Bayesian data analysis for gravitational wave astrophysics

C. Messenger[1]

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

E-mail: christopher.messenger@glasgow.ac.uk

Abstract. We present some basic notes on the core concepts at the heart of gravitational wave data analysis. We start with Bayes theorem and explore the ideas associated with search strategies via matched-filtering and then lead on to Bayesian parameter estimation and model selection.

1. Introduction

Before we get really started we note that the majority of the content presented here follows the arguments provided by [1, 2]. In addition, many of the topics discussed here are available as practical examples from the gravitational wave (GW) online science centre [3] where you can experiment with analysing real GW data, and Dr. Matthew Pitkin's excellent online resource [4] where you can experiment with a variety of different Bayesian parameter estimation and model selection codes.

My aim is to make some of this material understandable and clear in a way that I found the general literature *not* to be when I was learning this at the start of my adventure into GW astrophysics. In brief, everything starts with Bayes theorem so my advice is to always rely on it as a point of reference when you feel confused or lost.

We begin by defining one of the most useful quantities, the Fourier transform. Given a continuous function of time $x(t)$, it's Fourier transform and inverse are defined by

$$\tilde{x}(f) = \int_{-\infty}^{\infty} x(t)e^{-2\pi ift}dt \tag{1}$$

$$x(t) = \int_{-\infty}^{\infty} \tilde{x}(t)e^{2\pi ift}df. \tag{2}$$

The discrete Fourier transform defintion that we will use is

$$\tilde{x}_k = \Delta t \sum_{j=0}^{N-1} x_j e^{-2\pi ijk/N}$$

$$x_j = \Delta f \sum_{k=0}^{N-1} \tilde{x}_k e^{2\pi ijk/N} \tag{3}$$

where the timeseries elements x_j are uniformly sampled values of the underlying function $x(t)$ sampled at time $t_j = j\Delta t$. The corresponding frequency domain elements $\tilde{x}_k$ are sampled at $f_k = k/T$.

2. Bayes theorem

Let us start simply by asking what is the joint probability of 2 events A, B occuring. Using basic probability theory we can write this in 2 equivelent ways

$$P(A, B) = P(B, A)$$

$$P(A|B)P(B) = P(B|A)P(A) \tag{4}$$

where we have used the fact that the probability of both A *and* B is the same as the probability of B *and* A. The ordering does not matter. We have also used the notation $P(\cdot|\cdot)$ where the vertical line seperates the quantities for which the probability statement is addressing (left) and the quantities that are *given* or held fixed (right).

The next step is to simply perform a minor rearrangement to obtain Bayes Theorem

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}. \tag{5}$$

In this notation we are using the fact that $P(\cdot)$ indicates a probability (and not a probability density as we will discuss later).

In order for these probabilities to be correctly normalised we require that

$$\sum_{\{A\}} P(A|B) = 1 \tag{6}$$

where we are able to define $\{A\}$ as indicating all possible states of A, and hence

$$\sum_{\{A\}} P(A|B) = \frac{\sum_{\{A\}} P(B|A)P(A)}{P(B)}$$
$$P(B) = \sum_{\{A\}} P(B|A)P(A). \tag{7}$$

2.1. Simple discrete example

It may help to clarify some aspects of these seemingly obvious expressions with a simple example. Imagine Frances is getting married tomorrow, at an outdoor ceremony in the Scotland. In recent years, it has rained only 200 days each year and so we will state that $P(A) = 200/365$ and $P(\bar{A}) = 165/356$. In this case we are defining A as the state of "raining tomorrow", $\bar{A}$ as the state of "not raining tomorrow", and B as the state of "rain predicted".

Unfortunately, the weather report has predicted rain for tomorrow. When it actually rains, the weather report correctly forecasts rain 90% of the time and hence we are able to assign $P(B|A) = 0.9$. When it doesn't rain, they incorrectly forecasts rain 10% of the time and so $P(B|\bar{A}) = 0.1$. It therefore follows, that since there are only 2 allowed states of "raining tomorrow" we can conclude that

$$p(A|B) = \frac{p(B|A)P(A)}{P(B|A)P(A) + P(B|\bar{A})P(\bar{A})} = 0.92 \tag{8}$$

and hence Frances should bring an umbrella.

2.2. Continuous variables

If we were start again from Eq. 4 and instead use the continuous variables θ and x, rather than discrete variables, we would have written

$$\begin{aligned} p(\theta, x)\mathrm{d}\theta\mathrm{d}x &= p(x, \theta)\mathrm{d}x\mathrm{d}\theta \\ p(\theta|x)p(x)\mathrm{d}\theta\mathrm{d}x &= p(x|\theta)p(\theta)\mathrm{d}x\mathrm{d}\theta \\ p(\theta|x) &= \frac{p(x|\theta)p(\theta)}{p(x)} \end{aligned} \tag{9}$$

where the lower case $p(\cdot)$ indicates a probability density and the $d\theta, dx$ are infinitesimal changes in the continuous parameters θ, x respectively. It is clear here that they cancel immediately and we are left with Bayes theorem in exactly the same form as before.

We can now start labeling the individual components of this expression with their commonly used names. The right hand side (RHS) term $p(x|\theta)$ is known as the *likelihood* of the parameters θ, or simply the probability density function (PDF) of the data given the parameters. The second RHS numerator term $p(\theta)$ is known as the *prior* PDF (or just the prior) on the parameters. The denominator term $p(x)$ is known as the *Bayesian Evidence* or the *marginal likelihood*. Finally, the left hand side (LHS) $p(\theta|x)$ is known as the *posterior* PDF or simply the posterior.

Likelihood - The likelihood on the parameters θ is the probability of obtaining or measuring x (the data) given the parameters θ. This is usually the quantity that can be easily calculated for a given signal model (described by the parameters θ) and a given noise model describing the properties of the noisy detector.

Prior - The prior represents your state of knowledge on the parameters θ *before* you performed any measurements. It is vitally important that this information is independent of anything you might learn or assume about your measurement x or measurement apparatus. For example, if my GW detector cannot see sources below 10 Hz that does not mean that the prior on the frequency of a continuously emitting source is zero below 10 Hz.

Posterior - The posterior is usually what we are attempting to obtain. It is the PDF on the parameters θ conditional-on (given) the data/measurements x. It's shape is determined by the product of the likelihood and the prior.

Evidence - The Bayesian evidence is the probability of obtaining the data x marginalised (or averaged) over all possible parameter values θ. It can be viewed as the normalisation factor for the product of the likelihood and prior such that the posterior is correctly normalised and it is important to note that it is *independent* of θ.

When you embark on a Bayesian analysis (GW or not) you need to know (at least) 3 things. You need to have a model for the signal - something that predicts the noise-free

form of the signal you expect in the data. This is usually something described by some parameters $\boldsymbol{\theta}$. You also need to know the allowed ranges of those parameters and more specifically their prior PDFs. Finally you need to have an understanding of your noise model - specifically how the noise behaves in the dataset you intend to analyse. Bayesian analyses can be applied to many scenarios but here we will focus on the additive noise case where the data $x(t)$ is the sum of the signal $s(t)$ and noise $n(t)$ such that

$$x(t) = s(t, \theta) + n(t). \tag{10}$$

We've written this in the continuous time domain but it is equally valid in the continuous or discrete time or frequency domains.

2.3. Gaussian likelihood

The signal model and its accompanying parameters and priors are defined by the problem you wish to address. In GW data analysis this can be a variety of possible astrophysical (or instrumental) transient or continuous waveforms. We will assume that these are understood and will generally not refer to specific models for the remainder of these notes. However, the noise model is something we should address.

Gaussian noise is not the only type of noise that could be assumed but it is used for the majority of cases. When we say that we will use Gaussian noise, what we mean is that the detector noise can (in general) be modelled as a multivariate Gaussian such that

$$p(\boldsymbol{n}) = \frac{1}{\sqrt{2\pi \det(C)}} \exp\left[-\frac{1}{2}\boldsymbol{n}C^{-1}\boldsymbol{n}^{\mathrm{T}}\right] \tag{11}$$

where the data is in this case a vector $\boldsymbol{n} = (n_0, n_1, n_2, \ldots, n_{N-1})$ and C is an $N \times N$ covariance matrix with $(\cdot)^T$ indicating a vector transpose.

We will show that GW data and (generally) data generated with non-flat spectra but with stationary Gaussian processes can be modelled as having diagonal covariance matrices when dealing with the data in the frequency domain. In this case the likelihood of the noise becomes

$$p(\boldsymbol{n}) = \prod_{j=0}^{N-1} \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left[-\frac{n_j^2}{2\sigma_j^2}\right] \tag{12}$$

where this should be treated as a likelihood since we are asking what is the probability of measuring data (in this case a vector of data $\boldsymbol{n} = (n_0, n_1, n_2, \ldots, n_{N-1})$) given the noise parameters. However, in this example there are no noise parameters of interest. Note that this doesn't have to be the case in general.

If we now consider (as done in Eq. 10) that our vector-form data $\boldsymbol{x}$ that may contain a signal such

$$\boldsymbol{x} = \boldsymbol{s}(\boldsymbol{\theta}) + \boldsymbol{n} \tag{13}$$

we can then ask what is the probability of obtaining a measurement $\boldsymbol{x}$ given a signal described by the parameters $\boldsymbol{\theta}$. This would be

$$\begin{aligned} p(\boldsymbol{x}|\boldsymbol{\theta}) &= p(\boldsymbol{n} = \boldsymbol{x} - \boldsymbol{s}(\boldsymbol{\theta})|\boldsymbol{\theta}) \\ &= \left(\prod_{j=0}^{N-1} \frac{1}{\sqrt{2\pi}\sigma_j}\right) \exp\left[-\frac{1}{2}\sum_{j=0}^{N-1}\frac{(x_j - s_j(\boldsymbol{\theta}))^2}{\sigma_j^2}\right] \\ &\propto \exp\left[-\frac{1}{2}\sum_{j=0}^{N-1}\frac{(x_j - s_j(\boldsymbol{\theta}))^2}{\sigma_j^2}\right] \end{aligned} \tag{14}$$

So in the simple case of uncorrelated Gaussian noise the likelihood is simply proportional to the exponentiated negative half χ-squared.

$$p(\boldsymbol{x}|\boldsymbol{\theta}) \propto e^{-\chi^2(\boldsymbol{\theta})/2} \tag{15}$$

We will see later why dropping the contants of proportionality is acceptable when considering parameter estimation. We can only drop them if they are independent of the parameters of interest $\boldsymbol{\theta}$ however we must not drop anything when considering model selection.

3. The power spectral density

The expectation value of a quantity x is given by

$$\langle x \rangle = \int x\, p(x) dx. \tag{16}$$

where $p(x)$ is the probability distribution function of x.

If $p(x)$ doesn't change over time then the a sequence of random variables drawn from the distribution is a **stationary random process**. In this case the expectation value can be approximated by the time average

$$\langle x \rangle = \lim_{T\to\infty} \frac{1}{T}\int_{-T/2}^{T/2} x(t) dt \tag{17}$$

Assume that we now have a process with zero mean $\langle x \rangle = 0$. The power in the timeseries is defined as

$$\langle x^2 \rangle = \lim_{T\to\infty} \frac{1}{T}\int_{-T/2}^{T/2} x^2(t) dt \tag{18}$$

Using Parseval's theorem

$$\int_{-\infty}^{\infty} |x(t)|^2 dt = \int_{-\infty}^{\infty} |\tilde{x}(f)|^2 df \tag{19}$$

and the fact that the orginal timeseries is real, we obtain

$$\begin{aligned} \langle x^2 \rangle &= \lim_{T\to\infty} \frac{2}{T}\int_{0}^{\infty} |\tilde{x}(f)|^2 df \\ &= \int_0^\infty S_x(f) df \end{aligned} \tag{20}$$

where $S_x(f)$ is the power spectral density (PSD) of the process $x(t)$.

Therefore the PSD is defined by

$$S_x(f) = \lim_{T\to\infty} \frac{2}{T} \left| \int_{-T/2}^{T/2} x(t) e^{-2\pi i f t} dt \right|^2 \tag{21}$$

The autocorrelation function is defined as

$$R(t) = \langle x(t) x(t+\tau) \rangle \tag{22}$$

and based on Eq. 21 we see that

$$\begin{aligned} S_x(f) &= \lim_{T\to\infty} \frac{2}{T} \int_{-T/2}^{T/2} x(t) e^{-2\pi i f t} dt \int_{-T/2}^{T/2} x(t') e^{2\pi i f t'} dt' \\ &= 2 \int_{-\infty}^{\infty} x(t) e^{-2\pi i f \tau} d\tau \left(\lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} x(t') x(t'+\tau) dt' \right) \\ &= 2 \int_{-\infty}^{\infty} R(\tau) e^{-2\pi i f \tau} d\tau \end{aligned} \tag{23}$$

where we have changes variables such that $t = t' + \tau$. So we see that the PSD can also be represented as the Fourier transform of the autocorrelation function.

Finally we express the PSD in terms of the Fourier components of the data. If we start with

$$\begin{aligned} \langle \tilde{x}^*(f') \tilde{x}(f) \rangle &= \left\langle \int_{-\infty}^{\infty} x(t') e^{2\pi i f' t'} dt' \int_{-\infty}^{\infty} x(t') e^{-2\pi i f t} dt \right\rangle \\ &= \left\langle \int_{-\infty}^{\infty} x(t') e^{2\pi i f' t'} dt' \int_{-\infty}^{\infty} x(t'+\tau) e^{-2\pi i f (t'+\tau)} d\tau \right\rangle \\ &= \int_{-\infty}^{\infty} e^{-2\pi i (f-f') t'} dt' \int_{-\infty}^{\infty} e^{-2\pi i f \tau} \langle x(t') x(t'+\tau) \rangle d\tau \end{aligned} \tag{24}$$

The second integral is Fourier transform of the autocorrelation function, and the first is the definition of the Dirac delta function $\delta(f - f')$. We therefore find (from Eq. 23) that

$$\langle \tilde{x}^*(f') \tilde{x}(f) \rangle = \frac{1}{2} S_x(f) \delta(f - f'). \tag{25}$$

This is an important result that indicates that the noise at different frequencies must be uncorrelated such that $\langle \tilde{x}^*(f) \tilde{x}(f) \rangle = 0$ for $f \neq f'$. The single-sided PSD is now defined in terms of the covariance of the frequency domain noise and since different frequencies are uncorrelated, a covariance matrix for noise at discrete frequency bins will be diagonal. We will need to use the covariance in our calculations in later sections.

An important practical expansion on the definition of the PSD relates to how things change when you actually sample discretely. In this case we refer back to our Fourier transform definitions (Eq. 3) and find that

$$S_k \approx \frac{2}{T} \langle |\tilde{x}_k|^2 \rangle. \tag{26}$$

Since in the frequency domain, for noise with zero mean, the definition of the variance of the noise in the k'th frequency bin is

$$\sigma_k^2 = \langle |\tilde{x}_k|^2 \rangle = \frac{T}{2} S_k \tag{27}$$

and hence the expression for the Gaussian likelihood given earlier in Eq. 14 can be ammended to include the PSD of the given detector such that

$$p(\boldsymbol{x}|\boldsymbol{\theta}) \propto \exp\left[-\frac{2}{T}\sum_{k=0}^{N_f-1}\frac{|\tilde{x}_k - \tilde{s}_k(\boldsymbol{\theta})|^2}{S_k}\right] \tag{28}$$

where we note that the sum is now over the postive frequency bins (of which there are $N_f = N/2$‡) and account for this with an extra factor of 2 inside the exponential.

If we now expand further we obtain

$$\begin{aligned} p(\boldsymbol{x}|\boldsymbol{\theta}) &\propto \exp\left[\Re\left(-\frac{2}{T}\sum_{k=0}^{N_f-1}\frac{(\tilde{x}_k - \tilde{s}_k(\boldsymbol{\theta}))^2}{S_k}\right)\right] \\ &\propto \exp\left[\Re\left(-\frac{2}{T}\sum_{k=0}^{N_f-1}\frac{\tilde{x}_k^2}{S_k} - \frac{2}{T}\sum_{k=0}^{N_f-1}\frac{\tilde{s}_k^2(\boldsymbol{\theta})}{S_k} + \frac{2}{T}\sum_{k=0}^{N_f-1}\frac{2\tilde{x}_k\tilde{s}_k(\boldsymbol{\theta})}{S_k}\right)\right] \\ &\propto \exp\left\{-\frac{1}{2}\left[(\boldsymbol{x},\boldsymbol{x}) + (\boldsymbol{s}(\boldsymbol{\theta}),\boldsymbol{s}(\boldsymbol{\theta})) - 2(\boldsymbol{x},\boldsymbol{s}(\boldsymbol{\theta}))\right]\right\} \end{aligned} \tag{29}$$

where we have taken the opportunity to define the noise weighted inner product

$$\begin{aligned} (\boldsymbol{a},\boldsymbol{b}) &= \frac{4}{T}\Re\sum_{k=0}^{N_f-1}\frac{\tilde{a}_k\tilde{b}_k^*}{S_k} \\ &\equiv 4\Re\int_0^\infty \frac{\tilde{a}(f)\tilde{b}^*(f)}{S(f)}df. \end{aligned} \tag{30}$$

One of the importnat things to note here is that the likelihood is computed in the frequency domain due to the nature of the noise. We can treat each frequency bin as statistically independent and hence multiply the probabilities for each bin. In the time domain you can only do this when the noise spectrum is flat, i.e., $S(f) = \text{const}$.

We use this opportunity to quickly define the (integrated) optimal signal-to-noise ratio (SNR) since it can be written very concisely as

$$\begin{aligned} \rho_{\text{opt}} &= (\boldsymbol{s},\boldsymbol{s}) \\ &= 4\int_0^\infty \frac{|\tilde{s}(f)|^2}{S(f)}df. \end{aligned} \tag{31}$$

‡ This is true for an even number of time bins but for an odd number the result is $N_f = \text{floor}(N/2)+1$

4. Optimal filtering

We start this section by refering back to Bayes theorem and consider the case where we have 2 competing hypotheses (or models). In this case we wish to ask what the ratio of probabilities of obtaining the data $\boldsymbol{x}$ given one model versus another. We can write this as

$$\mathcal{B}_{10} = \frac{p(\boldsymbol{x}|H_1)}{p(\boldsymbol{x}|H_0)} \tag{32}$$

As should be clear, this is a ratio of Bayesian evidences or marginal likelihoods. In general, as we will see later, in order to compute each of these quantities we would marginalise the product of the likelihood and prior over the model parameters for each model.

In this case we will only consider point hypotheses by which we mean that each hypothesis is defined by a set of fixed (known) parameters. The models in question are the signal model, defined by a (fixed) choice of parameter $\boldsymbol{\theta}$ and a noise model with no parameters (equivelent to the signal model but with the signal amplitude set to zero). Since there are no parameters to marginalise over the marginal likelihood becomes the likelihood and using Eq. 29 we find that

$$\begin{aligned}\Lambda &= \frac{\exp\left\{-\frac{1}{2}\left[(\boldsymbol{x},\boldsymbol{x}) + (\boldsymbol{s}(\boldsymbol{\theta}),\boldsymbol{s}(\boldsymbol{\theta})) - 2(\boldsymbol{x},\boldsymbol{s}(\boldsymbol{\theta}))\right]\right\}}{\exp\left\{-\frac{1}{2}(\boldsymbol{x},\boldsymbol{x})\right\}} \\ &= \exp\left\{-\frac{1}{2}\left[(\boldsymbol{s}(\boldsymbol{\theta}),\boldsymbol{s}(\boldsymbol{\theta})) - 2(\boldsymbol{x},\boldsymbol{s}(\boldsymbol{\theta}))\right]\right\}\end{aligned} \tag{33}$$

which we can rename as the *likelihood ratio* commonly referred to as Λ.

If one considers a case where you are looking for a specific signal in repeated sets of data then we notice that the variation in likelihood ratio depends only on the inner product $(\boldsymbol{x}, \boldsymbol{s}(\boldsymbol{\theta}))$ and hence we recognise this as the "optimal"§ statistic. Any choice of threshold on the likelihood ratio for accepting the signal hypothesis can be translated into a threshold on this quantity. We call this the *matched filter* since it is essentially a noise weighted innner product of the expected signal with the data.

4.1. Maximising over unknown parameters

Let us first consider the amplitude of the signal. For GWs this could relate directly to the unknown distance to a source. If the signal model is written as $s(t) = Ag(t)$ where A is the unknown amplitude we can write

$$\begin{aligned}\ln\Lambda &= (\boldsymbol{x},\boldsymbol{s}) - \frac{1}{2}(\boldsymbol{s},\boldsymbol{s}) \\ &= A(\boldsymbol{x},\boldsymbol{g}) - \frac{A^2}{2}(\boldsymbol{g},\boldsymbol{g})\end{aligned} \tag{34}$$

§ This is only really optimal in the point hypothesis case

Taking the derivative with respect to A and solving for A gives us the analytically maximised amplitude

$$A_{\rm max} = \frac{(\boldsymbol{x}, \boldsymbol{g})}{(\boldsymbol{g}, \boldsymbol{g})} \tag{35}$$

which when substituted back into the log likelihood ratio gives us the amplitude maximised equivelent

$$\ln \Lambda_{\rm max} = \frac{1}{2} \frac{(\boldsymbol{x}, \boldsymbol{g})^2}{(\boldsymbol{g}, \boldsymbol{g})}. \tag{36}$$

The important thing to note here is that we do not need to know the amplitude which for GW signals means that we do not need to know the distance.

Now consider that we also don't know the specific arrival time of the signal in our data. In this case we parameterise the signal model as

$$\boldsymbol{s} = Ag(t - t_0) \tag{37}$$

where we have a known signal waveform but with unknown amplitude and arrival time t_0. The Fourier transform of the signal in this case is

$$\begin{aligned} \tilde{s}(f) &= \int_{-\infty}^{\infty} Ag(t - t_0) e^{-2\pi i f t} dt \\ &= \int_{-\infty}^{\infty} Ag(t') e^{-2\pi i f (t' + t_0)} dt' \\ &= A\tilde{g}(f) e^{-2\pi i f t_0} \end{aligned} \tag{38}$$

It then follows that the inner product is modified via

$$\begin{aligned} (\boldsymbol{x}, \boldsymbol{s}) &= 4A\Re \int_0^{\infty} \frac{\tilde{x}(f)\tilde{g}^*(f)}{S(f)} e^{-2\pi i f t_0} df \\ &\approx \frac{4A}{T} \Re \sum_{k=0}^{N_f - 1} \frac{\tilde{x}_k \tilde{g}_k}{S_k} \exp\left\{2\pi i j_0 k / N\right\} \end{aligned} \tag{39}$$

where $j_0 = t_0/\Delta t$ is the discrete sample index representing the unknown start time. Notice that the inner product now becomes a inverse Fourier transform of the noise weighted product of $\boldsymbol{x}$ and $\boldsymbol{g}$. The output is now a function of time and can be computed efficiently using the fast fourier transform (FFT) since we are essentially performing a convolution in the time domain.

The expression computed for the amplitude maximised loglikelihood ratio in Eq. 36 still holds but now becomes a function of time. In this case we haven't actually analytically maximised over the arrival time, but we have shown that it can be computed efficiently for all (discrete) possible arrival times

If we now generalise our signal model further such that it is a linear combination of 2 known waveforms $p(t)$ and $q(t)$ such that

$$\begin{aligned} s(t) &= Ag(t) \\ &= A\left(p(t)\cos\phi + q(t)\sin\phi\right) \end{aligned} \tag{40}$$

where ϕ is an unknown phase angle and $p(t)$ and $q(t)$ are orthogonal in the sense that $(\boldsymbol{p},\boldsymbol{q}) = 0$. We also require that $(\boldsymbol{p},\boldsymbol{p}) = (\boldsymbol{q},\boldsymbol{q}) = (\boldsymbol{g},\boldsymbol{g})$‖. We therefore find that the log-likelihood ratio becomes

$$\begin{aligned} \ln\Lambda &= (\boldsymbol{x},\boldsymbol{s}(\phi)) - \frac{1}{2}(\boldsymbol{s}(\phi),\boldsymbol{s}(\phi)) \\ &= A(\boldsymbol{x},\boldsymbol{p})\cos\phi + A(\boldsymbol{x},\boldsymbol{q})\sin\phi - \frac{A^2}{2}(\boldsymbol{g}(\phi),\boldsymbol{g}(\phi)) \\ &= A\sqrt{(\boldsymbol{x},\boldsymbol{p})^2 + (\boldsymbol{x},\boldsymbol{q})^2}\cos(\Phi-\phi) - \frac{A^2}{2}(\boldsymbol{g}(\phi),\boldsymbol{g}(\phi)) \end{aligned} \tag{41}$$

where $\Phi = \arctan((\boldsymbol{x},\boldsymbol{q})/(\boldsymbol{x},\boldsymbol{p}))$. It is clear that this quantity when maximised over ϕ returns

$$\ln\Lambda = A\sqrt{(x,p)^2 + (x,q)^2} - \frac{A^2}{2}(p,p) \tag{42}$$

Using the same procedure as earlier when maximising over the amplitude A we can do the same again to obtain the phase and amplitude maximised log-likelihood ratio

$$\ln\Lambda_{\max} = \frac{(\boldsymbol{x},\boldsymbol{p})^2 + (\boldsymbol{x},\boldsymbol{q})^2}{2(\boldsymbol{p},\boldsymbol{p})} \tag{43}$$

Similarly, the procedure for accounting for the unknown initial time can also be applied. This gives us the same quantity but with inner products defined as in Eq. 39.

As a Bayesian I must note that the process of maximisation over unknown parameters is a *non*-optimal procedure. The correct process is to marginalise over those parameters whilst incorportating prior information. Doing this is often computationally costly and hence the minor reduction in optimality is acceptable since maximisation is very effciently computed.

5. Parameter Estimation

We started this document with Bayes theorem and ended Sec. 2 with its definition. Let's now focus on the issue of Bayesian parameter estimation, by which we mean obtaining multi-dimensional posterior PDFs on the parameters of a given model.

The reason we want these quantities is that they represent the updated knowledge of our model parameters based on the data we have access to. Our initial state of knowledge was represented by our prior PDF and by multiplying by the likelihood we

‖ These properties are satisfied in general by the different polarisation components of the GW.

increment that knowledge. After our experiment and data processsing we can then use our posterior as the prior for a future analysis and hence the incremental and sequential application of Bayes theorem leads to a consistent procedure for inference.

If you are only interested in the posterior then you can simplify Bayes theorem to be

$$p(\boldsymbol{\theta}|\boldsymbol{x}, I) \propto p(\boldsymbol{x}|\boldsymbol{\theta}, I)p(\boldsymbol{\theta}|I) \tag{44}$$

where we have introduced the parameter I which we will use to represent all other assumed relevant information. It is a rather cover-all term used to represent all of the things that have been assumed in the process.

Note that we have dropped the Bayesian evidence $p(\boldsymbol{x}|I)$ since it is independent of the parameters $\boldsymbol{\theta}$ and hence for a given dataset is just a constant. Once we've computed the LHS of Eq. 44 we can always normlalise it ourselves anyway. Note that it's a probability density function and hence should satisfy

$$\int d\boldsymbol{\theta} p(\boldsymbol{\theta}|\boldsymbol{x}, I) = 1. \tag{45}$$

So looking at Eq. 44 you might notice that for a given (fixed) dataset $\boldsymbol{x}$, our target is an N-dimensional function (where N is the number of parameters in $\boldsymbol{\theta}$). The RHS is the product of 2 such functions so why can't we just construct a grid of possible $\boldsymbol{\theta}$ values and just evaluate the LHS?

The answer is that we can. However, you would need to make the grid fine enough to capture the inherent function variation in the parameter space. Apriori you don't necessarily know what that it (although you can estimate it). Also, if you have a high dimensional space the number of physical grid points grows exponentially with dimensionality and very quickly. For even simple problems, the number of likelihood and prior evaluations then becomes prohibitively large. I will add that for 1-3D situations a brute force gridding it is often the simplest and quickest approach.

Imagine that you do have the ability to compute the full N-dimensional posterior, what do you actually do with it next? We need to distill the information down to meaningful quantities and so out first step might be to restate our posterior for a subset $\{j\}$ of our dimensions by marginalising (or averaging) over the others

$$p(\theta_{\{j\}}|\boldsymbol{x}, I) = \int d\boldsymbol{\theta}_{\{k\}\neq\{j\}} p(\boldsymbol{\theta}|\boldsymbol{x}, I) \tag{46}$$

where in this case the full N-dimensional joint posterior has already been correctly normalised. This process can be done over any number of parameters. The standard approach is to compute the 1-D and all of the 2-D (pair of parameters) marginalised posteriors.

When distilling our information even further we may want to compute confidence intervals on our 1-D distributions and the median, mean or mode of the posterior. The

mean is well defined as

$$\langle \theta_j \rangle = \int d\theta_j \theta_j p(\theta_j|\boldsymbol{x}, I) \tag{47}$$

but is not the characteristic of choice in most Bayesian applications. Instead the median or mode (location of the maximum) of the posterior is used

$$0.5 = \int_{\infty}^{\theta_j^{\rm med}} d\theta_j p(\theta_j|\boldsymbol{x}, I) \tag{48}$$

$$\theta_j^{\rm max} = \max\left[p^{-1}(\theta_j|\boldsymbol{x}, I)\right] \tag{49}$$

together with a set of confidence bounds. Note that the poserior inverse used above refers to the inverse of the function as opposed to its reciprocal.

Confidence bounds are defined at a specific level of enclosed probability (or confidence). For a given confidence level (usually 68%, 90%, 95%) the lower and upper bounds contain the desired integrated probability. There is no one specific set of bounds for a given confidence but there are 2 special cases that we can choose. They can be either symmetric or minimal non-symmetric where the former has bounds defined such that

$$\begin{aligned} \frac{1-C}{2} &= \int_{-\infty}^{\theta_j^{\rm low}} d\theta_j p(\theta_j|\boldsymbol{x}, I) \\ \frac{1+C}{2} &= \int_{\theta_j^{\rm upp}}^{\infty} d\theta_j p(\theta_j|\boldsymbol{x}, I). \end{aligned} \tag{50}$$

Here the same level of probability is contained below the median as it is above the median. The minimal non-symmetric case is the one in which the desired confidence/probability is enclosed within the smallest region. This can be acheived by solving these 2 equations

$$C = \int_{\theta_j^{\rm low}}^{\theta_j^{\rm upp}} d\theta_j p(\theta_j|\boldsymbol{x}, I) \tag{51}$$

$$p(\theta_j^{\rm low}|\boldsymbol{x}, I) = p(\theta_j^{\rm upp}|\boldsymbol{x}, I). \tag{52}$$

The symmetric interval will always contain within it the median whilst the minimal non-symmtric interval will always contain the mode.

The latter differs when one is dealing with a multi-modal distribution (one with multiple maxima). In this case there *may* be multiple intervals whose integrated enclosed probability equals the desired confidence.

5.1. Markov Chain Monte Carlo

As was mentioned in the previous section, a brute-force approach to sampling the parameter space in order to evaluate the posterior over its full extent is computationally infeasible in most realistic cases. Hence we require a smarter strategy.

A Markovian process is one in which the future state of a system is determined soley by it's current state and not determined by any of its past history. So the probability of the z_n'th state of a system would be defined by

$$p(z_n|z_{n-1}, z_{n-2}, \ldots, z_0) = p(z_n|z_{n-1}) \tag{53}$$

An example would be the location of a goldfish randomly swimming in a tank of water (assuming the aprocraphal 7-second memory of a goldfish) sampled every 7 seconds.

A Monte Carlo algorithm is one in which random sampling helps us to obtain numerical results. For example, to integrate over the 2D area of a unit circle one might randomly generate samples on the range $[-1, 1]$ for both x and y coordinates and record the fraction satisfying $x^2 + y^2 < 1$. Dividing that fraction by 4 gives as estimate of the π (the area). This is an example of what is known as rejection sampling.

Another use might be for computing integrals such as the expectation value of a parameter z where one could use the approximation

$$\langle z \rangle = \int z p(z) dz \approx \frac{1}{N} \sum_{j=1}^{N} z_j \tag{54}$$

where z_j are samples drawn randomly from $p(z)$.

A Markov chain Monte Carlo (MCMC) borrows from both of these principles and in general attempts to generate samples from a target distribution (defined on a multi-dimensional parameter space) via a form of random walk jumping from one possible location to the next following a set of probabilitic rules. The output is a list of (ultimately) independent random draws from the target distribution that one can then use for computing efficient integrations or (in our case) can be histogrammed to simply represent the probability distributions that we want.

The original algorithm for generating these samples is the Metropolis-Hastings (MH) algorithm (named after Nicholas Metropolis who first published the method in 1953 and and W.K. Hastings who extended it in 1970). To do this you must be able to evaluate a function $f(\boldsymbol{\theta})$ that is proportional to your desired probability distribution (in most cases the posterior $p(\boldsymbol{\theta}|\boldsymbol{x}, I)$). You are also required to supply a proposal distribution $q(\boldsymbol{\theta}'|\boldsymbol{\theta}_i)$ that defines how you randomly jump from one location to another in your parameter space. With these is hand, performing the following steps will generate samples from said distribution.

Why does this work? To answer that let's think about what's happening from the start. The initial guess state is just that "a guess" and has nothing to do with the final answer we hope to get. In fact, if our target distribution is quite localised somewhere in the parameter space then the value of $f(\boldsymbol{\theta}_0)$ is likely to be quite low. Hence, you can see that very quickly, new states will be accepted with higher values and our "chain" of parameter locations will move towards the high probability regions. This initial "searching" stage is known as the *burn-in* and does not have the desired property that the chain is in "thermal equalibrium". The rate at which the burn-in

Algorithm 1 Metropolis-Hastings algorithm

pick an initial state $\boldsymbol{\theta}_0$
set $i = 0$
while $i < N$ **do**
    randomly generate a candidate state $\boldsymbol{\theta}'$ from $q(\boldsymbol{\theta}'|\boldsymbol{\theta}_i)$
    calculate the acceptance probability $\alpha(\boldsymbol{\theta}', \boldsymbol{\theta}_i) = \min\left(1, \frac{f(\boldsymbol{\theta}')}{f(\boldsymbol{\theta}_i)} \frac{q(\boldsymbol{\theta}_i|\boldsymbol{\theta}')}{q(\boldsymbol{\theta}'|\boldsymbol{\theta}_i)}\right)$
    generate a uniform random number $r \in [0, 1]$
    if $r < \alpha$ **then**
        accept the new state and set $\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}'$
    else
        reject the new state and set $\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i$
    end if
    set $i = i + 1$
end while

stage reaches this state is dependent on the both the target function and the choice of proposal distribution since $q(\boldsymbol{\theta}', \boldsymbol{\theta}_i)$ determines the length scale of each jump.

Once the chain has found the region of high probability the behaviour of the chain changes. One simplistic way to think about this is to imagine the space is broken up into 2 regions and each region has the a fixed value of $f(\boldsymbol{\theta})$ such that for region 1 $f_1(\boldsymbol{\theta}) = 2$ and for region 2 $f_1(\boldsymbol{\theta}) = 3$. We also consider the standard case of a symmetric proposal distribution where $q(\boldsymbol{\theta}', \boldsymbol{\theta}_i) = q(\boldsymbol{\theta}_i, \boldsymbol{\theta}')$. Every time a jump is proposed from within 1 region to another location within the same region it always accepts that jump. Hence after many such jumps no specific interval inside that region has more samples than any other. Similarly, when a jump is proposed from region 1 to region 2, since it's always the case that $f_2 > f_1$ then that jump is also always accepted. However, when a jump from region 2 to region 1 is proposed it is only accepted with a probability of $f_1/f_2 = 2/3$. If left to run for many iterations you find that the chain will occupy region 1 a fraction of $(2/3)/(1 + 2/3) = 2/5$ of the time and region 2 for the corresponding $3/5$ of the time. This fraction mirrors the desired result that the samples are drawn from the target distribution.

The extension to a continuous rather than a discrete case is just adding more regions and taking the limit of each region tending to zero volume. The concept is still the same and returns samples drawn from the target distribution independent of the choice of the proposal distribution. Note that the normalisation between $f(\boldsymbol{\theta})$ and the probability distribution your really want to explore is irrelevent since any constant factors cancel out. You can see how this is ideal for the case of the Bayesian posterior where you have access to the product of likelihood and priors but not the Bayesian evidence.

Muliple modes - One thing that we've implicitly assumed is that the target distribution is mono-modal, i.e., it has a single concentration of probability within an enclosed volume of parameter space with only 1 maxima. In general this may not be

the case and annoyingly (for MCMC approaches) a multi-modal distribution can be problematic since the chain will most likely head off to the nearest maxima and get stuck there. This is especially problematic if that maxima is very small compared to the global maxima since then the chain gets stuck nowhere near the correct answer. If the distribution is made up of multiple equivelent sized maxima the chain will then sample from part of the full distribution but not all of it. One solution to this problem is to run multiple chains, each starting from a different initial random location. You can then compare the chains to identify which modes have been found and if confident that all have been identified samples can be resampled from the combined chains with weights proportional to the value of the target function $f(\boldsymbol{\theta})$ between samples from different chains.

Convergence - How can you tell that the chain has converged? The state of convergence on a single mode of the target distribution can be crudely identified by eye. The characteristics of such a state are that the statistical properties of the chain are stationary (so not varying with time). A more mathematical approach is to compute the autocorrelation of the chain and check for its stability over time and that it tends to zero for large τ (see Eq. 22).

Correlation - Due to the nature of how trial samples are drawn within the MH algorithm the samples themsleves are statistically correlated, i.e., neighbouring samples in the chain will not be well seperated since the proposal distribution for each jump was centred on the previous location. To account for this, when using the samples (for whatever purpose) take only every n'th sample where n is large enough to assume independence and can be obtained from calculation of the autocorrelation of the samples.

Marginalisation - Armed with a collection of samples drawn from the target distribution $p(\boldsymbol{\theta}|\boldsymbol{x}, I)$ how can we use them? We don't actually have access to the function, just samples drawn from it. The density of the samples are proportional to the target distribution. The most basic thing we can do is to histogram the data to obtain an approximation of the complete N-dimensional posterior PDF. The most practical data products we can derive however, are the low dimesnion (2 and 1-D) marginalised posterior distributions, which we described in Eq. 46 in terms of the joint distribution itslef. When we instead have samples this reduces to

$$p(\boldsymbol{\theta}_{\{j\}}|\boldsymbol{x}, I) \propto \rho\left(\{\boldsymbol{\theta}\}_{\{j\}}\right) \tag{55}$$

where $\{j\}$ are the set of parameter dimensions of interest, $\{\boldsymbol{\theta}\}_{\{j\}}$ are the complete set of samples on those parameter dimensions and $\rho()$ is a density operator of your choosing, the most basic being a simple histogram. The common alternative to the histogram is the kernel density estimate of which there are multiple types.

It should be stated that there are other random sampling algorithms and variants of the MH approach with which MCMC samples can be generated. The most commonly used in GW data analysis include parallel tempering, and more recently, Hamiltonian Monte-Carlo techniques.

6. Model selection

If we now turn our attention back to the main definition of Bayes Theorem but make a point of explicitly stating that our component probability functions are conditional on our particular choice of model M. One might think of this in terms of the example where our model is specifically that of General Relativity (GR). Hence, with this minor notation change we have

$$p(\boldsymbol{\theta}^{(M)}|\boldsymbol{x}, M, I) = \frac{p(\boldsymbol{x}|\boldsymbol{\theta}^{(M)}, M, I)p(\boldsymbol{\theta}^{(M)}|M, I)}{p(\boldsymbol{x}|M, I)}. \tag{56}$$

Note that we remain as general as possible such that the actual set of parameters $\boldsymbol{\theta}^{(M)}$ describing the model belong to that model.

As was touched upon in Sec. 4 we now wish to first ask a specific question about the model. One might think that the question to ask is "What is the probability of measuring my data given a particular model $p(\boldsymbol{x}|M, I)$?". However, such a question is next to meaningless since the probability of obtaining any point hypothesis (in this case a very specific set of data) is zero by definition. You may then think to instead ask the inverted question "What is the probability of my model given my measured data $p(M|\boldsymbol{x}, I)$?". Unfortunately we fall into the same trap here since there are technically an unlimited number of models. In both of these cases we can compute these quantities (as probability densities rather than probabilities) but the only meaning we can obtain from them is by taking their ratios between different models.

In the Bayesian language the previous argument leads us to define the Bayes factor between the 2 models a and b as

$$\mathcal{B}_{a,b} = \frac{p(\boldsymbol{x}|M_a, I)}{p(\boldsymbol{x}|M_b, I)} \tag{57}$$

$$= \frac{\int d\boldsymbol{\theta}^{(M_a)} p(\boldsymbol{x}|\boldsymbol{\theta}^{(M_a)}, M_a, I)p(\boldsymbol{\theta}^{(M_a)}|M_a, I)}{\int d\boldsymbol{\theta}^{(M_b)} p(\boldsymbol{x}|\boldsymbol{\theta}^{(M_b)}, M_b, I)p(\boldsymbol{\theta}^{(M_b)}|M_b, I)} \tag{58}$$

where we've written this out explicitly to show that the Bayesian evidences (or marginal likelihoods) are computed by integrating (marginalising) over potentially completely different parameter spaces or different dimensionality and governed by completely different priors. A more subtle difference between models might be that the models and parameters themsleves may be identical but prescribe different priors in each case.

An interesting and extremely powerful feature of the Bayes factor is that is naturally accounts for and penalises overly-complex models..e., it has an inbuilt Occam's razor. The concept of Occam's razor is encapsulated within the following statement attributed to William of Ockham (1287-1347) "Entities are not to be multiplied without neccesity" or in more modern day terms "When presented with competing hypotheses to solve a problem, one should select the solution with the fewest assumptions."

Looking at Eq. 57 (and taking an extreme case) it becomes clear why this happens. Imagine having 2 models that differ only in their prior space ranges and in both cases

the priors are uniform distributions over those ranges. Since the priors must be correctly normalised PDFs then the value of the prior will be inversely proportional to the product of ranges in each dimension. Hence $p(\boldsymbol{\theta}^{(M_a)}|M_a, I) > p(\boldsymbol{\theta}^{(M_b)}|M_b, I)$. The Bayes Factor is the ratio of the Bayesian evidences, which are the prior weighted averages of the likelihood function (which, if both sets of prior ranges encloses the bulk of the likelihood, will be identical between models). Therefore in this example we see that the Bayes Factor is effectively the inverse of the ratio between prior ranges. The more complex (or in this case, the less certain. more conservative model) becomes penalised.

The Bayes Factor is a quantity governed entirely by the data and one can view it in broad terms as "what the data is telling us about the relative merits of the models". However, it doesn't incorporate any information about how likely the models are of being true prior to including the data. You can imagine a situation where had a very poor or insensitive experiment for which the data was consistent with both a well established model and also a rather crazy model. You might conclude therefore that the Bayes Factor is ~ 1. This *does not* mean that the crazy model is as valid as the established one, it just means that the data was unhelpful in distinguishing them.

The ultimate Bayesian model selection question we want to ask is one on the models conditional on the data (rather than the other way round). Using Bayes theorem we can express the Bayesian evidence as

$$p(\boldsymbol{x}|M, I) = \frac{p(M|\boldsymbol{x}, I)p(p(\boldsymbol{x}|I)}{p(M|I)} \tag{59}$$

It then follows that we can define the Bayesian *Odds Ratio* as

$$\mathcal{O}_{a,b} = \frac{p(M_a|\boldsymbol{x}, I)}{p(M_b|\boldsymbol{x}, I)} \tag{60}$$

$$= \frac{p(\boldsymbol{x}|M_a, I)}{p(\boldsymbol{x}|M_b, I)} \frac{p(M_a|I)}{p(M_b|I)} \tag{61}$$

where we see that the Odds ratio is the ratio of the probability of models given the data (rather than the reverse statement defined by the Bayes Factor). This quantity is in fact the product of the Bayes Factor and what we call the *Prior Odds* which is the ratio of our prior belief in our models *before* we perform the new measurement. Going back to our previous toy model example with the "crazy" model, we can now incorporate our level of belief in that model realtive to the reasonable model through the prior odds. Despite the Bayes Factor being of order unity, by apriori assigning a high level of belief in the standard model over the crazy one we obtain an odds ratio that is consistent with our existing knowledge.

This latter case is an ideal example to show the general way in which Bayesian analysis is an iterative process whereby prior information is updated via measurement (likelihood). In this case a poor experiment has added nothing to our state of information and we are left with what we believed beforehand. Alternatively we can look at this as setting a high threshold for any experiment aiming to validate the crazy model. The Bayes Factor would have to be so high as to outweigh our prior belief.

So how would you actually assign a prior odds ratio? In an ideal case you simply take the value of the Odds Ratio computed from the most recent analysis. This response kind of avoids the point since it simply moves the question further and further backwards down the chain. Soneone at some point had to assign the original prior odds. The real answer is that things can get quite subjective since ultimately this is a statement of *statistical* belief. In practice most people chicken-out and assign an odds ratio of unity (essentially quoting the Bayes Factor).

6.1. Nested Sampling

To practically perform model selection and evaluate Eqns. 57,60 we clearly need to compute the Bayesian evidence $p(\boldsymbol{x}|M,I)$. We can write this explicitly as the following integral

$$\begin{aligned} p(\boldsymbol{x}|M,I) &= \int d\boldsymbol{\theta}^{(M)} p(\boldsymbol{x}|\boldsymbol{\theta}^{(M)}, M_a, I) p(\boldsymbol{\theta}^{(M)}|M,I) \\ &= \int d\boldsymbol{\theta} \mathcal{L}(\boldsymbol{\theta}) \pi(\boldsymbol{\theta}) \end{aligned} \tag{62}$$

where we try to follow the notation used in [2] for conciseness and consistency. In this case $\mathcal{L}(\boldsymbol{\theta})$ and $\pi(\boldsymbol{\theta})$ are the likelihood and prior functions.

As noted in Sec. 5.1 the standard Monte-Carlo methods for performing parameter estimation take advantage of the fact that we do not need to know the normalisation constant between our target distribution and the function $f(\boldsymbol{\theta})$ used to sample from it. For our applications this normalisation constant is the Bayesian evidence so we require another strategy.

The strategy we will cover here is *Nested Sampling* behind which is the general idea of turning a multi-dimensional integral into an easier to compute 1-D integral. We will start by defining the proportion of the prior space with likelihood value greater than the undefined quantity λ

$$\xi(\lambda) = \int\limits_{\mathcal{L}(\boldsymbol{\theta})>\lambda} d\boldsymbol{\theta}\pi(\boldsymbol{\theta}) = \int\limits_{\mathcal{L}(\boldsymbol{\theta})>\lambda} d\xi \tag{63}$$

where we have defined the element of prior mass $d\xi = d\boldsymbol{\theta}\pi(\boldsymbol{\theta})$. Note that $\xi(\lambda)$ is a fraction ranging from 0 to 1 with those boundaries corresponding to $\lambda \geq 0$ at $\xi = 1$ and $\lambda = \mathcal{L}_{\rm max}$ at $\xi = 0$.

We can then re-express the likelihood as

$$\mathcal{L}(\xi(\lambda)) = \lambda \tag{64}$$

Again, as noted in [2] the functions $\mathcal{L}(\boldsymbol{\theta})$ and $\mathcal{L}(\xi)$ are not the same function sin e one takes a vector argumnet and the other takes a scalar. However, they do return the same quantity, the likelihood, just as a function of different variables.

It now follows that the Bayesian evidence is simply the 1-D integral

$$p(\boldsymbol{x}|M,I) = \int_0^1 \mathcal{L}(\xi)d\xi \tag{65}$$

One can think of this as a transformation into the 1-D prior space in terms of considering $N-1$-dimensional shells of constant likelihood embedded in the full N-dimensional space. The parameter ξ controls which shell we are considering and $d\xi$ is the infinitessimal prior volume that this shell occupies.

Algorithmically we can compute this according to the following procedure

Algorithm 2 Nested Sampling algorithm

pick an initial set of $N_{\rm live}$ points $\boldsymbol{\theta}_0, \ldots, \boldsymbol{\theta}_{N_{\rm live}}$,
compute the likelihoods $\{\mathcal{L}_{\rm live}\} = \{\mathcal{L}(\boldsymbol{\theta}_j)\}$ for each point
initialise the evidence $Z = 0$
initialise $\xi_0 = 1$
set $i = 1$
while $i < N$ **do**
    set $\mathcal{L}_i = \min[\{\mathcal{L}_{\rm live}\}]$
    set $\xi_i = \exp\left(-i/N_{\rm live}\right)$
    set $w_i = \xi_{i-1} - \xi_i$
    set $Z = Z + L_i w_i$
    set $i = i + 1$
    replace the minimum likelihood live point with a new point drawn from the prior such that $\mathcal{L}(\boldsymbol{\theta}_{\rm new}) > \mathcal{L}_i$
end while

The algorithm itself makes use of an evolving set of points initially distributed uniformly over the prior volume. This is equivelent to uniformly distributing the points in the ξ space over its full range $[0,1]$. After each iteration the lowest likelihood point corresponding to ξ^* is removed and replaced with a higher value point, this time drawn from the range $[0,\xi^*]$. Given $N_{\rm live}$ samples drawn from a uniform distribution we can think of this as a shrinkage of the prior volume enclosing the live points. If we define the shrinkage fraction as $t = \xi_{\rm max}/\xi^*$ we can write that the probability that *all* samples shrinkage $\{t'_j\}$ are below a certain shrinkage t as

$$\begin{aligned} P(\{t'_j\} < t) &= P(t' < t)^{N_{\rm live}} \\ &= t^{N_{\rm live}} \end{aligned} \tag{66}$$

and since this is a probability, we can obtain the PDF by differentiating with respect to t to obtain

$$p(t) = N_{\rm live} t^{N_{\rm live}-1} \tag{67}$$

which has an expectation value

$$\begin{aligned}\langle t\rangle &= \int_0^1 dt\, tp(t) \\ &= N_{\rm live}\int_0^1 dt\, t^{N_{\rm live}-1} \\ &= \frac{N_{\rm live}}{N_{\rm live}+1}\end{aligned} \tag{68}$$

which for $N_{\rm live} \gg 1$ gives us

$$\ln\langle t\rangle \approx -\frac{1}{N_{\rm live}} \tag{69}$$

If on each iteration the live points (on average) occupy a fraction t of the previous volume then the fractional volume enclosed will evolve as

$$\begin{aligned}\xi_0 &= 1 \\ \xi_1 &= \exp\left(-\frac{1}{N_{\rm live}}\right) \\ \xi_2 &= \exp\left(-\frac{2}{N_{\rm live}}\right) \\ &\cdots \\ \xi_i &= \exp\left(-\frac{i}{N_{\rm live}}\right)\end{aligned} \tag{70}$$

Hence the weighting factor in the Nested Sampling algorithm being defined as the difference between these successive fractions

$$w_i = \xi_{i-1} - \xi_i \tag{71}$$

We leave the practical details of exactly how to sample from the prior volume efficiently at each iteration, how to choose $N_{\rm live}$ and how to define convergence of this integral to a later date.

7. Gravitational Wave example

7.1. Some handy tricks

We will make use of these tricks later on but it seems sensible to mention them at this point. These are in essence just simple probability rules for manipulating probabilistic expressions and can be divided into these categories.

(i) **Bayes theorem** - Often you find that it is better to have $p(a|b)$ where you might currently have $p(b|a)$ in an expression. If you are unable to compute the latter then it may serve you better to replace it using Bayes theorem (see Eq. 9).

(ii) **Expand joint distributions** - Joint distributions can be hard to define so if you have something like $p(a, b, c|d)$ you can expand it via

$$p(a, b, c|d) = p(a|b, c, d)p(b, c|d) = p(a|b, c, d)p(b|c, d)p(c|d) \tag{72}$$

or any other permutation that provides calculable probabilities.

(iii) **Dropping dependencies** - You may notice that you have a term $p(a|b, c)$ but you know that a only depends on c and not b. You can then show that

$$\begin{aligned} p(a, b|c) &= p(a|b, c)p(b|c) \\ p(a|b, c) &= \frac{p(a, b|c)}{p(b|c)} \\ &= \frac{p(a|c)p(b|c)}{p(b|c)} \\ &= p(a|c) \end{aligned} \tag{73}$$

where we have used the fact that the joint distribution of parameters can be written as $p(a, b|c) = p(a|c)p(b|c)$ if those parameter are independent of each other.

(iv) **Invoke additional parameters** - Sometimes you may have $p(a|b)$ in an expression but you are unable to compute it. However, you do know what $p(a|b, c)$ and $p(c|b)$ is so you can (using rule 2) replace it with

$$p(a|b) = \int p(a, c|b)dc = \int p(a|b, c)p(c|b)dc \tag{74}$$

(v) **Discrete variables** - You may have parameters in your model that are not continuous. In this case marginalisation reverts to a summation and PDFs on these parameters become probability mass function (PMF). A marginalisation over the binary parameter Y would look like

$$p(a|b) = \sum_{X=X_1,X_2} p(a|b, X)p(X|b) \tag{75}$$

7.2. Example with selection effects

One of the increasingly important aspects of GW analysis is dealing with selection effects. The primary example of this is that we place a reasonably hard threshold on the SNR of our detections. However, for compact binaries the SNR of an event is a function of the distance to the source, the mass of the source, its inclination, polarisation angle and sky position. Hence, when setting an SNR threshold that determined whether we will analyse an event, we are implicitly favouring regions of parameter space for our detections and therefore potentially biasing our parameter estimation.

If we wanted to compute the true posterior PDF on the parameters of a compact binary coalescence (CBC) event that we had only analysed becuase it passed an SNR threshold $\rho_{\rm th}$ then we must first ask the question, "Does that change our parameter

priors?". The answer is a definite *NO!*. The priors are what you believed before you performed the analysis and should have nothing to do with the effectiveness of your measurement. Instead we must alter our likelihood.

The likelihood is the probability of measuring the the data $\boldsymbol{x}$ given the parameters $\boldsymbol{\theta}$. By setting a threshold on SNR you are directly restricting the space of allowed data which in turn restricts the space of allowed parameters. However, it is the former that we must address within the likelihood. We do this by first asking what is the likelihood conditional on both the parameters $\boldsymbol{\theta}$ and the state of making a detection, which we denote as D. Using our allowed statistical manipulations we obtain

$$\begin{aligned} p(\boldsymbol{x}|\boldsymbol{\theta}, D, I) &= \frac{p(D|\boldsymbol{x}, \boldsymbol{\theta}, I)p(\boldsymbol{x}|\boldsymbol{\theta}, I)}{p(D|\boldsymbol{\theta}, I)} \\ &= \frac{p(\boldsymbol{x}|\boldsymbol{\theta}, I)}{p(D|\boldsymbol{\theta}, I)} \end{aligned} \tag{76}$$

where we have identified that $p(D|\boldsymbol{x}, \boldsymbol{\theta}, I) = 1$ for all cases that we end up actually looking at (since they are detections). However, we now have the standard likelihood in the numerator (independent of the state of detection) but in the denominator we have another parameter dependent function, the probability of detection given a set of parameters. This latter quantity can be evaluated as

$$p(D|\boldsymbol{\theta}, I) = \int_{\rho_{\rm th}}^{\infty} d\rho \, p(\rho|\boldsymbol{\theta}, I) \tag{77}$$

where $p(\rho|\boldsymbol{\theta}, I)$ is the probability distribution of the detected SNR given a particular set of parameters. From this we can see that for location in parameter space where the detection probability is low, since this is in the denominator, the likelihood is boosted. This seems counterintuitive but makes sense with further thought. The final posterior can now be written as

$$p(\boldsymbol{\theta}|\boldsymbol{x}, I) = \frac{p(\boldsymbol{x}|\boldsymbol{\theta}, I)}{p(D|\boldsymbol{\theta}, I)} \frac{p(\boldsymbol{\theta}|D, I)}{p(\boldsymbol{x}|D, I)} \tag{78}$$

Now imagine 2 locations in parameter space $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$, where the unmodified likelihoods $p(\boldsymbol{x}|\boldsymbol{\theta}_1, I) = p(\boldsymbol{x}|\boldsymbol{\theta}_2, I)$ are identical and so are the priors $p(\boldsymbol{\theta}_1|I) = p(\boldsymbol{\theta}_2, I))$. However, the probability of detection for location 1 is twice that of location 2 such that $p(D|\boldsymbol{\theta}_1, I)/p(D|\boldsymbol{\theta}_2, I) = 2$. The true values of the parameters should not be decided based on the ability of the experiment to detect them (the source knows nothing about our detector) so we account for this by boosting the likelihood of location 2.

7.3. Hierarchical Models

8. New methods: Machine learning

If there is time I will briefly sketch out the state of the machine learning applications in the GW field. These can be broken down into a number of overlapping sub-topics

1	1	1	2	2	3	3	3	2	2	1	1	1	0	0
2	3	4	5	6	7	7	7	6	5	4	3	2	1	1
4	6	8	11	12	14	14	14	12	11	8	6	4	3	1
6	9	12	16	18	20	21	20	18	16	12	9	6	4	2
7	11	14	18	21	23	24	23	21	18	14	11	7	5	3
6	9	12	16	18	20	21	20	18	16	12	9	6	4	2
4	6	8	11	12	14	14	14	12	11	8	6	4	3	1

Table A1. Grid of likelihood values used for the pracatical examples done with the Summer School students. The room has seats in a grid of 15×7 and the integrated area is ≈ 1000.

(i) Deep learning for astrophysical searches (compact binary coalescence, continuous waves, bursts)

(ii) Deep learning for parameter estimation (compact binary coalescence)

(iii) Neural networks for accelerating parameter estimation and model selection

(iv) Machine learning for identifying and classifying transient detector noise artefacts

(v) Machine learning for active feedback control in GW detectors

(vi) Machine learning for real-time cancelation of Newtonian noise

Appendix A. Practical exercise

In order to practically show how MCMC and Nested Sampling works we have set up a simulated likelihood function on a 2D grid of lecture theatre seats. This function is provided in Table. A1

References

[1] Jolien D. E. Creighton and Warren G. Anderson. *Gravitational-wave physics and astronomy: An introduction to theory, experiment and data analysis*. 2011.

[2] D.S. Sivia. *Data Analysis: A Bayesian Tutorial*. Data Analysis: A Bayesian Tutorial. Clarendon Press, 1996.

[3] GW Open Science Centre. `https://www.gw-openscience.org/`. Accessed: 2019-03-11.

[4] Matthew Pitkin's samplers demo. `http://mattpitkin.github.io/samplers-demo/pages/samplers-samplers-everywhere/`. Accessed: 209-03-11.