, "A Nested Sampling Algorithm for Cosmological Model Selection", The Astrophysical Journal **638**, L51–L54 (2006).

[13] D. Parkinson and A. R. Liddle, "Bayesian model averaging in astrophysics: a review", Statistical Analysis and Data Mining: The ASA Data Science Journal **6**, 3–14 (2013).

[14] Planck Collaboration, "Planck 2013 results. XXII. Constraints on inflation", Astronomy & Astrophysics **571**, A22 (2014).

[15] J. Veitch et al., "Parameter estimation for compact binaries with ground-based gravitational-wave observations using the LALInference software library", Physical Review D **91**, 042003 (2015).

[16] S. Aitken and O. E. Akman, "Nested sampling for parameter inference in systems biology: application to an exemplar circadian model", BMC Systems Biology **7**, 72 (2013).

[17] R. Dybowski et al., "Nested Sampling for Bayesian Model Comparison in the Context of Salmonella Disease Dynamics", PLOS ONE **8**, 1–17 (2013).

[18] N. Pullen and R. J. Morris, "Bayesian Model Comparison and Parameter Inference in Systems Biology Using Nested Sampling", PLOS ONE **9**, 1–11 (2014).

[19] F. Feroz and J. Skilling, "Exploring multi-modal distributions with nested sampling", AIP Conference Proceedings **1553**, 106–113 (2013).

[20] W. J. Handley, M. P. Hobson, and A. N. Lasenby, "polychord: nested sampling for cosmology", Monthly Notices of the Royal Astronomical Society: Letters **450**, L61–L65 (2015).

[21] W. J. Handley, M. P. Hobson, and A. N. Lasenby, "polychord: next-generation nested sampling", Monthly Notices of the Royal Astronomical Society **453**, 4384–4398 (2015).

[22] J. Buchner, "A statistical test for Nested Sampling algorithms", Statistics and Computing, 1–10 (2014), arXiv:1407.5459 [stat.CO].

[23] J. S. Speagle, "dynesty: a dynamic nested sampling package for estimating Bayesian posteriors and evidences", Monthly Notices of the Royal Astronomical Society **493**, 3132–3158 (2020).

[24] J. Buchner et al., "X-ray spectral modelling of the AGN obscuring region in the CDFS: Bayesian model selection and catalogue", Astronomy & Astrophysics **564**, A125 (2014).

[25] M. Schindler and D. Phillips, "Bayesian methods for parameter estimation in effective field theories", Annals of Physics **324**, 682–708 (2009).

[26] S. Wesolowski et al., "Bayesian parameter estimation for effective field theories", Journal of Physics G: Nuclear and Particle Physics **43**, 074001 (2016).

[27] E. T. Jaynes, "Information Theory and Statistical Mechanics", Physical Review **106**, 620–630 (1957).

[28] M. I. Gordin and B. A. Lifšic, "Central limit theorem for stationary Markov processes", Soviet Mathematics, Doklady **19**, 392–394 (1978).

[29] S. Duane et al., "Hybrid Monte Carlo", Physics Letters B **195**, 216–222 (1987).

[30] J. A. Christen, *A general purpose scale-independent MCMC algorithm*, tech. rep. I-07-16 (CIMAT, Guanajuato, Mexico, 2007).

[31] J. Goodman and J. Weare, "Ensemble samplers with affine invariance", Communications in Applied Mathematics and Computational Science **5**, 65–80 (2010).

[32] D. Foreman-Mackey et al., "emcee: The MCMC Hammer", Publications of the Astronomical Society of the Pacific **125**, 306–312 (2013).

[33] D. Schittenhelm and P. Wacker, *Nested Sampling And Likelihood Plateaus*, 2020, arXiv:2005.08602 [math.ST].

[34] A. Fowlie, W. Handley, and L. Su, *Nested sampling cross-checks using order statistics*, 2020, arXiv:2006.03371 [stat.CO].

[35] H. A. David and H. N. Nagaraja, *Order Statistics*, Wiley Series in Probability and Statistics (John Wiley & Sons, Inc., Hoboken, New Jersey, 2003).

[36] K. P. Burnham and D. R. Anderson, *Model Selection and Multimodel Inference* (Springer, 2002).

[37] M. Betancourt, "Nested Sampling with Constrained Hamiltonian Monte Carlo", AIP Conference Proceedings **1305**, 165–172 (2011).

[38] J. Skilling, "Bayesian computation in big spaces-nested sampling and Galilean Monte Carlo", AIP Conference Proceedings **1443**, 145–156 (2012).

[39] J. Skilling, "Galilean and Hamiltonian Monte Carlo", Proceedings **33**, 19 (2019).

[40] J. R. Shaw, M. Bridges, and M. P. Hobson, "Efficient Bayesian inference for multimodal problems in cosmology", Monthly Notices of the Royal Astronomical Society **378**, 1365–1370 (2007).

[41] S. Manikandan, "Measures of central tendency: Median and mode", Journal of pharmacology & pharmacotherapeutics **2**, 214–215 (2011).

[42] R. Bellman, *Adaptive control processes: A guided tour. (A RAND Corporation Research Study)*. Princeton, N. J.: Princeton University Press, XVI, 255 p. 1961.

[43] D. Foreman-Mackey, "corner.py: Scatterplot matrices in Python", The Journal of Open Source Software **24** (2016).

A
Posterior probability distributions

The nested sampling algorithm provides two main outputs: the model evidence and a set of parameter samples from the posterior joint probability density function. Throughout this work the focus has been on the estimation of the evidence as the main quantity of interest. However, for completeness we here present and compare examples of obtained posterior samples, visualized using marginal distributions similar to those of Figure 2.2. We described in Section 3.2.2 how the list of discarded points from each nested sampling iteration can be interpreted as a set of samples $\{\boldsymbol{\theta}_k\}$, weighted with probabilities w_k. This means that a sample from the posterior is produced according to:

1. obtain the list of samples and weights $(\boldsymbol{\theta}_k, w_k)$ from a nested sampling run
2. choose index k randomly with probability w_k
3. give the corresponding $\boldsymbol{\theta}_k$.

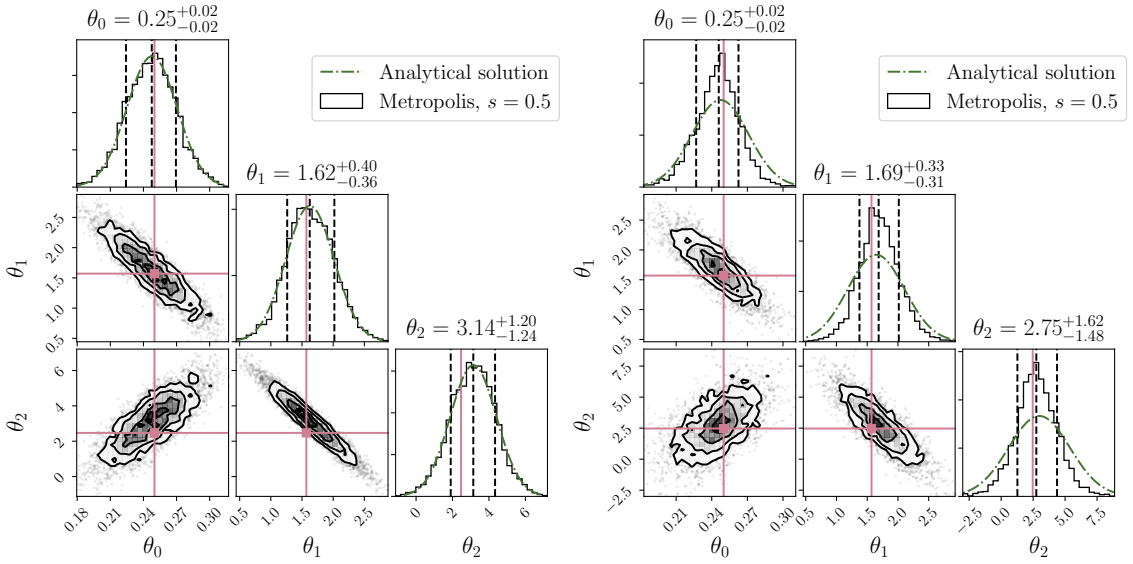
Moreover, it is possible to choose a subset of $\{\boldsymbol{\theta}_k\}$ such that this subset becomes equally weighted using staircase sampling [4]. This is done by constructing a cumulative staircase function

$$\mathcal{S}_k = u + \nu \sum_{j=1}^{k} w_j, \quad u \sim U(0,1), \tag{A.1}$$

where ν is the desired number of samples in the subset. By gradually increasing k and recording its value every time $\mathcal{S}_k$ exceeds an integer 1,2,3,... we obtain a list of indices that can be used to extract the equally-weighted subset. In this way, every sample $\boldsymbol{\theta}_k$ appears in the new set with mean multiplicity $\langle n_k \rangle = \nu w_k$.

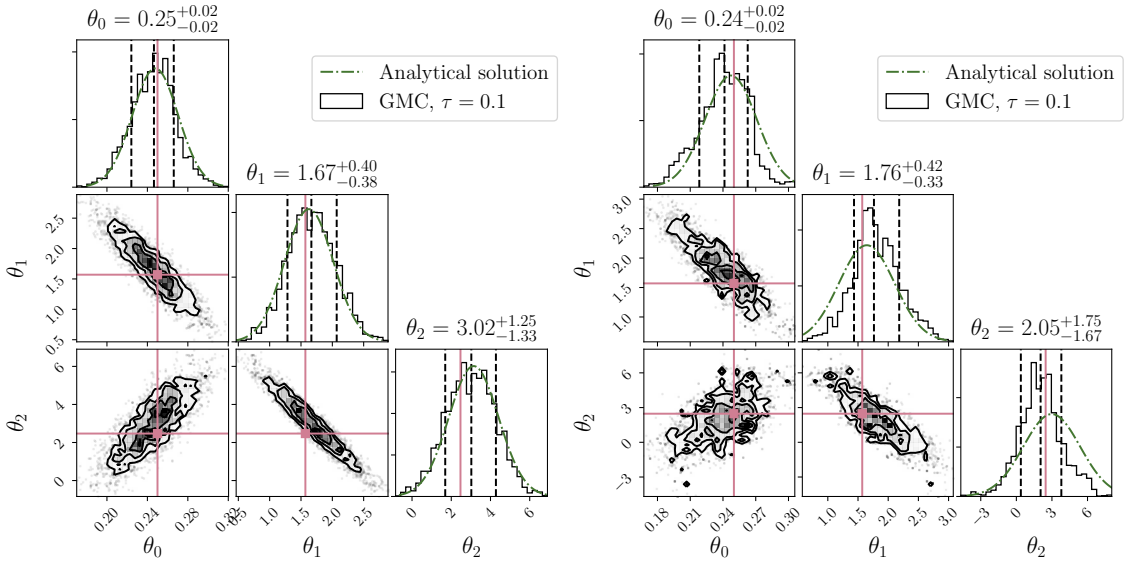
Probability distributions represented by equally-weighted samples obtained through staircase sampling are here displayed as histograms in one and two dimensions by binning of the samples and marginalization. Figures A.1, A.2, A.3 and A.4 show the histograms from the Metropolis method, GMC, the stretch move method and `PyMultiNest`, respectively, for the EFT toy example with $n = 3$ and $n = 24$ model parameters. The figures are created using the Python module `corner` [43]. For the $n = 24$ case only the 3 first parameters are displayed for comparability and clarity. The distributions (histograms) are compared to the analytical solution, discussed in Section 2.1.4, for the one-dimensional marginalizations on the diagonals and it is clear that all methods produce quite accurate results for $n = 3$. In contrast, the accuracy for $n = 24$ is significantly lower compared to the analytical solution. This is obviously expected bearing in mind the discussions on the curse of dimensionality in Chapter 5. It should further be stressed that the total number of samples are roughly equal for the two dimensionalities which implies a drastically lower sample density for $n = 24$ compared to $n = 3$. However, Metropolis and `PyMultiNest` in Figures A.1b and A.4b seem to be able to produce more regular and sym-

metric distributions for $n = 24$ compared to GMC and the stretch move in Figures A.2b and A.3b. This division into two categories is in fact in agreement with the observation in Section 5.2 where the Metropolis and `PyMultiNest` methods consistently overestimates whereas GMC and the stretch move consistently underestimates the evidence. We also observe the bias towards higher likelihoods, discussed in Section 5.2, for Metropolis and `PyMultiNest` in that their distributions for $n = 24$ are too narrow and have too high maxima. As mentioned, these different behaviors need further investigation.



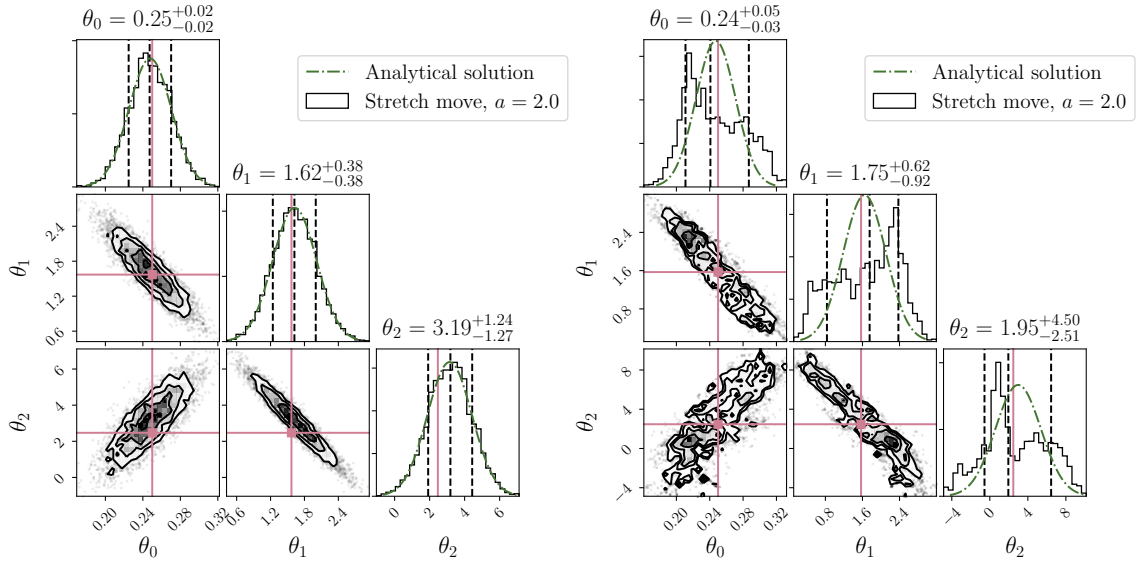
(a) $n = 3$ **(b)** $n = 24$ (only first 3 parameters displayed)

Figure A.1: Marginal posterior distributions obtained using the constrained Metropolis method with $s = 0.5$ and $\langle N_\mathrm{s} \rangle = 40$. The distributions are compared to the analytical solution (dash-dotted). Vertical dashed lines represent the sample median and 16^{th} and 84^{th} percentiles, respectively. The Taylor coefficients (Equation (2.17)) are indicated by the square markers and vertical solid lines. The histograms contain ~ 5000 samples in each case. $N = 1000$ and $f_\mathrm{ln} = 0.01$.



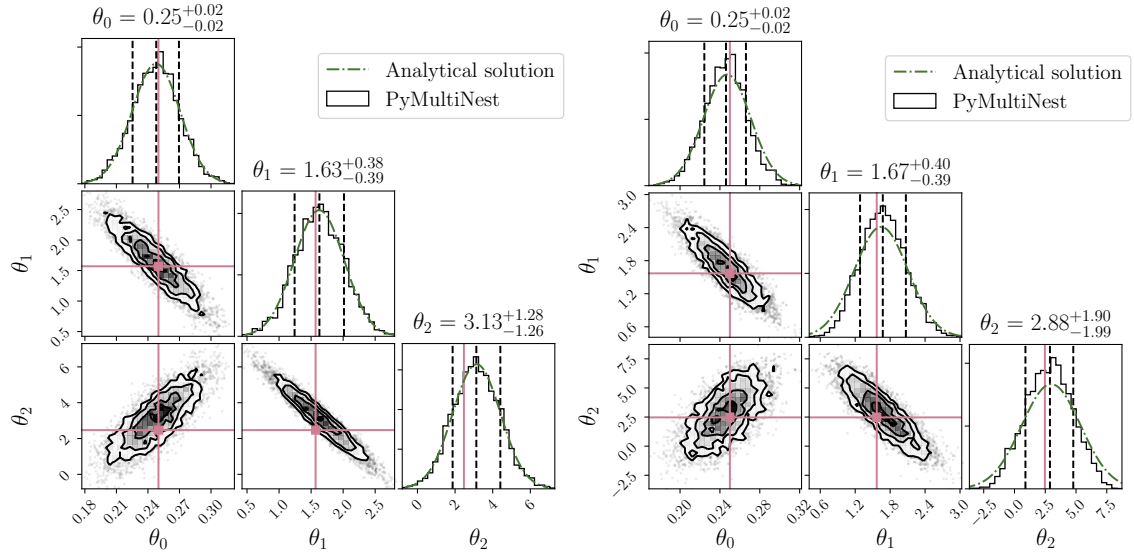
(a) $n = 3$ **(b)** $n = 24$ (only first 3 parameters displayed)

Figure A.2: Marginal posterior distributions obtained using the GMC method with $\tau = 0.1$ and $\langle N_\mathrm{s} \rangle = 40$. The distributions are compared to the analytical solution (dash-dotted). Vertical dashed lines represent the sample median and 16$^{\mathrm{th}}$ and 84$^{\mathrm{th}}$ percentiles, respectively. The Taylor coefficients (Equation (2.17)) are indicated by the square markers and vertical solid lines. The histograms contain $\sim$ 5000 samples in each case. $N = 1000$ and $f_\mathrm{ln} = 0.01$.



(a) $n = 3$ **(b)** $n = 24$ (only first 3 parameters displayed)

Figure A.3: Marginal posterior distributions obtained using the constrained stretch move method with $a = 2.0$ and $\langle N_\mathrm{s} \rangle = 40$. The distributions are compared to the analytical solution (dash-dotted). Vertical dashed lines represent the sample median and 16$^{\mathrm{th}}$ and 84$^{\mathrm{th}}$ percentiles, respectively. The Taylor coefficients (Equation (2.17)) are indicated by the square markers and vertical solid lines. The histograms contain $\sim$ 5000 samples in each case. $N = 1000$ and $f_\mathrm{ln} = 0.01$.



(a) $n = 3$ **(b)** $n = 24$ (only first 3 parameters displayed)

Figure A.4: Marginal posterior distributions obtained using `PyMultiNest`. The distributions are compared to the analytical solution (dash-dotted). Vertical dashed lines represent the sample median and 16$^{\text{th}}$ and 84$^{\text{th}}$ percentiles, respectively. The Taylor coefficients (Equation (2.17)) are indicated by the square markers and vertical solid lines. The histograms contain ~ 5000 samples in each case. $N = 1000$ and $f_{\text{ln}} = 0.01$.