Computer Intensive Statistics: APTS 2023–24 Supporting Notes

Richard Everitt

2024-04-24

Contents

Chapter 1

Introduction

These notes were intended to supplement the *Computer Intensive Statistics* lectures and laboratory sessions rather than to replace or directly accompany them. As such, material is presented here in an order which is logical for reference purposes after the week and *not* precisely the order in which it was to be discussed during the week. There is much more information in these notes concerning some topics than would have been covered during the week itself. One of their main functions is to provide pointers to the relevant literature for anyone wanting to learn more about these topics.

Acknowledgement. These notes are based on a set developed by Adam Johansen, which can be traced back to a lecture course given by him and Ludger Evers in Bristol in 2007–08. Many of the better figures were originally prepared by Ludger Evers, and more recent contributions from previous APTS module leader Paul Jenkins. Any errors are my own and should be directed to richard.everitt@warwick.ac.uk.

1.1 Three Views of Sample Approximation

Many of the techniques described in these notes are simulation-based or otherwise make use of sample approximations of quantities. In the preliminary notes there is some discussion of the approximation of π, for example, by representing it in terms of an expectation that can be approximated using a sample average.

In general there are three increasingly abstract ways of viewing the justification of this type of approach. Thinking in these terms can be very helpful when trying to understand what these techniques are aiming to do and why we might expect them to work, and so it's worth thinking about this even before getting in to the details of particular algorithms.

1.1.1 Direct Approximation

This is a definition of Monte Carlo methods due to Halton (1970):

> Representing the solution of a problem as a parameter of a hypothetical population, and using a random sequence of numbers to construct a sample of the population, from which statistical estimates of the parameter can be obtained.

Recalling the approximation of π in the preliminary material, we constructed a simple random sample from a population described by a Bernoulli distribution with parameter $\pi/4$ and then used a simple estimate of the population parameter as an estimate of $\pi/4$.

Although in one sense this is the simplest view of a simulation-based approach to inference, it requires a specific construction for each problem we want to address.

1.1.2 Approximation of Integrals

The next level of indirection is to view Monte Carlo methods as algorithms for approximation of *integrals.* The quantity we wish to estimate is written as an expectation with respect to a probability distribution and a large sample from that population is then used to approximate that expectation; we can easily justify this via the (strong) law of large numbers and the central limit theorem.

That is, given $I = \int \varphi(x) f(x) dx$, we sample a collection, $X_1, \ldots, X_n$, of n independent random variables with distribution f and use the sample mean of $\varphi(X_i)$ as an approximation of I:

$$\hat{I}_n = \frac{1}{n} \sum_{i=1}^{n} \varphi(X_i).$$

The strong law of large numbers tells us that $\hat{I}_n \overset{a.s.}{\to} I$, and the central limit theorem (CLT) tells us that, provided that $\varphi(X)$ has finite variance, $\sqrt{n/\mathbb{V}\mathrm{ar}\left[\varphi(X)\right]}[\hat{I}_n - I] \overset{\mathcal{D}}{\to} Z$, where $X \sim f$ and Z is a standard normal random variable. The CLT tells us something about the *rate* at which the estimate converges. Notice that this rate is independent of the space in which the X_i live: this is the basis of the (slightly misleading) claim that the Monte Carlo method *beats the curse of dimensionality.* Although the rate is independent of dimension (Z is scaled by a quantity which decreases at a rate of $\sqrt{n}$, independent of dimension), the associated *constants* typically do depend on the dimension of the sampling space (in this simple case, this is because $\mathbb{V}\mathrm{ar}\left[\varphi(X)\right]$ is *typically* larger when X takes values in a larger space)...

In the case of the estimation of π, we can let $X_i = (X_i^x, X_i^y)$ with $X_i^x \overset{\text{iid}}{\sim} \mathsf{U}[-1, +1]$, $X_i^y \overset{\text{iid}}{\sim} \mathsf{U}[-1, +1]$, and X_i^x, X_i^y independent of one another. So we have $f(x, y) = \frac{1}{4}\mathbb{I}_{[-1,+1]}(x)\mathbb{I}_{[-1,+1]}(y)$, where $\mathbb{I}_A(x)$ denotes the indicator function on a set A evaluated at the point x, i.e. it takes the value 1 if $x \in A$ and 0 otherwise.

We consider the points which land within a disc of unit radius centred at the origin, $S_1 = \{(x, y) : x^2 + y^2 \leq 1\}$, and the proportion of points drawn from f which lie within S_1. The population value of this quantity (i.e. the probability that a particular point lies within the circle) is clearly the expectation of a function taking the value 1 within S_1 and 0 outside it: $\pi/4 = \int \mathbb{I}_{S_1}(x, y) f(x, y) dx$.

Note that this is just a particular case of the useful fact that the probability of any event A is equal to the expectation of an indicator function on that set.

1.1.3 Approximation of Distributions

The most abstract view of the Monte Carlo method, and indeed other approaches to simulation-based inference, is through the lens of distributional approximation. Rather than constructing an approximation of the quantity of interest directly, or of an integral representation of that quantity of interest, we could consider the method as providing an approximation of the distribution of interest itself.

If we are interested in some property of a probability distribution—a probability, an expectation, a quantile,..., then a natural approach would be to obtain an approximation of that distribution and to use the corresponding property of that approximation as an approximation of the quantity of interest. The natural simulation-based approach is to use the empirical distribution associated with a (large) sample from the distribution of interest. That is, we consider a discrete distribution which places mass $1/n$ on each of n points obtained by sampling from f, and then use this distribution as an approximation to f itself.

The (measure-theoretic) way of writing such an approximation is:

$$\hat{f}^n = \frac{1}{n} \sum_{i=1}^{n} \delta_{x_i},$$

where $x_1, \ldots, x_n$ are realisations of $X_1, \ldots, X_n \overset{\text{iid}}{\sim} \pi$ and δ_x denotes the probability distribution (i.e. measure) which places mass 1 at x. In the case in which π is a distribution over the real numbers we can think in terms of its distribution function and write

$$\hat{F}^n(x) = \sum_{i=1}^{n} \frac{1}{n} \mathbb{I}_{(-\infty,x]}(x_i),$$

which just tells us that we approximate $\mathbb{P}(X \leq x)$ with the proportion of the sampled values which lie below x.

In the case of the estimation of π, we saw in the previous section that we can represent π as an expectation with respect to $f(x, y) = \frac{1}{4}\mathbb{I}_{[-1,+1]}(x)\mathbb{I}_{[-1,+1]}(y)$. We immediately recover our approximation of π by taking the expectation under $\hat{f}^n$ rather than f.

This may seem like an unnecessarily complicated or abstract view of the approach, but it is very general and encompasses many ostensibly different approaches to Monte Carlo estimation.

1.2 The Usefulness of Sample Approximation

It may seem surprising at first, but it is often possible to obtain samples from distributions with respect to which it is not possible to compute expectations explicitly. *Sampling* is a different problem from *integrating*, and one may be soluble while the other is not. We can use the approximation provided by artificial samples in these cases to approximate quantities of interest that we might not be able to approximate adequately by other means.

In some other situations we will see, we may have to deal with settings in which we have access to a sample whose *distribution* we don't know; in such settings we can still use the approximation of the distribution provided by the sample itself.

1.3 Further Reading

A great many books have been written on the material discussed here, but it might be useful to identify some examples with particular strengths:

- An elementary self-contained introduction written from a similar perspective to these notes is provided by Voss (2013).
- A more in-depth study of Monte Carlo methods, particularly Markov chain Monte Carlo, is provided by Robert and Casella (2004).
- A slightly more recent collection of MCMC topics, including chapters on Hamiltonian Monte Carlo, is given by Brooks et al. (2011).
- A book with many examples from the natural sciences, which might be more approachable to those with a background in those sciences, is given by Liu (2001).

Where appropriate, references to both primary literature and good tutorials are provided throughout these notes.

Chapter 2

Simulation-Based Inference

2.1 Simulation

Much of what we will consider in this module involves sampling from distributions; simulating some generative process to obtain realisations of random variables. A natural question to ask is: how can we actually *do* this? What does it mean to sample from a distribution, and how can we actually implement such a procedure using a computer?

2.1.1 Pseudorandom Number Generation

Actually, strictly speaking, we can't generally obtain realisations of random variables of a specified distribution using standard hardware. We settle for sequences of numbers which have the same relevant statistical properties as random numbers and, more particularly, we will see that given a sequence of standard uniform (i.e. $\mathsf{U}[0,1]$) random variables, we can use some simple techniques to transform them to obtain random variables with other distributions of interest[1].

A *pseudorandom number generator* is a deterministic procedure which, when applied to some *internal state*, produces a value that can be used as a proxy for a realisation of a $\mathsf{U}[0,1]$ random variable and a new internal state. Such a procedure is initialised by the supply of some seed value and then applied iteratively to produce a sequence of realisations. Of course, these numbers are not in any meaningful sense *random*, indeed, to quote von Neumann:

> Anyone who considers artithmetical methods of reproducing random digits is, of course, in a state of sin. . . . there is no such thing as a random number—there are only methods of producing random numbers, and a strict arithmetic procedure is of course not such a method.

There are some very bad pseudorandom number generators in which there are very obvious patterns in the output and their use could seriously bias the conclusions of any statistical method based around simulation. So-called *linear congruential generators* were very popular for a time—but thankfully that time has largely passed, and unless you're involved with legacy code or hardware you're unlikely to encounter such things.

We don't have time to go into the details and intricacies of the PRNG in this module and as long as we're confident that we're using a PRNG which is *good enough for our purposes* then we needn't worry too much about its precise inner workings. Thankfully, a great deal of time and energy has gone into developing and testing PRNGs, including the Mersenne-twister-19937 (Matsumoto and Nishimura 1998) used by default in the current implementation of R (R Core Team 2013).

Parallel Computing and PRNGs

Parallel implementation of Monte Carlo algorithms requires access to parallel sources of random numbers. Of course, we really need to avoid simulating streams of random numbers in parallel with unintended relationships between the variables in the different streams. This is not at all trivial. Thankfully, some good solutions to the problem do exist—see Salmon et al. (2011) for an example.

[1] This material is included for completeness, but isn't actually covered in this module; a good summary was provided in the *Statistical Computing* module.

Quasi-Random Numbers

Quasi-random numbers, like pseudo-random numbers, are deterministic sequences of numbers which are intended to have, in an appropriate sense, similar statistical properties to pseudorandom numbers, but that is the limit of the similarities between these two things. Quasi-random number sequences (QRNS) are intended to have a particular *maximum discrepancy* property. See Morokoff and Caflisch (1995) for an introduction to the Quasi Monte Carlo technique based around such numbers; or Niederreiter (1992) for a book-length introduction.

Real Random Numbers

Although standard computers don't have direct access to any mechanism for generating truly random numbers; dedicated hardware devices that provide such a generator do exist. There exist sequences of numbers obtained by transformations of physical noise sources; see https://www.random.org/ for example. Surprisingly, the benefits of using such numbers—rather than those obtained from a *good* PRNG—do not necessarily outweigh the disadvantages (greater difficulty in replicating the results; difficulties associated with characterising the distribution of the input noise and hence the output random variables. . .). We won't discuss these sources any further in these notes.

Finite Precision

Although the focus of this section has been noting that the *real random numbers* employed in computational algorithms are usually not random, it is also worthwhile noting that they are also, typically, not really *real numbers* either. Most computations are performed using finite precision arithmetic. There is some discussion of simulation from the perspective of finite precision arithmetic as far back as Devroye (1986, chap. 15). Again, we won't generally concern ourself with this detail here; some of this was covered in the *Statistical Computing* module.

2.1.2 Transformation Methods

Having established that sources (of numbers which have similar properties to those of) random numbers uniformly distributed over the unit interval are available, we now turn our attention to turning random variables with such a distribution into random variables with other distributions. In principle, applying such transformations to realisations of $\mathsf{U}[0, 1]$ random variables will provide us with realisations of random variables of interest.

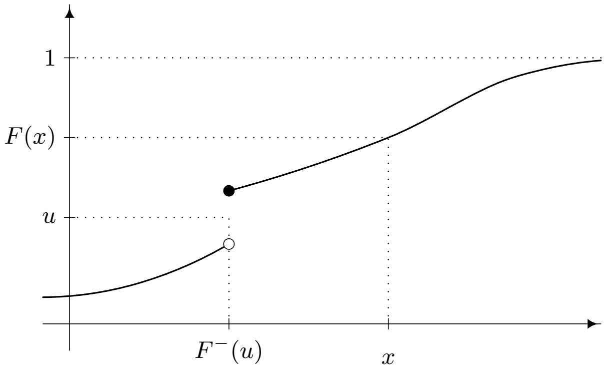

Figure 2.1: Illustration of the definition of the generalised inverse F^- of a CDF F.

One of the simplest methods of generating random samples from a distribution with some cumulative distribution function (CDF) $F(x) = \mathbb{P}(X \leq x)$ is based on the inverse of that CDF. Although the CDF is, by definition, an increasing function, it is not necessarily *strictly* increasing or continuous and so may not be invertible. To address this we define the *generalised inverse* $F^-(u) := \inf\{x : \ F(x) \geq u\}$. (In some fields this is also called the *quantile function*.) Figure 2.1 illustrates its definition. If F is continuous and strictly increasing, then $F^-(u) = F^{-1}(u)$. That is, the generalised inverse of a distribution function is a genuine generalisation of the inverse in that when F is invertible, F^- coincides with its inverse and when F is not invertible F^- is well defined nonetheless.

Theorem 2.1. *(Inversion Method).* Let $U \sim \mathsf{U}[0, 1]$ and F be a CDF. Then $F^-(U)$ has the CDF F.

Proof. It is easy to see (e.g. in Figure 2.1) that $F^-(u) \leq x$ is equivalent to $u \leq F(x)$. Thus for $U \sim \mathsf{U}[0, 1]$,

$$\mathbb{P}(F^-(U) \leq x) = \mathbb{P}(U \leq F(x)) = F(x),$$

thus F is the CDF of $X = F^{-}(U)$. □

Example 2.1. (Exponential Distribution). *The exponential distribution with rate $\lambda > 0$ has the CDF $F_\lambda(x) = 1 - \exp(-\lambda x)$ for $x \geq 0$. Thus $F_\lambda^{-}(u) = F_\lambda^{-1}(u) = -\log(1-u)/\lambda$, and we can generate random samples from $\mathsf{Exp}(\lambda)$ by applying the transformation $-\log(1-U)/\lambda$ to a uniform $\mathsf{U}[0,1]$ random variable U.*

As U and $1-U$, of course, have the same distribution we can instead use $-\log(U)/\lambda$ to save a subtraction operation.

When the generalised inverse of the CDF of a distribution of interest is available in closed form, the Inversion Method can be a very efficient tool for generating random numbers. However very few distributions possess a CDF whose (generalised) inverse can be evaluated efficiently. Take, for example, the Normal distribution, whose CDF is not even available in closed form.

The generalised inverse of the CDF is just one possible transformation. Might there be other transformations that yield samples from the desired distribution? An example of such a method is the Box–Muller method for generating Normal random variables. Such specialised methods can be very efficient, but typically come at the cost of considerable case-specific implementation effort (aside from the difficulties associated with devising such methods in the first place).

Example 2.2. (Box–Muller Method for Normal Simulation). *Using the transformation-of-density formula, one can show that $X_1, X_2 \overset{iid}{\sim} \mathsf{N}(0,1)$ iff their polar coordinates (R, θ) with*

$$X_1 = R \cdot \cos(\theta), \qquad X_2 = R \cdot \sin(\theta),$$

are independent, $\theta \sim \mathsf{U}[0, 2\pi]$, and $R^2 \sim \mathsf{Exp}(1/2)$ (Box and Muller 1958). Using $U_1, U_2 \overset{iid}{\sim} \mathsf{U}[0,1]$ and Example 2.1 we can generate R and θ by

$$R = \sqrt{-2\log(U_1)}, \qquad \theta = 2\pi U_2,$$

and thus

$$X_1 = \sqrt{-2\log(U_1)} \cdot \cos(2\pi U_2), \qquad X_2 = \sqrt{-2\log(U_1)} \cdot \sin(2\pi U_2)$$

are two independent realisations from a $\mathsf{N}(0,1)$ distribution.

The idea of transformation methods like the Inversion Method was to generate random samples from a distribution other than the target distribution and to transform them such that they come from the desired target distribution. Transformation methods such as those described here are typically extremely efficient but it can be difficult to find simple transformations to produce samples from complicated distributions, especially in multivariate settings.

Many ingenious transformation schemes have been devised for specific classes of distributions (see Devroye (1986) for a good summary of these), but there are many interesting distributions for which no such transformation scheme has been devised. In these cases we have to proceed differently. One option is to sample from a distribution other than that of interest, in which case we have to find other ways of correcting for the fact that we sample from the "wrong" distribution. One method for doing exactly this is described in the next section; at the end of this chapter we see an alternative way of using samples from 'proposal' distributions to approximate integrals with respect to another distribution (Section 2.4.2).

2.1.3 Rejection Sampling

The basic idea of rejection sampling is to sample from a *proposal distribution* (sometimes referred to as an *instrumental distribution*) and to reject samples that are "unlikely" under the target distribution in a principled way. Assume that we want to sample from a target distribution whose density f is known to us. The simple idea underlying rejection sampling (and several other Monte Carlo algorithms) is the following rather trivial identity:

$$f(x) = \int_0^{f(x)} 1\, du = \int_0^{\infty} \underbrace{\mathbb{I}_{[0,f(x)]}(u)}_{f(x,u):=} du.$$

Thus $f(x) = \int_0^\infty f(x,u)du$ can be interpreted as the marginal density of a uniform distribution on the *area under the density* $f(x)$, $\{(x,u) : 0 \leq u \leq f(x)\}$. This equivalence is very important in simulation, and has been referred to as the *fundamental theorem of simulation.* Figure 2.2 illustrates this idea.

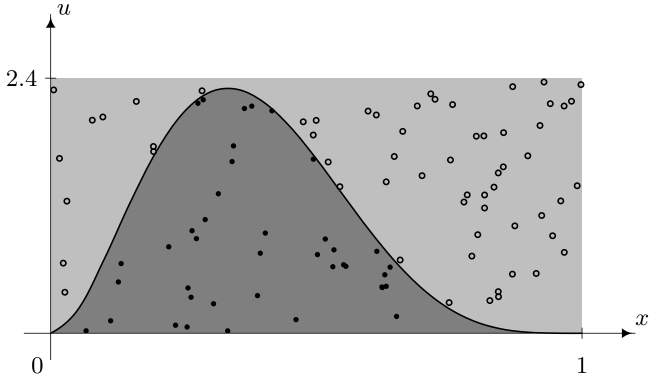

Figure 2.2: Sampling from the area under the curve (dark grey) corresponds to sampling from the Beta $(3, 5)$ density. We use a uniform distribution over the light grey rectangle as as proposal distribution. Empty circles denote rejected values, filled circles denote accepted values.

This suggests that we can generate a sample from f by sampling from the area under the curve—but it doesn't tell us how to sample uniformly from this area, which may be quite complicated (especially if we try to extend the idea to sampling from the distribution of a multivariate random variable).

Example 2.3. (Sampling from a Beta distribution). *The Beta (a, b) distribution $(a, b > 0)$ has the density*

$$f(x) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} x^{a-1}(1-x)^{b-1}, \qquad \textit{for } 0 < x < 1,$$

where $\Gamma(a) = \int_0^\infty t^{a-1} \exp(-t)\ dt$ is the Gamma function. For $a, b > 1$ the Beta (a, b) density is unimodal with mode $(a-1)/(a+b-2)$. Figure 2.2 shows the density of a Beta $(3, 5)$ distribution. It attains its maximum of $1680/729 \approx 2.305$ *at $x = 1/3$.*

Using the above identity we can draw from Beta $(3, 5)$ by drawing from a uniform distribution on the area under the density $\{(x, u) :\ 0 < u < f(x)\}$ (the area shaded in dark gray in Figure 2.2). In order to sample from the area under the density, we will use a similar trick to that used in the estimation of π in the preliminary material. We will sample from the light grey rectangle and and keep only the samples that fall in the area under the curve. Figure 2.2 illustrates this idea.

Mathematically speaking, we sample independently $X \sim \mathsf{U}[0, 1]$ and $U \sim \mathsf{U}[0, 2.4]$. We keep the pair (X, U) if $U < f(X)$, otherwise we reject it. The conditional probability that a pair (X, U) is kept if $X = x$ is

$$\mathbb{P}(U < f(X)|X = x) = \mathbb{P}(U < f(x)) = \frac{f(x)}{2.4}.$$

As X and U were drawn independently, we can rewrite our algorithm as: Draw X from $\mathsf{U}[0, 1]$ and accept X with probability $f(X)/2.4$, otherwise reject X.*

The method proposed in Example 2.3 is based on bounding the density of the Beta distribution by a box. Whilst this is a powerful idea, it cannot be applied directly to many other distributions, as many probability densities are either unbounded or have unbounded support (the whole real line, for example). However, we might be able to bound the density of $f(x)$ by $M \cdot g(x)$, where $g(x)$ is a density from which we can easily sample and M is a finite constant.

Algorithm 2.1 (Rejection sampling). *Given two densities f, g, with $f(x) \leq M \cdot g(x)$ for all x, we can generate a sample from f as follows.*

1. *Draw $X \sim g$.*
2. *Accept X as a sample from f with probability*

$$\frac{f(X)}{M \cdot g(X)},$$

otherwise go back to step 1.

Proof. Denote by E the set of all possible values X can take (for our purposes it can be assumed to be some subset of $\mathbb{R}^d$ but can, in principle, be a much more general space and f and g can be densities with respect to essentially any common reference measure). We have, for any (measurable) $\mathcal{X} \subseteq E$,

$$\mathbb{P}(X \in \mathcal{X} \text{ and is accepted}) = \int_{\mathcal{X}} g(x) \underbrace{\frac{f(x)}{M \cdot g(x)}}_{=\mathbb{P}(X \text{ is accepted}|X=x)} dx = \frac{\int_{\mathcal{X}} f(x)\, dx}{M}, \tag{2.1}$$

and thus

$$\mathbb{P}(X \text{ is accepted}) = \mathbb{P}(X \in E \text{ and is accepted}) = \frac{1}{M}, \tag{2.2}$$

yielding

$$\mathbb{P}(x \in \mathcal{X}|X \text{ is accepted}) = \frac{\mathbb{P}(X \in \mathcal{X} \text{ and is accepted})}{\mathbb{P}(X \text{ is accepted})} = \frac{\int_{\mathcal{X}} f(x)\, dx/M}{1/M} = \int_{\mathcal{X}} f(x)\, dx. \tag{2.3}$$

Thus the density of the values accepted by the algorithm is f. □

Remark. If we know f only up to a multiplicative constant, i.e. if we only know $\bar{f}(x)$, where $f(x) = C \cdot \bar{f}(x)$, we can carry out rejection sampling using

$$\frac{\bar{f}(X)}{M \cdot g(X)}$$

as the probability of rejecting X, provided $\bar{f}(x) \leq M \cdot g(x)$ for all x. Then by essentially the same argument as was used in (2.1)–(2.3) we have

$$\mathbb{P}(X \in \mathcal{X} \textit{ and is accepted}) = \int_{\mathcal{X}} g(x) \frac{\bar{f}(x)}{M \cdot g(x)}\, dx = \frac{\int_{\mathcal{X}} \bar{f}(x)\, dx}{M} = \frac{\int_{\mathcal{X}} f(x)\, dx}{C \cdot M},$$

$$\mathbb{P}(X \textit{ is accepted}) = 1/(C \cdot M),$$

and thus

$$\mathbb{P}(x \in \mathcal{X}|X \textit{ is accepted}) = \frac{\int_{\mathcal{X}} f(x)\, dx/(C \cdot M)}{1/(C \cdot M)} = \int_{\mathcal{X}} f(x)\, dx.$$

Example 2.4. (Rejection sampling from the $\mathsf{N}(0,1)$ distribution using a Cauchy proposal). *Assume we want to sample from the $\mathsf{N}(0,1)$ distribution with density

$$f(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$$

using a Cauchy distribution with density

$$g(x) = \frac{1}{\pi(1+x^2)}$$

as proposal distribution. Of course, there is not much point is using this method is practice: the Box–Muller method is more efficient. The smallest M we can choose such that $f(x) \leq Mg(x)$ is $M = \sqrt{2\pi} \cdot \exp(-1/2)$.

Figure 2.3 illustrates the results. As before, filled circles correspond to accepted values whereas open circles correspond to rejected values.*

Note that it is impossible to do rejection sampling the other way round: sampling from a Cauchy distribution using a $\mathsf{N}(0,1)$ distribution as proposal distribution. There is no $M \in \mathbb{R}$ such that

$$\frac{1}{\pi(1+x^2)} \leq M \cdot \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2}\right);$$

the Cauchy distribution has heavier tails than the Normal distribution. This illustrates a general principle that rejection sampling requires a proposal with heavier tails than the target distribution and the still more general principle that tail behaviour is often critical to the performance (or even correctness) of Monte Carlo algorithms.

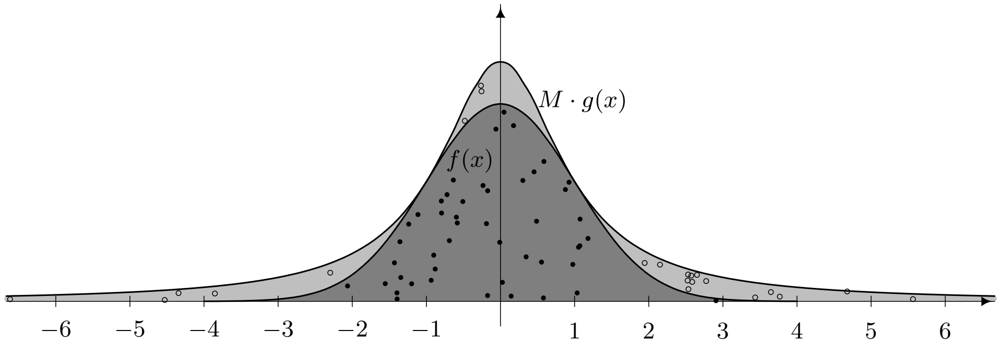

Figure 2.3: Sampling from the area under the density $f(x)$ (dark grey) corresponds to sampling from the $\mathsf{N}(0, 1)$ density. The proposal $g(x)$ is a $\mathsf{Cauchy}(0, 1)$.

2.2 Monte Carlo Testing

One of the simplest forms of simulation-based inference goes under the name of *Monte Carlo Testing* or, sometimes, *randomized testing*. The idea is appealingly simple and rather widely applicable.

Recall the basic idea of testing (or null hypothesis significance testing to be a little more precise). Given a *null hypothesis* about the data, compute some *test statistic* (i.e. a real valued summary function of the observed data) whose distribution is known under the null hypothesis and would be expected to deviate systematically from this under the alternative hypothesis. If a value of the test statistic shows a deviation which would be expected no more than $\alpha\%$ of the time when the null hypothesis is true (but which is expected to be more common under the alternative hypothesis), then one concludes that there is evidence which justifies *rejecting* the null hypothesis at the $\alpha\%$ level.

In principle this is reasonably straightforward, but there are often practical difficulties with following such a procedure. In particular, what if we do not know the distribution of the test statistic under the null hypothesis? The classical solution is to appeal to asymptotic theory for large samples to characterise the distribution of this statistic. This has two drawbacks: it is only *approximately* correct for finite samples, and it can be extremely difficult to do.

One simple solution which seems to have been first suggested formally by Barnard (1963) is to use simulation. This approach has taken a little while to gain popularity, despite some far-sighted early work (Besag and Diggle 1977 for example), perhaps in part because of limited computational resources and in part because of perceived difficulties with replication.

If T denotes the test statistic obtained from the actual data and $T_1, T_2, \ldots$ denote those obtained from repeated sampling under the null hypothesis, then, if the null hypothesis is true, $(T_1, \ldots, T_k, T)$ comprises a collection of $k + 1$ iid replicates of the test statistic. The probability that T is the largest of $(T_1, \ldots, T_k, T)$ is exactly $1/(k + 1)$ (by symmetry) and the probability that T is in the largest l of $(T_1, \ldots, T_k, T)$ is, similarly, $l/(k + 1)$.

By this reasoning, we can construct a hypothesis test at the 5% significance level by drawing $k = 19$ realisations of the test statistic and rejecting the null hypothesis if and only if T is greater than any of those synthetic replicates. This test is clearly *exact*: the probability of rejection if the null hypothesis is true is exactly as is specified. However, there is a loss of power as a result of the randomization, and there is no guarantee that two people presented with the same data will reach the same conclusion (if they both simulate *different* artificial replicates then one may reject and the other may not). However, for a "large enough" value of k these departures from the exact idealised test which this Monte Carlo procedure mimics are very small.

Although this idea might seem slightly arcane and removed from the other ideas we've been discussing in this section, it really is motivated by the same ideas. The empirical distribution of the artificial sample of test statistics converges to the true sampling distribution as the sample size becomes large, and we're then just using the empirical quantiles as a proxy for the quantiles of the true distribution. With a little bit of care, as seen here, this can be done in such a way that the false positive ("type I", if you insist) error probability is exactly that specified by the level of the test.

2.3 The Bootstrap

The *bootstrap* is based around a similar idea: if we want to characterise the distribution of an estimator then one option would be to simulate many replicates of it, and to use the resulting empirical distribution function as a proxy for the actual distribution function of the estimator. However, we don't typically know the distribution of the estimator[2].

If we knew that our dataset comprised realisations from some specific, known, distribution then it would be straightforward to simulate the distribution of a statistic by generating a large number of synthetic data sets and computing the statistic associated with each synthetic data set. In practice, however, we generally don't even know that distribution (after all, if we did there wouldn't be much statistics left to do. . .).

The idea behind the bootstrap is that the empirical distribution of a large (simple random) sample from some distribution is typically very close to the distribution itself (in various senses which we won't make precise here). In order to exploit this, we draw many replicates of the entire data set by sampling *with replacement from that data set* (i.e. by sampling from the associated empirical distribution) to obtain so-called bootstrap replicates. The statistic is then calculated for each of these replicates, and the resulting empirical distribution of the resulting statistic values, which we will term the *bootstrap distribution*, is used as a proxy for the true sampling distribution of that statistic.

If we're interested in some particular property of the sampling distribution of the test statistic (such as the variance of the statistic which might be useful in the construction of an approximate confidence interval), then we can simply: estimate that property of the estimator under the true sampling distribution with the property of that estimator under the *bootstrap* distribution, and use simulation get the latter.

A little more precisely, let $T = h(X_1, \ldots, X_n)$ denote a quantity calculated as a function of the original simple random sample of size n, $X_1, \ldots, X_n$ (i.e. T is a statistic calculated as a function h of some actually observed data). In order to approximate the sampling distribution of T we do the following:

```
Obtain Bootstrap Samples         For $i = 1, \ldots, B$:
                                   • Sample $X^\star_{i,1}, \ldots, X^\star_{i,n} \overset{\text{iid}}{\sim} \frac{1}{n} \sum_{j=1}^{n} \delta_{X_j}$
                                 End For
Compute Summaries                For $i = 1, \ldots, B$:
                                   • Set $T^\star_i = h(X^\star_{i,1}, \ldots, X^\star_{i,n})$.
                                 End For
Compute Empirical Distribution     • Set $f^\star_T = \frac{1}{B} \sum_{i=1}^{B} \delta_{T^\star_i}$.
                                   • Set $F^\star_T(t) = \frac{1}{B} \sum_{i=1}^{B} \mathbb{I}_{(-\infty,t]}(T^\star_i)$.
```

Compute Approximations of Interest | e.g. Sampling variance of T is $\mathbb{V}\mathrm{ar}_{f_T}[T]$; approximate with $\mathbb{V}\mathrm{ar}_{f^\star_T}[T]| = \frac{1}{B} \sum_{i=1}^{N} (T^\star_i)^2 - \left[\frac{1}{B}| \sum_{i=1}^{B} T^\star_i\right]^2$, which is none other than the sample variance of the statistic obtained from the bootstrap sample.

2.3.1 Bootstrap Confidence Intervals

One major use of the bootstrap is in the construction of (approximate) confidence intervals for statistics for which it might be difficult to construct exact confidence intervals.

2.3.1.1 Asymptotic Approach

We saw in the previous section that we can obtain approximations of the variance of an estimator using bootstrap techniques. The simplest method for constructing an approximate confidence interval using the bootstrap is to use such a variance estimate together with an assumption of approximate (or asymptotic) normality in order to arrive at an approximate (or asymptotic) confidence interval.

Taking this approach, we would arrive at an interval with endpoints of

$$T_n(X_1, \ldots, X_n) \pm z_{\alpha/2} \sqrt{\mathbb{V}\mathrm{ar}_{f_{T^\star}}[T_n]},$$

[2] Actually, an algorithm known as the *parametric bootstrap* does consider exactly this case, essentially resulting in an importance sampling estimate.

where z_α denotes the level α critical points of the standard normal distribution.

Although this approach may seem appealing in its simplicity, it can be expected to perform well only when the sampling distribution of the summary statistic is approximately normal. Imposing this additional assumption rather defeats the object of using bootstrap methods rather than employing simpler approximations directly.

2.3.1.2 Bootstrap Percentile Intervals

The next level of sophistication is to use the empirical distribution of the bootstrap realisations as a proxy for the sampling distribution of the statistic of interest. We arrive directly at an approximate confidence interval of the form $[t^\star_{1-\alpha/2}, t^\star_{\alpha/2}]$ where $t^\star_\alpha$ denotes the level α critical value of the bootstrap distribution.

Again this is a nice simple approach, but it does depend rather strongly on the quality of the approximation of the sampling distribution of T by the bootstrap distribution, and this is determined by the original sample size, amongst other factors.

2.3.1.3 Approximate Pivotal Quantity Approach

Another common approach to the problem has better asymptotic properties than that of the previous section and should generally be preferred in practice.

Assume that T is an estimator of some real population parameter, θ, and that the quantity $R = T - \theta$ is pivotal. (Recall that if X is drawn from a distribution parametrised by θ then $R = R(X, \theta)$ is *pivotal* for θ if its distribution is independent of θ.) As before, assume that we are able to obtain a large number of bootstrap replicates of T, denoted $T^\star_1, \ldots, T^\star_B$.

Let F_R denote the distribution function of R, so that: $F_R(r) := \mathbb{P}(R \leq r)$. It's a matter of straightforward algebra to establish that:

$$\begin{aligned}\mathbb{P}(L \leq \theta \leq U) = \mathbb{P}(L - T \leq \theta - T \leq U - T) &= \mathbb{P}(T - U \leq R \leq T - L)\\ &= F_R(T - L) - F_R(T - U).\end{aligned}$$

If we seek a confidence interval at level α it would be natural, therefore, to insist that $T - L = F_R^{-1}(1 - \alpha/2)$ and that $T - U = F_R^{-1}(\alpha/2)$ (assuming that F_R is invertible, of course).

Defining L and U in this way, we arrive at coverage of $1 - \alpha$ for the interval $[L, U]$ with $L = T - F_R^{-1}(1 - \alpha/2)$ and $U = T - F_R^{-1}(\alpha/2)$. Unfortunately, we can't use this interval directly because we don't know F_R and we certainly don't know F_R^{-1}.

This is where we invoke the usual bootstrap argument. If we are able to assume that the bootstrap replicates are to T *as* T is to θ then we can obtain a collection of bootstrap replicates of the pivotal quantity which we may define as: $R^\star_i = T^\star_i - T$. We can then go on to define the associated empirical distribution function and, more importantly, we can obtain the quantiles of this distribution. Letting $r^\star_\alpha$ denote the level α quantile of our bootstrap distribution, we obtain a *bootstrap pivotal confidence interval* of the form $[L^\star, U^\star]$ with:

$$L^\star = T - r^\star_{1-\alpha/2}, \quad U^\star = T + r^\star_{\alpha/2}.$$

Such confidence intervals can be shown to be asymptotically correct under fairly weak regularity conditions.

Remark. Although this approach may seem somewhat more complicated and less transparent than the methods discussed previously, it can be seen that the rate of convergence of bootstrap approximations of pivotal quantities can be $\mathcal{O}(1/n)$, in contrast to the $\mathcal{O}(1/\sqrt{n})$ obtained by appeals to asymptotic normality or the use of bootstrap approximations of non-pivotal quantities. See Young (1994) for a concise argument based around Edgeworth expansions for an illustration of statistical asymptotics in practice.

A more extensive theoretical consideration of these, and several other, approaches to the construction of bootstrap confidence intervals is provided by Hall (1986). A readable survey of developments in bootstrap methodology was provided by Davison, Hinkley, and Young (2003).

2.4 Monte Carlo Integration

Perhaps the most common application of simulation based inference is to the approximation of (intractable) integrals. We'll consider the approximation of expectations of the form $I_h = \mathbb{E}_f[h] = \int h(x)f(x)dx$, noting that more general integrals can always be written in this form by decomposing the integrand as the product of a probability density and whatever remains.

2.4.1 Simple / Perfect / Naïve Monte Carlo

The canonical approach to Monte Carlo estimation of I_h is to draw $X_1, \ldots, X_n \overset{\text{iid}}{\sim} f$ and to employ the estimator:

$$\hat{I}_h^n = \frac{1}{n}\sum_{i=1}^{n} h(X_i).$$

The strong law of large numbers tells us (provided that I_h exists) that $\lim_{n\to\infty} \hat{I}_h^n = I_h$. Furthermore, provided that $\mathbb{V}\mathrm{ar}_f[h] = \sigma^2 < \infty$, the Central Limit Theorem can be invoked to tell us that:

$$\sqrt{n}[\hat{I}_h^n - I] \xrightarrow{\mathcal{D}} \mathsf{N}\left(0, \sigma^2\right),$$

as $n \to \infty$, providing a rate of convergence.

If this were all there was to say about the topic then this could be a very short module. In fact, there are two reasons that we must go beyond this *perfect* Monte Carlo approach:

1. Often, if we wish to evaluate expectations with respect to some distribution π then we *can't* just simulate directly from π. Indeed, this is likely to be the case if we require sophisticated simulation-based methods to approximate the integral.

2. Even if we can sample from π, in some situations we obtain better estimates of I_h if we instead sample from another carefully selected distribution and correct for the discrepancy.

In Section 2.1 we saw some algorithms for sampling from some distributions; in Chapter 3 we will see another technique which will allow us to work with more challenging distributions. But first we turn our attention to a technique which can be used both to allow us to employ samples from distributions simpler than π to approximate I_h, and to provide better estimators than the perfect Monte Carlo approach if we are able to sample from a distribution tuned to both f and h.

2.4.2 Importance Sampling

In rejection sampling we compensated for the fact that we sampled from the proposal distribution $g(x)$ instead of $f(x)$ by rejecting some of the proposed values. *Importance sampling* is based on the idea of instead using *weights* to correct for the fact that we sample from the proposal distribution $g(x)$ instead of the target distribution $f(x)$.

Indeed, importance sampling is based on the elementary identity

$$\mathbb{P}(X \in \mathcal{X}) = \int_{\mathcal{X}} f(x)\, dx = \int_{\mathcal{X}} g(x) \underbrace{\frac{f(x)}{g(x)}}_{=:w(x)}\, dx = \int_{\mathcal{X}} g(x)w(x)\, dx \tag{2.4}$$

for all measurable $\mathcal{X} \subseteq E$ and $g(\cdot)$, such that $g(x) > 0$ for (almost) all x with $f(x) > 0$. We can generalise this identity by considering the expectation $\mathbb{E}_f[h(X)]$ of a measurable function h:

$$\begin{aligned}\mathbb{E}_f[h(X)] &= \int f(x)h(x)\, dx \\ &= \int g(x) \underbrace{\frac{f(x)}{g(x)}}_{=:w(x)} h(x)\, dx = \int g(x)w(x)h(x)\, dx = \mathbb{E}_g[w(X) \cdot h(X)],\end{aligned} \tag{2.5}$$

if $g(x) > 0$ for (almost) all x with $f(x) \cdot h(x) \neq 0$.

Assume we have a sample $X_1, \ldots, X_n \sim g$. Then, provided $\mathbb{E}_g\left[|w(X) \cdot h(X)|\right]$ exists,

$$\frac{1}{n}\sum_{i=1}^{n} w(X_i)h(X_i) \overset{a.s.}{\underset{n\to\infty}{\longrightarrow}} \mathbb{E}_g\left[w(X) \cdot h(X)\right],$$

and thus by (2.5)

$$\frac{1}{n}\sum_{i=1}^{n} w(X_i)h(X_i) \overset{a.s.}{\underset{n\to\infty}{\longrightarrow}} \mathbb{E}_f\left[h(X)\right].$$

In other words, we can estimate $\mu := \mathbb{E}_f\left[h(X)\right]$ by using

$$\tilde{\mu} := \frac{1}{n}\sum_{i=1}^{n} w(X_i)h(X_i).$$

Note that whilst $\mathbb{E}_g\left[w(X)\right] = \int_E \frac{f(x)}{g(x)} g(x)\, dx = \int_E f(x) = 1$, the weights $w_1(X), \ldots, w_n(X)$ do not necessarily sum up to n, so one might want to consider the *self-normalised* version

$$\hat{\mu} := \frac{1}{\sum_{i=1}^{n} w(X_i)}\sum_{i=1}^{n} w(X_i)h(X_i).$$

This gives rise to the following algorithm:

Algorithm 2.2 (Importance Sampling). *Choose g such that supp$(g) \supseteq$ supp$(f \cdot h)$.*

1. *For $i = 1, \ldots, n$:*
   i. *Generate $X_i \sim g$.*
   ii. *Set $w(X_i) = \frac{f(X_i)}{g(X_i)}$.*
2. *Return either*

$$\hat{\mu} = \frac{\sum_{i=1}^{n} w(X_i)h(X_i)}{\sum_{i=1}^{n} w(X_i)}$$

or

$$\tilde{\mu} = \frac{\sum_{i=1}^{n} w(W_i)h(X_i)}{n}.$$

The following theorem gives the bias and the variance of importance sampling.

Theorem 2.2. *(Bias and Variance of Importance Sampling).*

(a) $\mathbb{E}_g\left[\tilde{\mu}\right] = \mu$,

(b) $\mathbb{V}\text{ar}_g\left[\tilde{\mu}\right] = \dfrac{\mathbb{V}\text{ar}_g\left[w(X) \cdot h(X)\right]}{n}$,

(c) $\mathbb{E}_g\left[\hat{\mu}\right] = \mu + \dfrac{\mu\mathbb{V}\text{ar}_g\left[w(X)\right] - \mathbb{C}ov_g\left[\left[w(X), w(X) \cdot h(X)\right]\right]}{n} + \mathcal{O}(n^{-2})$,

(d) $\mathbb{V}\text{ar}_g\left[\hat{\mu}\right] = \dfrac{\mathbb{V}\text{ar}_g\left[w(X) \cdot h(X)\right] - 2\mu\mathbb{C}ov_g\left[\left[w(X), w(X) \cdot h(X)\right]\right] + \mu^2\mathbb{V}\text{ar}_g\left[w(X)\right]}{n} + \mathcal{O}(n^{-2})$.

Proof.

(a) $\mathbb{E}_g\left[\dfrac{1}{n}\displaystyle\sum_{i=1}^{n} w(X_i)h(X_i)\right] = \dfrac{1}{n}\displaystyle\sum_{i=1}^{n}\mathbb{E}_g\left[w(X_i)h(X_i)\right] = \mathbb{E}_f\left[h(X)\right]$.

(b) $\mathbb{V}\text{ar}_g\left[\dfrac{1}{n}\displaystyle\sum_{i=1}^{n} w(X_i)h(X_i)\right] = \dfrac{1}{n^2}\displaystyle\sum_{i=1}^{n}\mathbb{V}\text{ar}_g\left[w(X_i)h(X_i)\right] = \dfrac{\mathbb{V}\text{ar}_g\left[w(X)h(X)\right]}{n}$.

For (c) and (d) see Liu (2001, p35).

□

Note that the theorem implies that, contrary to $\tilde{\mu}$, the self-normalised estimator $\hat{\mu}$ is biased. The self-normalised estimator $\hat{\mu}$, however, might have a lower variance. In addition, it has another advantage: we only need to know the density up to a multiplicative constant, as is often the case in Bayesian modelling, for example. Assume $f(x) = C \cdot \bar{f}(x)$, then

$$\hat{\mu} = \frac{\sum_{i=1}^n w(X_i)h(X_i)}{\sum_{i=1}^n w(X_i)} = \frac{\sum_{i=1}^n \frac{f(X_i)}{g(X_i)}h(X_i)}{\sum_{i=1}^n \frac{f(X_i)}{g(X_i)}} = \frac{\sum_{i=1}^n \frac{C\cdot\bar{f}(X_i)}{g(X_i)}h(X_i)}{\sum_{i=1}^n \frac{C\cdot\bar{f}(X_i)}{g(X_i)}} = \frac{\sum_{i=1}^n \frac{\bar{f}(X_i)}{g(X_i)}h(X_i)}{\sum_{i=1}^n \frac{\bar{f}(X_i)}{g(X_i)}},$$

i.e. the self-normalised estimator $\hat{\mu}$ does not depend on the normalisation constant C. By a closely analogous argument, one can show that is also enough to know g only up to a multiplicative constant. On the other hand, as demonstrated by the proof of Theorem 2.2 it is a lot harder to analyse the theoretical properties of the self-normalised estimator $\hat{\mu}$.

Although the above equations (2.4) and (2.5) hold for every g with $\operatorname{supp}(g) \supseteq \operatorname{supp}(f \cdot h)$ and the importance sampling algorithm converges for a large choice of such g, one typically only considers choices of g that lead to *finite variance estimators*. The following two conditions are each sufficient (albeit rather restrictive; see Geweke (1989) for some other possibilities) to ensure that $\tilde{\mu}$ has finite variance:

- $f(x) \leq M \cdot g(x)$ and $\mathbb{V}\mathrm{ar}_f\left[h(X)\right] < \infty$.
- E is compact, f is bounded above on E, and g is bounded below on E.

So far we have only studied whether a g is an appropriate proposal distribution, i.e. whether the variance of the estimator $\tilde{\mu}$ (or $\hat{\mu}$) is finite. This leads to the question which proposal distribution is *optimal*, i.e. for which choice $\mathbb{V}\mathrm{ar}\left[\tilde{\mu}\right]$ is minimal. The following theorem, variants of which date back at least to Goertzel (1949), answers this question:

Theorem 2.3. *(Optimal proposal).* The proposal distribution g that minimises the variance of $\tilde{\mu}$ is

$$g^*(x) = \frac{|h(x)|f(x)}{\int_E |h(t)|f(t)\,dt}.$$

Proof. We have from Theorem 2.2 (b) that

$$\begin{aligned} n \cdot \mathbb{V}\mathrm{ar}_g\left[\tilde{\mu}\right] &= \mathbb{V}\mathrm{ar}_g\left[w(X)\cdot h(X)\right] \\ &= \mathbb{V}\mathrm{ar}_g\left[\frac{h(X)\cdot f(X)}{g(X)}\right] = \mathbb{E}_g\left[\left(\frac{h(X)\cdot f(X)}{g(X)}\right)^2\right] - \Bigg(\underbrace{\mathbb{E}_g\left[\frac{h(X)\cdot f(X)}{g(X)}\right]}_{=\mathbb{E}_g[\tilde{\mu}]=\mu}\Bigg)^2. \end{aligned}$$

The second term is independent of the choice of proposal distribution, thus we need minimise only $\mathbb{E}_g\left[\left(\frac{h(X)\cdot f(X)}{g(X)}\right)^2\right]$. Substituting $g^\star$ into this expression we obtain:

$$\begin{aligned} \mathbb{E}_{g^\star}\left[\left(\frac{h(X)\cdot f(X)}{g^\star(X)}\right)^2\right] &= \int_E \frac{h(x)^2\cdot f(x)^2}{g^\star(x)}\,dx = \left(\int_E \frac{h(x)^2\cdot f(x)^2}{|h(x)|f(x)}\,dx\right)\cdot\left(\int_E |h(t)|f(t)\,dt\right) \\ &= \left(\int_E |h(x)|f(x)\,dx\right)^2 \end{aligned}$$

On the other hand, we can apply Jensen's inequality to $\mathbb{E}_g\left[\left(\frac{h(X)\cdot f(X)}{g(X)}\right)^2\right]$, yielding

$$\mathbb{E}_g\left[\left(\frac{h(X)\cdot f(X)}{g(X)}\right)^2\right] \geq \left(\mathbb{E}_g\left[\frac{|h(X)|\cdot f(X)}{g(X)}\right]\right)^2 = \left(\int_E |h(x)|f(x)\,dx\right)^2$$

i.e. the estimator obtained by using an importance sampler employing proposal distribution $g^\star$ attains the minimal possible variance amongst the class of importance sampling estimators. □

An important corollary of Theorem 2.3 is that importance sampling can be *super-efficient*: when using the optimal $g^\star$ from Theorem 2.3 the variance of $\tilde{\mu}$ is less than the variance obtained when sampling directly from f:

$$\begin{aligned} n \cdot \mathbb{V}\mathrm{ar}_f \left[\frac{h(X_1) + \cdots + h(X_n)}{n} \right] &= \mathbb{E}_f \left[h(X)^2 \right] - \mu^2 \\ &\geq \left(\mathbb{E}_f \left[|h(X)| \right] \right)^2 - \mu^2 \\ &= \left(\int_E |h(x)| f(x) \, dx \right)^2 - \mu^2 = n \cdot \mathbb{V}\mathrm{ar}_{g^\star} \left[\tilde{\mu} \right] \end{aligned}$$

where the inequality follows from Jensen's inequality. Unless h is (almost surely) constant, the inequality is strict. There is an intuitive explanation to the super-efficiency of importance sampling. Using $g^\star$ instead of f causes us to focus on regions which balance both high probability density, f, and substantial values of the function, where $|h|$ is large, which contribute the most to the integral $\mathbb{E}_f[h(X)]$.

Theorem 2.3 is, however, a rather formal optimality result. When using $\tilde{\mu}$ we need to know the normalisation constant of $g^\star$, which if h is everywhere positive is exactly the integral we are attempting to approximate—and is likely to be equally difficult to evaluate even when that is not the case! Furthermore, we need to be able to draw samples from $g^\star$ efficiently. The practically important implication of Theorem 2.3 is that we should choose an instrumental distribution g whose shape is close to the one of $f \cdot |h|$.

Example 2.5. (Computing for $X \sim \mathsf{t}_3$). *Assume we want to compute $\mathbb{E}_f[|X|]$ for X from a t-distribution with 3 degrees of freedom (t_3) using a Monte Carlo method. Consider three different schemes.*

- *Sampling $X_1, \ldots, X_n$ directly from t_3 and estimating $\mathbb{E}_f[|X|]$ by*

$$\frac{1}{n} \sum_{i=1}^{n} |X_i|.$$

- *Alternatively we could use importance sampling using a t_1 (which is nothing other than a Cauchy distribution) as proposal distribution. The idea behind this choice is that the density $g_{\mathsf{t}_1}(x)$ of a t_1 distribution is closer to $f(x)|x|$, where $f(x)$ is the density of a t_3 distribution, as Figure 2.4 shows.*
- *Third, we will consider importance sampling using a $\mathsf{N}(0,1)$ distribution as proposal distribution.*

*Note that the third choice yields weights of infinite variance, as the proposal distribution ($\mathsf{N}(0,1)$) has lighter tails than the distribution we want to sample from (t_3). The right-hand panel of Figure 2.5 illustrates that this choice yields a very poor estimate of the integral $\int |x| f(x)\, dx$. Sampling directly from the t_3 distribution can be seen as importance sampling with all weights $w_i \equiv 1$; this choice clearly minimises the variance of the weights. However, minimizing the variance of the **weights** does not imply that this yields an estimate of the integral $\int |x| f(x)\, dx$ of minimal variance. Indeed, after 1500 iterations the empirical standard deviation (over 100 realisations) of the direct estimate is 0.0345, which is larger than the empirical standard deviation of $\tilde{\mu}$ when using a t_1 distribution as proposal distribution, which is 0.0182. This suggests that using a t_1 distribution as proposal distribution is super-efficient (see Figure 2.5, although we should always be careful when assuming that empirical standard deviations are a good approximation of the true standard deviation).*

*Figure 2.6 somewhat explains why the t_1 distribution is a far better choice than the $\mathsf{N}(0,1)$ distribution. As the $\mathsf{N}(0,1)$ distribution does not have heavy enough tails, the weight tends to infinity as $|x| \to +\infty$. Thus large $|x|$ can receive **very** large weights, causing the jumps of the estimate $\tilde{\mu}$ shown in Figure 2.5. The t_1 distribution has heavy enough tails, to ensure that the weights are small for large values of $|x|$, explaining the small variance of the estimate $\tilde{\mu}$ when using a t_1 distribution as proposal distribution.*

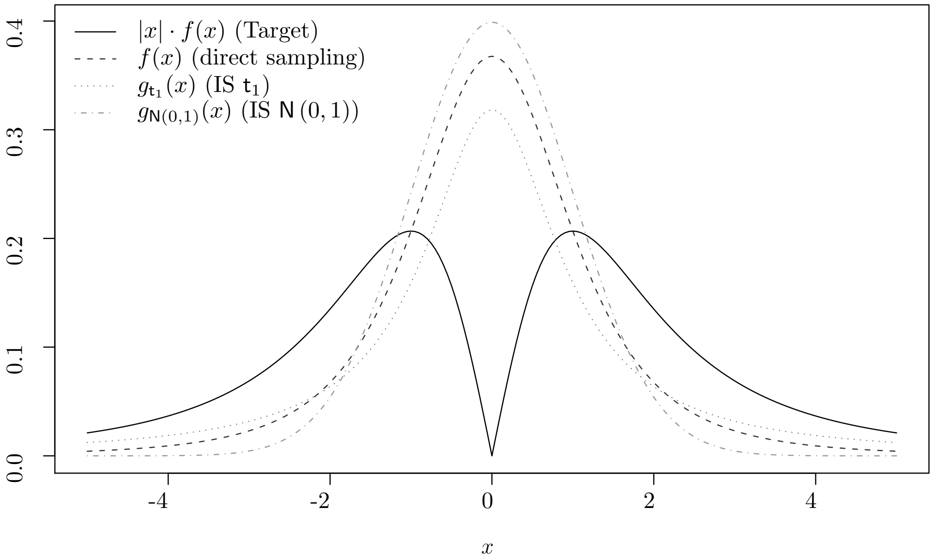

Figure 2.4: Illustration of different instrumental distributions for t_3 distribution.

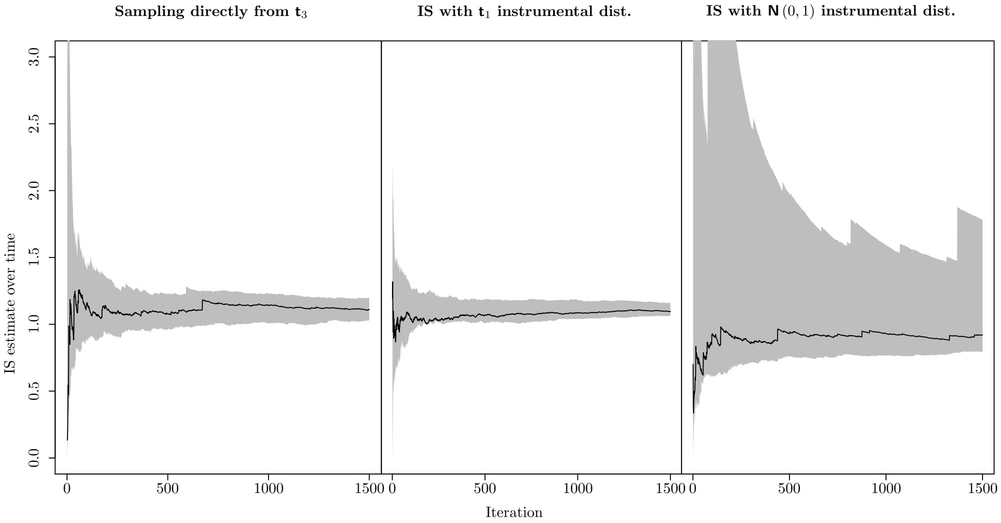

Figure 2.5: Estimates of $\mathbb{E}[|X|]$ for $X \sim \mathsf{t}_3$ obtained after 1 to 1500 iterations. The three panels correspond to the three different sampling schemes used. The areas shaded in grey correspond to the range of 100 replications.

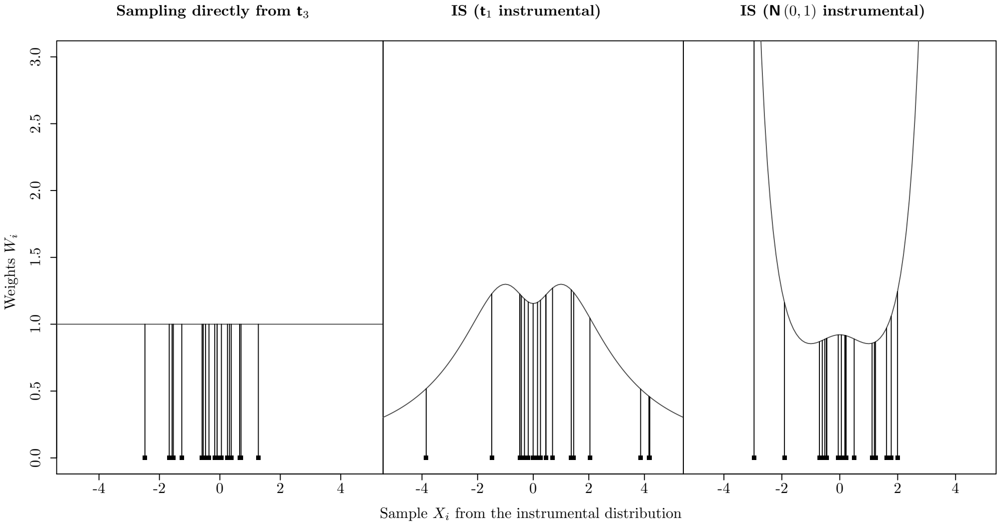

Figure 2.6: Weights W_i obtained for 20 realisations X_i from the different proposal distributions.

Chapter 3

Markov chain Monte Carlo

The focus of this chapter is on one class of simulation-based algorithms for approximating complex distributions without having to sample directly from those distributions. These methods have been tremendously successful in modern computational statistics.

Discrete time Markov processes on general state spaces, or Markov chains as we shall call such processes here, are described in some detail in the *Applied Stochastic Processes* module. Here, we investigate one particular use of these processes, as a mechanism for obtaining samples suitable for approximating complex distributions of interest.

Some definitions, background and useful results on Markov chains are provided in Appendix A.

3.1 The Basis of Markov chain Monte Carlo (MCMC)

In Chapter 2 we saw various methods for obtaining samples from distributions as well as some uses for such samples. The range of situations in which we like to make use of samples from distributions is, unfortunately, somewhat wider than the range of situations in which we can obtain such samples (easily, efficiently, or at all in some cases).

As the earlier *Applied Stochastic Processes* course will have provided a sound introduction to Markov chains for anyone able to attend it and not already familiar with them, we don't repeat that introduction here. Appendix A may provide a useful reference if these things are new to you (or, indeed, if it's some time since you've thought about them) and here we confine ourselves to a few essential definitions.

There are various conventions in the literature, but we will use the term *Markov chain* to refer to any discrete time Markov process, whatever may be its state space. (Sometimes the state space is also assumed to be discrete.) We'll assume here that the target distribution f is a continuous distribution over $E \subseteq \mathbb{R}^d$ for definiteness and for compact notation, but it should be realised that these techniques can be used in much greater generality.

For definiteness, we'll let K denote the density of the transition kernel of a Markov chain and we'll look for f-invariant Markov kernels, i.e. those for which:

$$\int_{x \in E} \int_{x' \in A} f(\boldsymbol{x}) K(\boldsymbol{x}, \boldsymbol{x}') d\boldsymbol{x} d\boldsymbol{x}' = \int_{x \in A} f(\boldsymbol{x}) d\boldsymbol{x}$$

for every measurable set A. We have assumed here that f admits a density and $K(x, \cdot)$ admits a density for any $\boldsymbol{x}$, and we can of course simplify this slightly and write the invariance condition as $\int_E f(\boldsymbol{x}) K(\boldsymbol{x}, \boldsymbol{x}') d\boldsymbol{x} = f(\boldsymbol{x}')$. Intuitively, an f-invariant Markov kernel is one which preserves the distribution f in the sense that if one samples $\boldsymbol{X} \sim f$ and then conditional upon $\boldsymbol{X}$ taking the value $\boldsymbol{x}$, sample $\boldsymbol{X}' \sim K(\boldsymbol{x}, \cdot)$ then, marginally, $\boldsymbol{X}' \sim f$.

If $\boldsymbol{X}_0, \boldsymbol{X}_1, \ldots$ is a Markov chain with some initial distribution μ_0 and f-invariant transition K then its clear that if at any time s that $\boldsymbol{X}_s \sim f$, then for every $t > s$ we have $\boldsymbol{X}_t \sim f$. That is, if the marginal distribution of the state of the Markov chain is f at *any* time then the marginal distribution of the state of the Markov chain at any later time is also f. This encourages us to consider using as an estimator of $I_h = \int h(\boldsymbol{x}) f(\boldsymbol{x}) d\boldsymbol{x}$ the sample path average of the function of interest:

$$\hat{I}_h^{MCMC} = \frac{1}{t} \sum_{i=1}^{t} h(\boldsymbol{X}_i)$$

which is of exactly the same form as the simple Monte Carlo estimator—except that it makes use of the trajectory of a Markov chain with f its invariant distribution instead of a collection of iid realisations from f itself.

However, there are two other issues which we need to consider before we could expect this estimator to have good properties:

- Is *any* $\boldsymbol{X}_t \sim f$? We know that if this is ever true it remains true for all subsequent times but we don't know that this situation is ever achieved.
- How does the dependence between consecutive states influence the estimator? Consider $X_1 \sim f$ and $X_i = X_{i-1}$ for all $i > 1$. This is a Markov chain whose states are all marginally distributed according to f, but we wouldn't expect $\hat{I}_h^{MCMC}$ to behave well if we used such a chain.

Next we'll briefly consider some problems which could arise and what behaviour we might need in order to have some confidence in an MCMC estimator before seeing some results which formally justify the approach.

3.1.1 Selected Properties and Potential Failure Modes

Not all f-invariant Markov kernels are suitable for use in Monte Carlo simulation. In addition to preserving the correct distribution, we need some notion of *mixing* or *forgetting*: we need the chain to move around the space in such a way that serial dependencies decay over time. The identity transition which sets $X_t = X_{t-1}$ with probability one is f-invariant for *every* f, but is of little use for MCMC purposes.

There are certain properties of Markov chains that are important because we can use them to ensure that various pathological things don't happen in the course of our simulations. We give a very brief summary here; see Appendix A for a slightly more formal presentation and some references.

Periodicity A Markov chain is *periodic* if the state space can be partitioned by a collection of more than one disjoint sets in such a way that the chain moves cyclically between elements of this partition. If such a partition exists then the number of elements in it is known as the *period* of the Markov chain; otherwise, the chain is termed *aperiodic*. In Markov chain Monte Carlo algorithms we generally require that the simulated chains are aperiodic (otherwise it's clear that the chain cannot ever forget in which element of the partition it started and, if initialised at some value, $\boldsymbol{x}_0$, will have disjoint support at time t and $t+1$ for all t and hence can never reach distribution f).

Reducibility A discrete space Markov chain is *reducible* if a chain cannot evolve from (almost) any point in the state space to any other; otherwise it is *irreducible*. In the case of chains defined on continuous spaces it is necessary to introduce a reference distribution, say ϕ, and to term the chain *ϕ-irreducible* if any set of positive probability under ϕ can be reached with positive probability from any starting point. In MCMC applications we require that the chains we use are f-irreducible; otherwise, the parts of the space that would be explored by the evolution of the chain would depend strongly on the starting value—and this would remain true even if the chain were run for an infinitely long time.

Transience Another significant type of undesirable behaviour is transience. Loosely speaking, a Markov chain is transient if it is expected to visit sets of positive probability under its invariant distribution only finitely often, even if permitted to run for infinite time. This means that in some sense the chain tends to drift off to infinity. In order for results like the law of large numbers to be adapted to the Markov chain setting, we require that sets of positive probability would in principle be visited arbitrarily often if the chain were to run for long enough. A ϕ-irreducible Markov chain is *recurrent* if the expected number of returns to any set of positive ϕ-probability is infinite. We'll focus on chains which have a stronger property. A ϕ-irreducible Markov chain is *Harris recurrent* if the probability that any set which has positive probability under ϕ is visited infinitely often by the chain (over an infinite time period) is one for *all* starting values.

3.1.2 A Few Theoretical Results

Formally, we justify Markov chain Monte Carlo by considering the asymptotic properties of the Markov chain (actually, in some situations it's possible to deal with finite sample properties but these are rather specialised settings). The following two results are two (amongst many similar theorems with subtly different conditions) which are to MCMC as the strong law of large numbers and the central limit theorem are to simple Monte Carlo.

Theorem 3.1. *(An Ergodic Theorem).* If $(\boldsymbol{X}_i)_{i\in\mathbb{N}}$ is an f-invariant, Harris recurrent Markov chain, then the following strong law of large numbers holds (convergence is with probability 1) for any integrable function $h : E \to \mathbb{R}$:

$$\lim_{t\to\infty} \frac{1}{t}\sum_{i=1}^{t} h(\boldsymbol{X}_i) \overset{a.s.}{=} \int h(x)f(x)dx.$$

Theorem 3.2. *(A Markov Chain Central Limit Theorem).* Under technical regularity conditions (see Jones (2004) for a summary of various combinations of conditions) it is possible to obtain a central limit theorem for the ergodic averages of a Harris recurrent, f-invariant Markov chain, and a function $h : E \to \mathbb{R}$ which has at least two finite moments (depending upon the combination of regularity conditions assumed, it may be necessary to have a finite moment of order $2 + \delta$ for some $\delta > 0$):

$$\lim_{t\to\infty} \sqrt{t}\left[\frac{1}{t}\sum_{i=1}^{t} h(\boldsymbol{X}_i) - \int h(x)f(x)dx\right] \overset{\mathcal{D}}{=} \mathsf{N}\left(0, \sigma^2(h)\right),$$

$$\sigma^2(h) = \mathbb{E}\left[(h(\boldsymbol{X}_1) - \bar{h})^2\right] + 2\sum_{k=2}^{\infty} \mathbb{E}\left[(h(\boldsymbol{X}_1) - \bar{h})(h(\boldsymbol{X}_k) - \bar{h})\right],$$

where $\bar{h} = \int h(x)\mu(x)dx$.

Although the variance is not a straightforward thing to calculate in practice (it's rarely possible) this expression *is* informative. It quantifies what we would expect intuitively, that the stronger the (positive[1]) relationship between successive elements of the chain the higher the variance of the resulting estimates. If $K(x, \cdot) = f(\cdot)$ so that we obtain a sequence of iid samples form the target distribution then we recover the variance of the simple Monte Carlo estimator.

3.2 Constructing MCMC Algorithms

So, having established that we can in principle employ Markov chains with f-invariant kernels to approximate expectations with respect to f, a natural question is *how can we construct an f-invariant Markov kernel*? Fortunately, there are some general methods which are very widely applicable.

3.2.1 Gibbs Samplers

We begin with a motivating example which shows that for realistic problems it may be possible to characterise and sample from the *full conditional distributions* associated with each variable separately; that is, the conditional distribution of any one variable given any particular value for all of the other variables, even when it is not possible to sample directly form their joint distribution. We'll use this to motivate a strategy of sampling iteratively from these full conditional distributions in order to obtain a realisation of a Markov chain, before going on to demonstrate that this approach falls within the MCMC framework described above.

Example 3.1. (Poisson change point model). *Assume the following Poisson model of two regimes for n random variables $Y_1, \ldots, Y_n$.*

$$Y_i \sim \mathsf{Poi}(\lambda_1) \quad \text{for } i = 1, \ldots, M,$$
$$Y_i \sim \mathsf{Poi}(\lambda_2) \quad \text{for } i = M+1, \ldots, n.$$

A conjugate prior distribution for λ_j is the $\mathsf{Gamma}\,(\alpha_j, \beta_j)$ distribution with density

$$f(\lambda_j) = \frac{1}{\Gamma(\alpha_j)}\lambda_j^{\alpha_j-1}\beta_j^{\alpha_j}\exp(-\beta_j\lambda_j).$$

The joint distribution of $Y_1, \ldots, Y_n$, λ_1, λ_2, and M is

$$f(Y_1, \ldots, Y_n, \lambda_1, \lambda_2, M) = \left(\prod_{i=1}^{M} \frac{\exp(-\lambda_1)\lambda_1^{Y_i}}{Y_i!}\right) \cdot \left(\prod_{i=M+1}^{n} \frac{\exp(-\lambda_2)\lambda_2^{Y_i}}{Y_i!}\right)$$
$$\cdot \frac{1}{\Gamma(\alpha_1)}\lambda_1^{\alpha_1-1}\beta_1^{\alpha_1}\exp(-\beta_1\lambda_1) \cdot \frac{1}{\Gamma(\alpha_2)}\lambda_2^{\alpha_2-1}\beta_2^{\alpha_2}\exp(-\beta_2\lambda_2).$$

[1] In principle, if we could arrange for negative correlation we could improve upon the independent case but in practice this isn't feasible.

If M is known, the posterior distribution of λ_1 given observations $Y_i = y_i$ for $1 \leq i \leq n$ has the density

$$f(\lambda_1 \mid y_1, \ldots, y_n, M) \propto \lambda_1^{\alpha_1 - 1 + \sum_{i=1}^{M} y_i} \exp(-(\beta_1 + M)\lambda_1),$$

so

$$\lambda_1 \mid y_1, \ldots y_n, M \sim \textsf{\textit{Gamma}}\left(\alpha_1 + \sum_{i=1}^{M} y_i, \beta_1 + M\right), \tag{3.1}$$

$$\lambda_2 \mid y_1, \ldots y_n, M \sim \textsf{\textit{Gamma}}\left(\alpha_2 + \sum_{i=M+1}^{n} y_i, \beta_2 + n - M\right). \tag{3.2}$$

Now assume that we do not know the change point M and that we assume a uniform prior on the set $\{1, \ldots, M-1\}$. It is easy to compute the distribution of M given the observations $y_1, \ldots y_n$, and λ_1 and λ_2. It is a discrete distribution with probability density function proportional to

$$p(M \mid \lambda_1, \lambda_2, y_1, \ldots y_n) \propto \lambda_1^{\sum_{i=1}^{M} y_i} \cdot \lambda_2^{\sum_{i=M+1}^{n} y_i} \cdot \exp((\lambda_2 - \lambda_1) \cdot M). \tag{3.3}$$

The conditional distributions in (3.1) *to* (3.3) *are all easy to sample from. It is however rather difficult to sample from the joint posterior of $(\lambda_1, \lambda_2, M)$.*

The example above suggests the strategy of alternately sampling from the (full) conditional distributions ((3.1) to (3.3) in the example). This tentative strategy however raises some questions.

- Is the joint distribution uniquely specified by the conditional distributions? We know that it is *not* determined by the collection of marginal distributions and so this is an important question (an algorithm which makes use of only these distributions could only be expected to provide information about the joint distribution if these conditionals do characterise that distribution).
- Sampling alternately from the conditional distributions yields a Markov chain: the newly proposed values only depend on the present values, not the past values. Will this approach yield a Markov chain with the correct invariant distribution? Will the Markov chain converge to the invariant distribution?

3.2.1.1 The Hammersley–Clifford Theorem

We begin by addressing the first of these questions via a rather elegant result known as the Hammersley–Clifford Theorem, although Hammersley and Clifford never actually published the result.

Definition 3.1. (Positivity condition). *A distribution with density $f(x_1, \ldots, x_p)$ and marginal densities $f_{X_i}(x_i)$ is said to satisfy the **positivity condition** if $f(x_1, \ldots, x_p) > 0$ for all $x_1, \ldots, x_p$ with $f_{X_i}(x_i) > 0$.*

The positivity condition thus implies that the support of the joint density f is the Cartesian product of the support of the marginals f_{X_i}.

Theorem 3.3. *(Hammersley–Clifford).* Let $(X_1, \ldots, X_p)$ satisfy the positivity condition and have joint density $f(x_1, \ldots, x_p)$. Then for all $(\xi_1, \ldots, \xi_p) \in \operatorname{supp}(f)$

$$f(x_1, \ldots, x_p) \propto \prod_{j=1}^{p} \frac{f_{X_j \mid X_{-j}}(x_j \mid x_1, \ldots, x_{j-1}, \xi_{j+1}, \ldots, \xi_p)}{f_{X_j \mid X_{-j}}(\xi_j \mid x_1, \ldots, x_{j-1}, \xi_{j+1}, \ldots, \xi_p)}.$$

Proof. We have

$$f(x_1, \ldots, x_{p-1}, x_p) = f_{X_p \mid X_{-p}}(x_p \mid x_1, \ldots, x_{p-1}) f(x_1, \ldots, x_{p-1}) \tag{3.4}$$

and by exactly the same argument

$$f(x_1, \ldots, x_{p-1}, \xi_p) = f_{X_p \mid X_{-p}}(\xi_p \mid x_1, \ldots, x_{p-1}) f(x_1, \ldots, x_{p-1}), \tag{3.5}$$

thus, using (3.4) and (3.5) in turn,

$$\begin{aligned}
f(x_1,\ldots,x_p) &= \underbrace{f(x_1,\ldots,x_{p-1})}_{=f(x_1,\ldots,,x_{p-1},\xi_p)/f_{X_p|X_{-p}}(\xi_p|x_1,\ldots,x_{p-1})} f_{X_p|X_{-p}}(x_p \mid x_1,\ldots,x_{p-1}) && (3.6)\\
&= f(x_1,\ldots,x_{p-1},\xi_p)\frac{f_{X_p|X_{-p}}(x_p \mid x_1,\ldots,x_{p-1})}{f_{X_p|X_{-p}}(\xi_p \mid x_1,\ldots,x_{p-1})} && (3.7)\\
&= \ldots && (3.8)\\
&= f(\xi_1,\ldots,\xi_p)\frac{f_{X_1|X_{-1}}(x_1 \mid \xi_2,\ldots,\xi_p)}{f_{X_1|X_{-1}}(\xi_1 \mid \xi_2,\ldots,\xi_p)}\cdots\frac{f_{X_p|X_{-p}}(x_p \mid x_1,\ldots,x_{p-1})}{f_{X_p|X_{-p}}(\xi_p \mid x_1,\ldots,x_{p-1})}. && (3.9)
\end{aligned}$$

The positivity condition guarantees that the conditional densities are non-zero. □

Note that the Hammersley–Clifford theorem does *not* guarantee the existence of a joint probability distribution for every choice of conditionals, as the following example shows. In Bayesian modelling such problems arise most often when using improper prior distributions.

Example 3.2. *Consider the following "model"*

$$\begin{aligned}
X_1 \mid X_2 &\sim \mathsf{Exp}(\lambda X_2),\\
X_2 \mid X_1 &\sim \mathsf{Exp}(\lambda X_1),
\end{aligned}$$

for which it would be easy to design a Gibbs sampler. Trying to apply the Hammersley–Clifford theorem, we obtain

$$f(x_1,x_2) \propto \frac{f_{X_1|X_2}(x_1 \mid \xi_2)\cdot f_{X_2|X_1}(x_2 \mid x_1)}{f_{X_1|X_2}(\xi_1 \mid \xi_2)\cdot f_{X_2|X_1}(\xi_2 \mid x_1)} = \frac{\lambda\xi_2\exp(-\lambda x_1\xi_2)\cdot\lambda x_1\exp(-\lambda x_1 x_2)}{\lambda\xi_2\exp(-\lambda\xi_1\xi_2)\cdot\lambda x_1\exp(-\lambda x_1\xi_2)} \propto \exp(-\lambda x_1 x_2)$$

The integral $\iint \exp(-\lambda x_1 x_2)\, dx_1\, dx_2$*, however, is not finite: there is no two-dimensional probability distribution with* $f(x_1,x_2)$ *as its density.*

3.2.1.2 Gibbs Sampling Algorithm

The generic Gibbs sampler is widely accepted as being first proposed by Geman and Geman (1984) and popularised within the general statistical community by Gelfand and Smith (1990). Denote $x_{-i} := (x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_p)$.

Algorithm 3.1 ((Systematic sweep) Gibbs sampler). *Starting with* $(X_1^{(0)},\ldots,X_p^{(0)})$ *iterate for* $t=1,2,\ldots$

- *1. Draw* $X_1^{(t)} \sim f_{X_1|X_{-1}}(\cdot \mid X_2^{(t-1)},\ldots,X_p^{(t-1)})$.
- $\vdots$
- *j. Draw* $X_j^{(t)} \sim f_{X_j|X_{-j}}(\cdot \mid X_1^{(t)},\ldots,X_{j-1}^{(t)},X_{j+1}^{(t-1)},\ldots,X_p^{(t-1)})$.
- $\vdots$
- *p. Draw* $X_p^{(t)} \sim f_{X_p|X_{-p}}(\cdot \mid X_1^{(t)},\ldots,X_{p-1}^{(t)})$.

Figure 3.1 illustrates the Gibbs sampler. The conditional distributions used in the Gibbs sampler are often referred to as *full conditionals* (being conditional upon everything except the variable being sampled at each step). Note that the Gibbs sampler is *not* reversible. Liu, Wong, and Kong (1995) proposed the following algorithm that yields a reversible chain.

Algorithm 3.2 (Random sweep Gibbs sampler). *Starting with* $(X_1^{(0)},\ldots,X_p^{(0)})$ *iterate or* $t=1,2,\ldots$

1. *Draw an index* j *from a distribution on* $\{1,\ldots,p\}$ *(e.g. uniform).*
2. *Draw* $X_j^{(t)} \sim f_{X_j|X_{-j}}(\cdot \mid X_1^{(t-1)},\ldots,X_{j-1}^{(t-1)},X_{j+1}^{(t-1)},\ldots,X_p^{(t-1)})$*, and set* $X_\iota^{(t)} := X_\iota^{(t-1)}$ *for all* $\iota \neq j$.

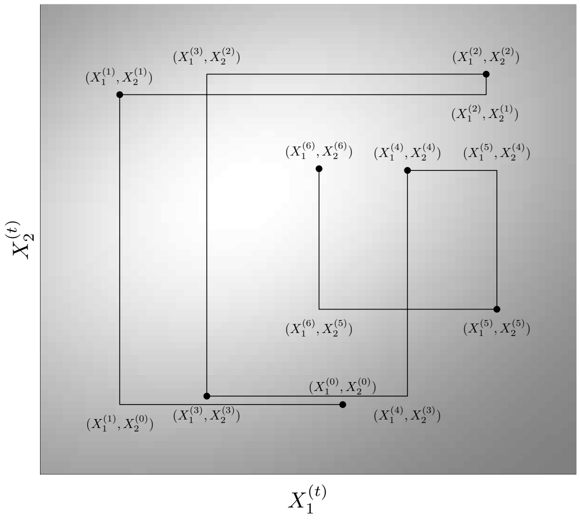

Figure 3.1: Illustration of the Gibbs sampler for a two-dimensional distribution.

3.2.1.3 Convergence of Gibbs Samplers

First we must establish whether that joint distribution $f(x_1, \ldots, x_p)$ is indeed the stationary distribution of the Markov chain generated by the Gibbs sampler. All the results in this section will be derived for the systematic scan Gibbs sampler (Algorithm 3.1). Very similar results hold for the random scan Gibbs sampler (Algorithm 3.2).

To proceed with such an analysis, we first have to determine the transition kernel corresponding to the Gibbs sampler.

Lemma 3.1. The transition kernel of the Gibbs sampler is

$$K(\boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t)}) = f_{X_1|X_{-1}}(x_1^{(t)} \mid x_2^{(t-1)}, \ldots, x_p^{(t-1)}) \cdot f_{X_2|X_{-2}}(x_2^{(t)} \mid x_1^{(t)}, x_3^{(t-1)}, \ldots, x_p^{(t-1)}) \cdot \cdots \cdot f_{X_p|X_{-p}}(x_p^{(t)} \mid x_1^{(t)}, \ldots, x_{p-1}^{(t)})$$

Proof. We have, for any (measurable $\mathcal{X}$):

$$\begin{aligned}
\mathbb{P}(\boldsymbol{x}^{(t)} \in \mathcal{X} \mid \boldsymbol{x}^{(t-1)} = \boldsymbol{x}^{(t-1)}) &= \int_{\mathcal{X}} f_{(\boldsymbol{x}^t|\boldsymbol{x}^{(t-1)})}(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)})\, d\boldsymbol{x}^{(t)} \\
&= \int_{\mathcal{X}} \underbrace{f_{X_1|X_{-1}}(x_1^{(t)} \mid x_2^{(t-1)}, \ldots, x_p^{(t-1)})}_{\text{corresponds to step 1. of the algorithm}} \cdot \underbrace{f_{X_2|X_{-2}}(x_2^{(t)} \mid x_1^{(t)}, x_3^{(t-1)}, \ldots, x_p^{(t-1)})}_{\text{corresponds to step 2. of the algorithm}} \cdot \ldots \\
&\quad \cdot \underbrace{f_{X_p|X_{-p}}(x_p^{(t)} \mid x_1^{(t)}, \ldots, x_{p-1}^{(t)})}_{\text{corresponds to step p. of the algorithm}}\, d\boldsymbol{x}^{(t)}.
\end{aligned}$$

□

Proposition 3.1. The joint distribution $f(x_1, \ldots, x_p)$ is indeed the invariant distribution of the Markov chain $(\boldsymbol{x}^{(0)}, \boldsymbol{x}^{(1)}, \ldots)$ generated by the Gibbs sampler.

Proof. Assume that $\boldsymbol{x}^{(t-1)} \sim f$, then

$$\begin{aligned}
\mathbb{P}(\boldsymbol{x}^{(t)} \in \mathcal{X}) &= \int_{\mathcal{X}} \int f(\boldsymbol{x}^{(t-1)}) K(\boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t)})\, d\boldsymbol{x}^{(t-1)}\, d\boldsymbol{x}^{(t)} \\
&= \int_{\mathcal{X}} \int \cdots \underbrace{\underbrace{\int f(x_1^{(t-1)}, \ldots, x_p^{(t-1)})\, dx_1^{(t-1)}}_{=f(x_2^{(t-1)},\ldots,x_p^{(t-1)})} f_{X_1|X_{-1}}(x_1^{(t)} \mid x_2^{(t-1)}, \ldots, x_p^{(t-1)})}_{=f(x_1^{(t)},x_2^{(t-1)},\ldots,x_p^{(t-1)})} \cdots \\
&\qquad \cdot f_{X_p|X_{-p}}(x_p^{(t)} \mid x_1^{(t)}, \ldots, x_{p-1}^{(t)}) dx_2^{(t-1)} \ldots dx_p^{(t-1)} d\boldsymbol{x}^{(t)} \\
&= \int_{\mathcal{X}} \int \cdots \underbrace{\underbrace{\int f(x_1^{(t)}, x_2^{(t-1)}, \ldots, x_p^{(t-1)})\, dx_2^{(t-1)}}_{=f(x_1^{(t)},x_3^{(t-1)},\ldots,x_p^{(t-1)})} f_{X_2|X_{-2}}(x_2^{(t)} \mid x_1^{(t)}, x_3^{(t-1)}, \ldots, x_p^{(t-1)})}_{=f(x_1^{(t)},x_2^{(t)},x_3^{(t-1)},\ldots,x_p^{(t-1)})} \cdots \\
&\qquad \cdot f_{X_p|X_{-p}}(x_p^{(t)} \mid x_1^{(t)}, \ldots, x_{p-1}^{(t)}) dx_3^{(t-1)} \ldots dx_p^{(t-1)} d\boldsymbol{x}^{(t)} \\
&= \ldots \\
&= \int_{\mathcal{X}} \underbrace{\underbrace{\int f(x_1^{(t)}, \ldots, x_{p-1}^{(t)}, x_p^{(t-1)})\, dx_p^{(t-1)}}_{=f(x_1^{(t)},\ldots,x_{p-1}^{(t)})} f_{X_p|X_{-p}}(x_p^{(t)} \mid x_1^{(t)}, \ldots, x_{p-1}^{(t)})}_{=f(x_1^{(t)},\ldots,x_p^{(t)})}\, d\boldsymbol{x}^{(t)} \\
&= \int_{\mathcal{X}} f(x_1^{(t)}, \ldots, x_p^{(t)})\, d\boldsymbol{x}^{(t)}
\end{aligned}$$

Thus f is the density of $\boldsymbol{x}^{(t)}$ (if $\boldsymbol{x}^{(t-1)} \sim f$). □

So far we have established that f is indeed the invariant distribution of the Gibbs sampler. Next, we have to analyse under which conditions the Markov chain generated by the Gibbs sampler will converge to f.

First of all we have to study under which conditions the resulting Markov chain is irreducible (really, we mean f-irreducible, of course, here and in the following we mean "irreducibility" with respect to the target distribution f). The following example shows that such irreducibility does not hold for every possible target distribution.

Example 3.3. (Reducible Gibbs sampler). *Consider Gibbs sampling from the uniform distribution on $C_1 \cup C_2$ with $C_1 := \{(x_1, x_2) : \|(x_1, x_2) - (1, 1)\| \leq 1\}$ and $C_2 := \{(x_1, x_2) : \|(x_1, x_2) - (-1, -1)\| \leq 1\}$, i.e.*

$$f(x_1, x_2) = \frac{1}{2\pi} \mathbb{I}_{C_1 \cup C_2}(x_1, x_2).$$

Figure 3.2 shows the density as well the first few samples obtained by starting a Gibbs sampler with $X_1^{(0)} < 0$ and $X_2^{(0)} < 0$.

It is easy to that when the Gibbs sampler is started in C_1 it will stay there and never reach C_2. The reason for this is that the conditional distribution $X_2 \mid X_1$ ($X_1 \mid X_2$) is for $X_1 < 0$ ($X_2 < 0$) entirely concentrated on C_1.

The following proposition gives a sufficient condition for irreducibility (and thus the recurrence) of the Markov chain generated by the Gibbs sampler. There are less strict conditions for the irreducibility and aperiodicity of the Markov chain generated by the Gibbs sampler (see e.g. Robert and Casella 2004 Lemma 10.11).

Proposition 3.2. If the joint distribution $f(x_1, \ldots, x_p)$ satisfies the positivity condition, the Gibbs sampler yields an f-irreducible, recurrent Markov chain.

Proof. Let $\mathcal{X} \subseteq \operatorname{supp}(f)$ be a set with $\int_{\mathcal{X}} f(x_1^{(t)}, \ldots, x_p^{(t)}) d(x_1^{(t)}, \ldots, x_p^{(t)}) > 0$. Then

$$\int_{\mathcal{X}} K(\boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t)}) d\boldsymbol{x}^{(t)} = \int_{\mathcal{X}} \underbrace{f_{X_1|X_{-1}}(x_1^{(t)} \mid x_2^{(t-1)}, \ldots, x_p^{(t-1)})}_{>0 \text{ (on a set of non-zero measure)}} \cdots \underbrace{f_{X_p|X_{-p}}(x_p^{(t)} \mid x_1^{(t)}, \ldots, x_{p-1}^{(t)})}_{>0 \text{ (on a set of non-zero measure)}} d\boldsymbol{x}^{(t)} > 0.$$

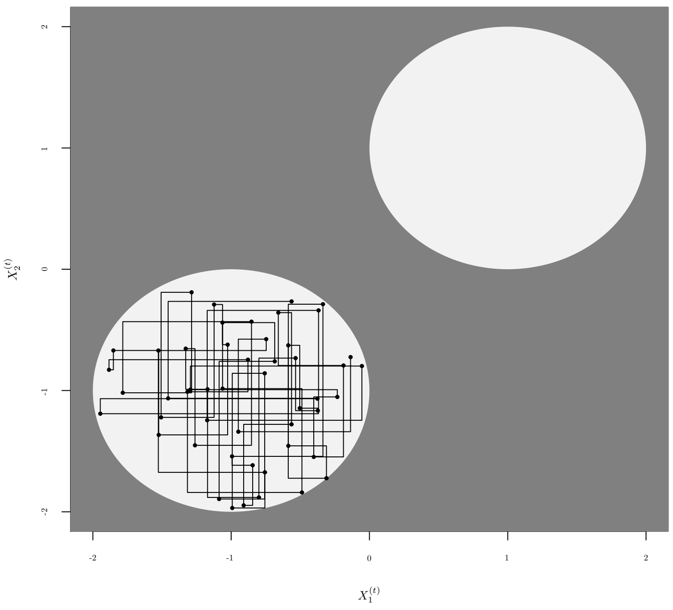

Figure 3.2: Illustration of a Gibbs sampler failing to sample from a distribution with unconnected support (uniform distribution on $\{(x_1, x_2) : \|(x_1, x_2) - (1, 1)\| \leq 1|\} \cup \{(x_1, x_2) : \|(x_1, x_2) - (-1, -1)\| \leq 1|\}$).

Thus the Markov Chain $(\boldsymbol{x}^{(t)})_t$ is strongly f-irreducible. As f is the invariant distribution of the Markov chain, it is recurrent. □

If the transition kernel is absolutely continuous with respect to the dominating measure, then recurrence even implies Harris recurrence (see e.g. Robert and Casella 2004 Lemma 10.9).

Now we have established all the necessary ingredients to state an ergodic theorem for the Gibbs sampler, which is a direct consequence of Theorems A.1 and A.2.

Theorem 3.4. If the Markov chain generated by the Gibbs sampler is irreducible and recurrent (which is e.g. the case when the positivity condition holds), then for any integrable function $h : E \to \mathbb{R}$:

$$\lim_{n\to\infty} \frac{1}{n} \sum_{t=1}^{n} h(\boldsymbol{x}^{(t)}) \to \mathbb{E}_f \left[h(\boldsymbol{X})\right]$$

for almost every starting value $\boldsymbol{x}^{(0)}$. If the chain is Harris recurrent, then the above result holds for every starting value $\boldsymbol{x}^{(0)}$.

Theorem 3.4 guarantees that we can approximate expectations $\mathbb{E}_f\left[h(\boldsymbol{x})\right]$ by their empirical counterparts using *a single* Markov chain.

Example 3.4. *Assume that we want to use a Gibbs sampler to estimate* $\mathbb{P}(X_1 \geq 0, X_2 \geq 0)$ *for a* $\mathsf{N}\left(\left(\begin{array}{c}\mu_1\\ \mu_2\end{array}\right), \left(\begin{array}{cc}\sigma_1^2 & \sigma_{12}\\ \sigma_{12} & \sigma_2^2\end{array}\right)\right)$ *distribution. The marginal distributions are*

$$X_1 \sim \mathsf{N}\left(\mu_1, \sigma_1^2\right) \qquad \textit{and} \qquad X_2 \sim \mathsf{N}\left(\mu_2, \sigma_2^2\right).$$

In order to construct a Gibbs sampler, we need the conditional distributions $X_1 \mid X_2 = x_2$ *and* $X_2 \mid X_1 = x_1$*. We* *have*[2]

$$f(x_1,x_2) \propto \exp\left(-\frac{1}{2}\left(\left(\begin{array}{c} x_1 \\ x_2 \end{array}\right)-\left(\begin{array}{c} \mu_1 \\ \mu_2 \end{array}\right)\right)'\left(\begin{array}{cc} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{array}\right)^{-1}\left(\left(\begin{array}{c} x_1 \\ x_2 \end{array}\right)-\left(\begin{array}{c} \mu_1 \\ \mu_2 \end{array}\right)\right)\right)$$
$$\propto \exp\left(-\frac{(x_1-(\mu_1+\sigma_{12}/\sigma_{22}^2(x_2-\mu_2)))^2}{2(\sigma_1^2-(\sigma_{12})^2/\sigma_2^2)}\right),$$

i.e.

$$(X_1 \mid X_2 = x_2) \sim \mathsf{N}\left(\mu_1+\sigma_{12}/\sigma_2^2(x_2-\mu_2), \sigma_1^2-(\sigma_{12})^2/\sigma_2^2\right).$$

Thus the Gibbs sampler for this problem consists of iterating for $t = 1, 2, \ldots$

1. *Draw* $X_1^{(t)} \sim \mathsf{N}\left(\mu_1+\sigma_{12}/\sigma_2^2(X_2^{(t-1)}-\mu_2), \sigma_1^2-(\sigma_{12})^2/\sigma_2^2\right)$.
2. *Draw* $X_2^{(t)} \sim \mathsf{N}\left(\mu_2+\sigma_{12}/\sigma_1^2(X_1^{(t)}-\mu_1), \sigma_2^2-(\sigma_{12})^2/\sigma_1^2\right)$.

Now consider the special case $\mu_1 = \mu_2 = 0$, $\sigma_1^2 = \sigma_2^2 = 1$ and $\sigma_{12} = 0.3$. Figures 3.4–3.6 show the sample paths of this Gibbs sampler.
Using Theorem 3.4 we can estimate $\mathbb{P}(X_1 \geq 0, X_2 \geq 0)$ by the proportion of samples $(X_1^{(t)}, X_2^{(t)})$ with $X_1^{(t)} \geq 0$ and $X_2^{(t)} \geq 0$. Figure 3.3 shows this estimate.

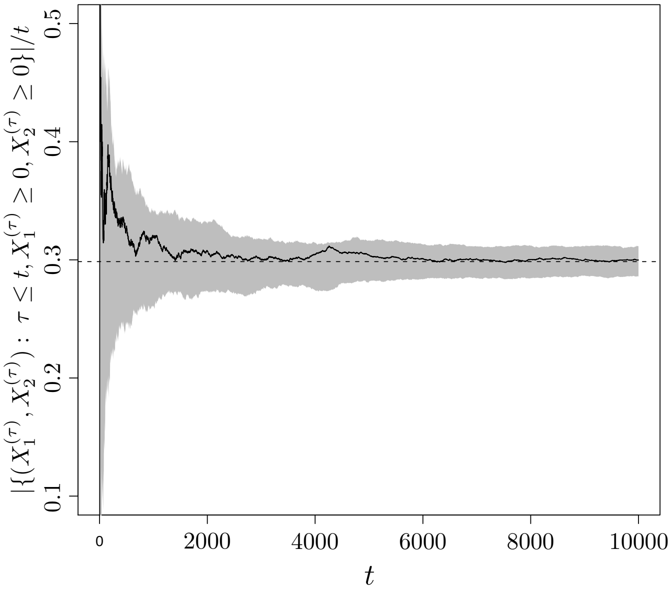

Figure 3.3: Estimate of the $\mathbb{P}(X_1 \geq 0, X_2 \geq 0)$ obtained using a Gibbs sampler. The area shaded in grey corresponds to the range of 100 replications.

[2] We make use of

$$\left(\left(\begin{array}{c} x_1 \\ x_2 \end{array}\right)-\left(\begin{array}{c} \mu_1 \\ \mu_2 \end{array}\right)\right)'\left(\begin{array}{cc} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{array}\right)^{-1}\left(\left(\begin{array}{c} x_1 \\ x_2 \end{array}\right)-\left(\begin{array}{c} \mu_1 \\ \mu_2 \end{array}\right)\right)$$
$$= \frac{1}{\sigma_1^2\sigma_2^2-(\sigma_{12})^2}\left(\left(\begin{array}{c} x_1 \\ x_2 \end{array}\right)-\left(\begin{array}{c} \mu_1 \\ \mu_2 \end{array}\right)\right)'\left(\begin{array}{cc} \sigma_2^2 & -\sigma_{12} \\ -\sigma_{12} & \sigma_1^2 \end{array}\right)\left(\left(\begin{array}{c} x_1 \\ x_2 \end{array}\right)-\left(\begin{array}{c} \mu_1 \\ \mu_2 \end{array}\right)\right)$$
$$= \frac{1}{\sigma_1^2\sigma_2^2-(\sigma_{12})^2}\left(\sigma_2^2(x_1-\mu_1)^2-2\sigma_{12}(x_1-\mu_1)(x_2-\mu_2)\right) + \text{ const}$$
$$= \frac{1}{\sigma_1^2\sigma_2^2-(\sigma_{12})^2}\left(\sigma_2^2x_1^2-2\sigma_2^2x_1\mu_1-2\sigma_{12}x_1(x_2-\mu_2)\right) + \text{ const}$$
$$= \frac{1}{\sigma_1^2-(\sigma_{12})^2/\sigma_2^2}\left(x_1^2-2x_1(\mu_1+\sigma_{12}/\sigma_2^2(x_2-\mu_2))\right) + \text{ const}$$
$$= \frac{1}{\sigma_1^2-(\sigma_{12})^2/\sigma_2^2}\left(x_1-(\mu_1+\sigma_{12}/\sigma_2^2(x_2-\mu_2)\right)^2 + \text{ const}$$

A Gibbs sampler is of course not the optimal way to sample from a $\mathsf{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ distribution. A more efficient way is: draw $Z_1, \ldots, Z_p \overset{\text{iid}}{\sim} \mathsf{N}(0,1)$ and set $(X_1, \ldots, X_p)' = \boldsymbol{\Sigma}^{1/2}(Z_1, \ldots, Z_p)' + \boldsymbol{\mu}$. As we shall see, in some instances the loss of efficiency arising from Gibbs sampling can be very severe.

Note that the realisations $(\boldsymbol{x}^{(0)}, \boldsymbol{x}^{(1)}, \ldots)$ form a Markov chain, and are thus *not* independent, but typically positively correlated. The correlation between the $\boldsymbol{x}^{(t)}$ is larger if the Markov chain moves only slowly (the chain is then said to be *slowly mixing*). For the Gibbs sampler this is typically the case if the variables X_j are strongly (positively or negatively) correlated, as the following example shows.

Example 3.5. (Sampling from a highly correlated bivariate Gaussian).

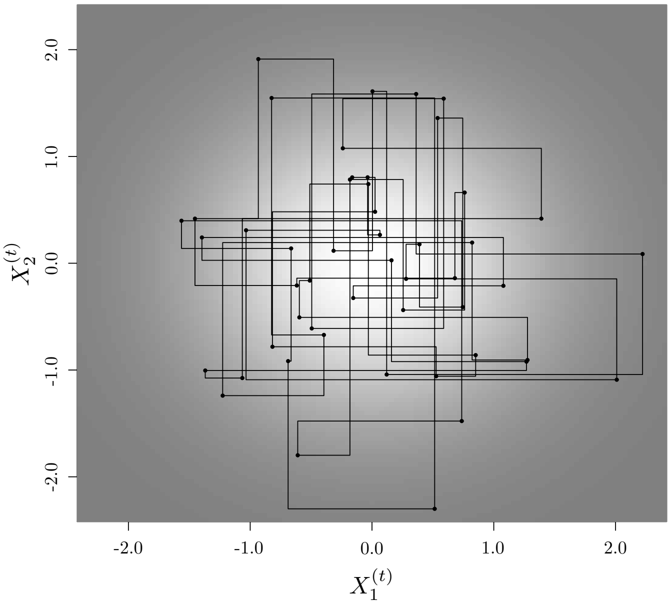

Figure 3.4: Gibbs sampler for a bivariate standard normal distribution (correlation $\rho(X_1, X_2) = 0.3$): first 50 iterations.

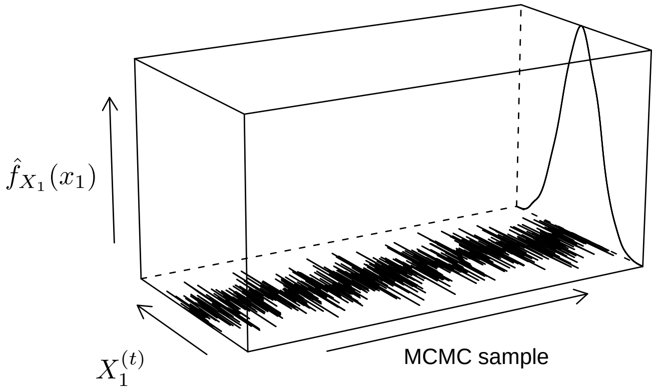

Figure 3.5: Gibbs sampler for a bivariate standard normal distribution (correlation $\rho(X_1, X_2) = 0.3$): Path of $X_1^{(t)}$ and estimated density of X after 1,000 iterations.

Figures 3.7–3.9 show the results obtained when sampling from a bivariate Normal distribution as in Example 3.4, however with $\sigma_{12} = 0.99$. This yields a correlation of $\rho(X_1, X_2) = 0.99$. This Gibbs sampler is a lot slower mixing than the one considered in Example 3.4 (and displayed in Figure 3.4–3.6): due to the strong correlation the Gibbs sampler can only perform very small movements. This makes subsequent samples $X_j^{(t-1)}$ and $X_j^{(t)}$ highly correlated

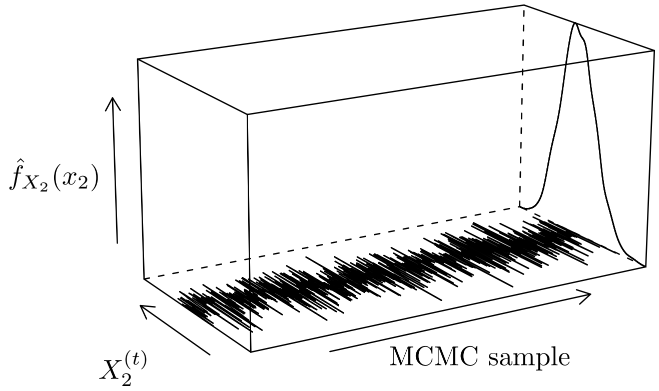

Figure 3.6: Gibbs sampler for a bivariate standard normal distribution (correlation $\rho(X_1, X_2) = 0.3$): Path of $X_2^{(t)}$ and estimated density of X_2 after 1,000 iterations.

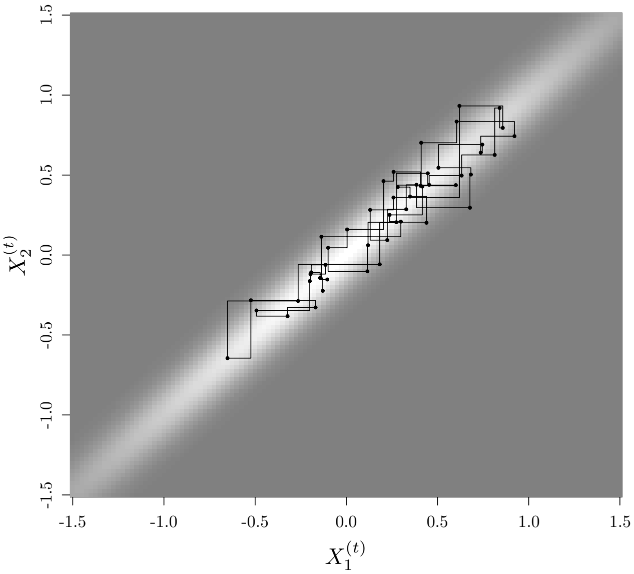

Figure 3.7: Gibbs sampler for a bivariate normal distribution with correlation $\rho(X_1, X_2) = 0.99$: first 50 iterations.

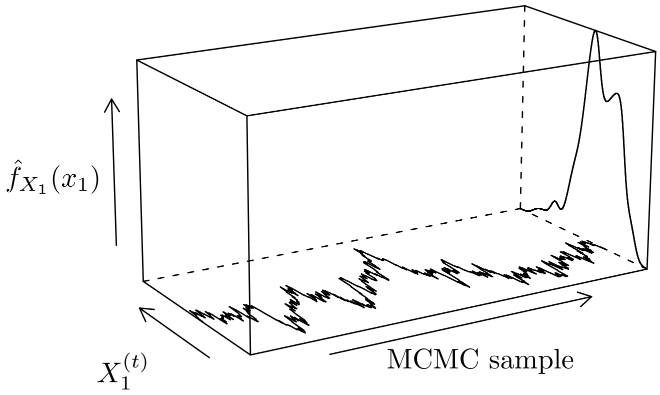

Figure 3.8: Gibbs sampler for a bivariate normal distribution with correlation $\rho(X_1, X_2) = 0.99$: Path of $X_1^{(t)}$ and estimated density of X_1 after 1,000 iterations.

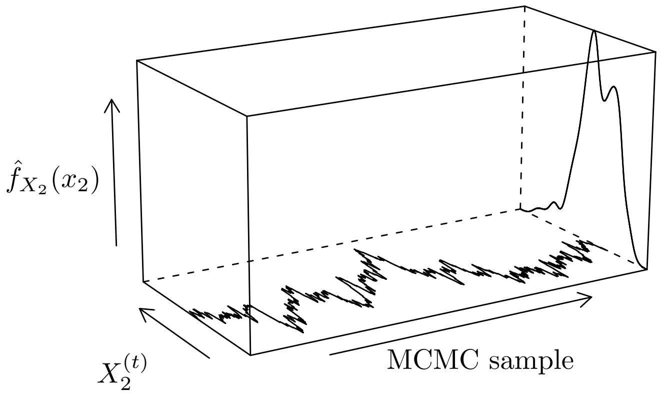

Figure 3.9: Gibbs sampler for a bivariate normal distribution with correlation $\rho(X_1, X_2) = 0.99$: Path of $X_2^{(t)}$ and estimated density of X_2 after 1,000 iterations.

and this leads to slower convergence, as the plot of the estimated densities show (compare Figures 3.5–3.6 with 3.8–3.9).

3.2.2 Metropolis and Beyond

Although the Gibbs sampler is appealing and appears generally applicable, there are some difficulties with it. In particular, it requires that the full conditional distributions are known and can be sampled from (and in order to be efficient these need to be the full conditional distributions of groups of highly-dependent subsets of the random variables which can further complicate the problem). We turn our attention now to a still more broadly-applicable class of MCMC algorithms based around an accept/reject mechanism.

The Metropolis–Hastings algorithm dates back to Metropolis et al. (1953) and Hastings (1970). Like rejection sampling (Algorithm 2.1), the Metropolis–Hastings algorithm is based on proposing values sampled from a proposal distribution, which are then accepted with a certain probability that reflects how likely it is that they are from the target distribution f.

The main drawback of the rejection sampling algorithm is that it is often very difficult to come up with a suitable proposal distribution that leads to an efficient algorithm. One way around this problem is to allow for "local updates", i.e. let the proposed value depend on the last accepted value. This makes it easier to come up with a suitable (conditional) proposal, however at the price of yielding a Markov chain instead of a sequence of independent realisations.

Algorithm 3.3 (Metropolis–Hastings). *Starting with* $\boldsymbol{X}^{(0)} := (X_1^{(0)}, \ldots, X_p^{(0)})$ *iterate for* $t = 1, 2, \ldots$

1. *Draw* $\boldsymbol{X} \sim q(\cdot \mid \boldsymbol{X}^{(t-1)})$.
2. *Compute*
$$\alpha(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)}) = \min\left\{1, \frac{f(\boldsymbol{X}) \cdot q(\boldsymbol{X}^{(t-1)} \mid \boldsymbol{X})}{f(\boldsymbol{X}^{(t-1)}) \cdot q(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)})}\right\}. \tag{3.10}$$
3. *With probability* $\alpha(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)})$ *set* $\boldsymbol{X}^{(t)} = \boldsymbol{X}$*, otherwise set* $\boldsymbol{X}^{(t)} = \boldsymbol{X}^{(t-1)}$.

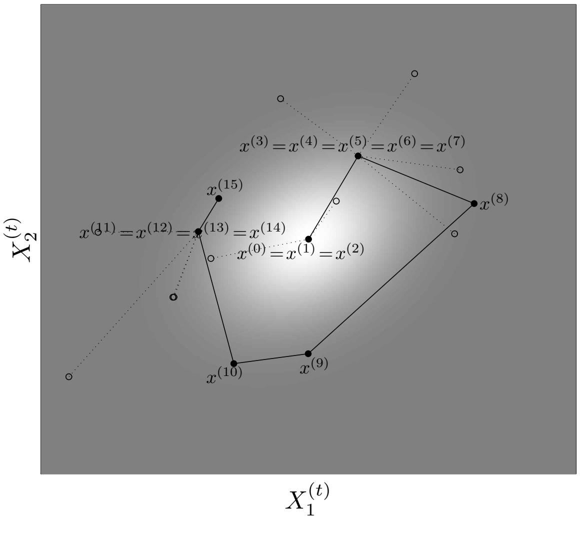
$x^{(3)} = x^{(4)} = x^{(5)} = x^{(6)} = x^{(7)}$
$x^{(15)}$
$x^{(8)}$
$x^{(11)} = x^{(12)} = x^{(13)} = x^{(14)}$
$x^{(0)} = x^{(1)} = x^{(2)}$
$x^{(10)}$
$x^{(9)}$
$X_2^{(t)}$
$X_1^{(t)}$

Figure 3.10: Illustration of the Metropolis–Hastings algorithm. Filled dots denote accepted states, open circles rejected values.

Figure 3.10 illustrates the Metropolis–Hastings algorithm. Note that if the algorithm rejects the newly proposed value (open disks joined by dotted lines in Figure 3.10) it *accepts its current value* $\boldsymbol{X}^{(t-1)}$ *instead.* (*The Metropolis–Hastings* *algorithm never rejects.*) The probability that the Metropolis–Hastings algorithm accepts the newly proposed state $\boldsymbol{X}$ given that it currently is in state $\boldsymbol{X}^{(t-1)}$ is

$$a(\boldsymbol{x}^{(t-1)}) = \int \alpha(\boldsymbol{x} \mid \boldsymbol{x}^{(t-1)}) q(\boldsymbol{x} \mid \boldsymbol{x}^{(t-1)})\, d\boldsymbol{x}.$$

Just like the Gibbs sampler, the Metropolis–Hastings algorithm generates a Markov chain, whose properties will be discussed in the next section.

Remark. The probability of acceptance (3.10) *does not depend on the normalisation constant: if* $f(\boldsymbol{x}) = C \cdot \pi(\boldsymbol{x})$*, then*

$$\frac{f(\boldsymbol{x}) \cdot q(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x})}{f(\boldsymbol{x}^{(t-1)}) \cdot q(\boldsymbol{x} \mid \boldsymbol{x}^{(t-1)})} = \frac{C\pi(\boldsymbol{x}) \cdot q(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x})}{C\pi(\boldsymbol{x}^{(t-1)}) \cdot q(\boldsymbol{x} \mid \boldsymbol{x}^{(t-1)})} = \frac{\pi(\boldsymbol{x}) \cdot q(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x})}{\pi(\boldsymbol{x}^{(t-1)}) \cdot q(\boldsymbol{x} \mid \boldsymbol{x}^{(t-1)})}.$$

Thus f only needs to be known up to normalisation constant. Similarly, it is enough to know $q(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x})$ up to a multiplicative constant independent of $\boldsymbol{x}^{(t-1)}$ and $\boldsymbol{x}$.

3.2.3 Convergence of Metropolis–Hastings

Lemma 3.2. The transition kernel[3] of the Metropolis–Hastings algorithm is

$$K(\boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t)}) = \alpha(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)}) q(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)}) + (1 - a(\boldsymbol{x}^{(t-1)}))\delta_{\boldsymbol{x}^{(t-1)}}(\boldsymbol{x}^{(t)}), \tag{3.11}$$

where $\delta_{\boldsymbol{x}^{(t-1)}}(\cdot)$ denotes a probability distribution which places a mass of one at $\boldsymbol{x}^{(t-1)}$.

Proof. We have

$$\begin{aligned}
\mathbb{P}(\boldsymbol{x}^{(t)} \in \mathcal{X} \mid \boldsymbol{x}^{(t-1)} = \boldsymbol{x}^{(t-1)}) =& \mathbb{P}(\boldsymbol{x}^{(t)} \in \mathcal{X}, \text{new value accepted} \mid \boldsymbol{x}^{(t-1)} = \boldsymbol{x}^{(t-1)})\\
&+ \mathbb{P}(\boldsymbol{x}^{(t)} \in \mathcal{X}, \text{new value rejected} \mid \boldsymbol{x}^{(t-1)} = \boldsymbol{x}^{(t-1)})\\
=& \int_{\mathcal{X}} \alpha(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)}) q(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)})\, d\boldsymbol{x}^{(t)}\\
&+ \underbrace{\quad \underbrace{\mathbb{I}_{\mathcal{X}}(\boldsymbol{x}^{(t-1)})}_{=\int_{\mathcal{X}} \delta_{\boldsymbol{x}^{(t-1)}}(d\boldsymbol{x}^{(t)})} \quad \underbrace{\mathbb{P}(\text{new value rejected} \mid \boldsymbol{x}^{(t-1)} = \boldsymbol{x}^{(t-1)})}_{=1-a(\boldsymbol{x}^{(t-1)})}}_{=\int_{\mathcal{X}} (1-a(\boldsymbol{x}^{(t-1)}))\delta_{\boldsymbol{x}^{(t-1)}}(d\boldsymbol{x}^{(t)})}\\
=& \int_{\mathcal{X}} \alpha(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)}) q(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)})\, d\boldsymbol{x}^{(t)}\\
&+ \int_{\mathcal{X}} (1 - a(\boldsymbol{x}^{(t-1)}))\delta_{\boldsymbol{x}^{(t-1)}}(d\boldsymbol{x}^{(t)}).
\end{aligned}$$

□

Proposition 3.3. The Metropolis–Hastings kernel (3.11) satisfies the **detailed balance condition**

$$K(\boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t)}) f(\boldsymbol{x}^{(t-1)}) = K(\boldsymbol{x}^{(t)}, \boldsymbol{x}^{(t-1)}) f(\boldsymbol{x}^{(t)})$$

and thus $f(\boldsymbol{x})$ is the invariant distribution of the Markov chain $(\boldsymbol{x}^{(0)}, \boldsymbol{x}^{(1)}, \dots)$ generated by the Metropolis–Hastings sampler. Furthermore the Markov chain is reversible.

Proof. We have that

$$\begin{aligned}
\alpha(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)}) q(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)}) f(\boldsymbol{x}^{(t-1)}) &= \min\left\{1, \frac{f(\boldsymbol{x}^{(t)}) q(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x}^{(t)})}{f(\boldsymbol{x}^{(t-1)}) q(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)})}\right\} q(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)}) f(\boldsymbol{x}^{(t-1)})\\
&= \min\left\{f(\boldsymbol{x}^{(t-1)}) q(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)}), f(\boldsymbol{x}^{(t)}) q(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x}^{(t)})\right\}\\
&= \min\left\{\frac{f(\boldsymbol{x}^{(t-1)}) q(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)})}{f(\boldsymbol{x}^{(t)}) q(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x}^{(t)})}, 1\right\} q(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x}^{(t)}) f(\boldsymbol{x}^{(t)})\\
&= \alpha(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x}^{(t)}) q(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x}^{(t)}) f(\boldsymbol{x}^{(t)}),
\end{aligned}$$

[3] *Note that, despite the slight abuse of notation, the transition kernel* (3.11) *is **not** absolutely continuous with respect to the Lebesgue measure (i.e. it doesn't have a simple density).*

and thus

$$\begin{aligned}K(\boldsymbol{x}^{(t-1)},\boldsymbol{x}^{(t)})f(\boldsymbol{x}^{(t-1)}) = {} & \underbrace{\alpha(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)})q(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)})f(\boldsymbol{x}^{(t-1)})}_{=\alpha(\boldsymbol{x}^{(t-1)}|\boldsymbol{x}^{(t)})q(\boldsymbol{x}^{(t-1)}|\boldsymbol{x}^{(t)})f(\boldsymbol{x}^{(t)})} \\ & + \underbrace{(1-a(\boldsymbol{x}^{(t-1)}))\underbrace{\delta_{\boldsymbol{x}^{(t-1)}}(\boldsymbol{x}^{(t)})}_{=0 \text{ if } \boldsymbol{x}^{(t)} \neq \boldsymbol{x}^{(t-1)}} f(\boldsymbol{x}^{(t-1)})}_{(1-a(\boldsymbol{x}^{(t)}))\delta_{\boldsymbol{x}^{(t)}}(\boldsymbol{x}^{(t-1)})f(\boldsymbol{x}^{(t)})} = K(\boldsymbol{x}^{(t)},\boldsymbol{x}^{(t-1)})f(\boldsymbol{x}^{(t)}).\end{aligned}$$

The other conclusions follow by Proposition A.2, suitably adapted to the continuous case (i.e. replacing the sums by integrals). □

Next we need to examine whether the Metropolis–Hastings algorithm yields an irreducible chain. As with the Gibbs sampler, this is not necessarily the case, as the following example shows.

Example 3.6. (Reducible Metropolis–Hastings). *Consider using a Metropolis–Hastings algorithm for sampling from a uniform distribution on* $[0,1] \cup [2,3]$ *and a* $\mathsf{U}\big[x^{(t-1)}-\delta, x^{(t-1)}+\delta\big]$ *distribution as proposal distribution* $q(\cdot \mid x^{(t-1)})$*. Figure 3.11 illustrates this example. It is easy to see that the resulting Markov chain is **not** irreducible if* $\delta \leq 1$*: in this case the chain either stays in* $[0,1]$ *or* $[2,3]$.

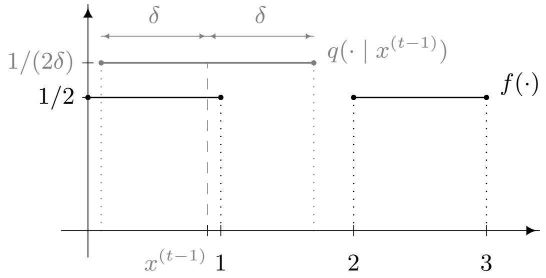

Figure 3.11: Illustration of Reducible Metropolis–Hastings.

Under mild assumptions on the proposal $q(\cdot \mid \boldsymbol{x}^{(t-1)})$ one can however establish the irreducibility of the resulting Markov chain:

- If $q(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)})$ is positive for all $\boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t)} \in \operatorname{supp}(f)$, then it is easy to see that we can reach any set of non-zero probability under f within a single step. The resulting Markov chain is thus strongly irreducible. Even though this condition seems rather restrictive, many popular choices of $q(\cdot \mid \boldsymbol{x}^{(t-1)})$ like multivariate Gaussian or t-distributions fulfil this condition.
- G. Roberts and Tweedie (1996, Theorem 2.2) gives a more general condition for the irreducibility of the resulting Markov chain: they only require that

$$\exists\, \epsilon > 0,\, \delta > 0 : \; q(\boldsymbol{x}^{(t)} \mid \boldsymbol{x}^{(t-1)}) > \epsilon \text{ if } \|\boldsymbol{x}^{(t-1)} - \boldsymbol{x}^{(t)}\| < \delta$$

  together with the boundedness (away from both zero and infinity) of f on any compact set.

The Markov chain $(\boldsymbol{x}^{(0)}, \boldsymbol{x}^{(1)}, \dots)$ is further aperiodic if there is positive probability that the chain remains in the current state, i.e. $\mathbb{P}(\boldsymbol{x}^{(t)} = \boldsymbol{x}^{(t-1)}) > 0$, which is the case if

$$\mathbb{P}\left(f(\boldsymbol{x}^{(t-1)})q(\boldsymbol{x} \mid \boldsymbol{x}^{(t-1)}) > f(\boldsymbol{x})q(\boldsymbol{x}^{(t-1)} \mid \boldsymbol{x})\right) > 0.$$

Note that this condition is *not* met if we use a “perfect” proposal which has f as invariant distribution: in this case we accept every proposed value with probability 1 (see e.g. remark after Example 3.10).

Proposition 3.4. The Markov chain generated by the Metropolis–Hastings algorithm is Harris-recurrent if it is irreducible.

Proof. Recurrence follows from the irreducibility and the fact that f is the invariant distribution. For a proof of Harris recurrence see Tierney (1994, Corollary 2). □

As we have now established (Harris-)recurrence, we are now ready to state an ergodic theorem (using Theorems A.1 and A.2.

Theorem 3.5. If the Markov chain generated by the Metropolis–Hastings algorithm is irreducible, then for any integrable function $h : E \to \mathbb{R}$:

$$\lim_{n\to\infty} \frac{1}{n} \sum_{t=1}^{n} h(\boldsymbol{x}^{(t)}) \to \mathbb{E}_f\left[h(\boldsymbol{X})\right]$$

for every starting value $\boldsymbol{x}^{(0)}$.

As with the Gibbs sampler, the above ergodic theorem allows for inference using a single Markov chain.

3.2.4 The random walk Metropolis algorithm

In this section we will focus on an important special case of the Metropolis–Hastings algorithm: the random walk Metropolis–Hastings algorithm. Assume that we generate the newly proposed state $\boldsymbol{X}$ not using the fairly general

$$\boldsymbol{X} \sim q(\cdot \mid \boldsymbol{X}^{(t-1)}), \tag{3.12}$$

from Algorithm 3.3, but rather

$$\boldsymbol{X} = \boldsymbol{X}^{(t-1)} + \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim g, \tag{3.13}$$

with g being a *symmetric* distribution. It is easy to see that (3.13) is a special case of (3.12) using $q(\boldsymbol{x} \mid \boldsymbol{x}^{(t-1)}) = g(\boldsymbol{x} - \boldsymbol{x}^{(t-1)})$. When using (3.13) the probability of acceptance simplifies to

$$\min\left\{1, \frac{f(\boldsymbol{X}) \cdot q(\boldsymbol{X}^{(t-1)} \mid \boldsymbol{X})}{f(\boldsymbol{X}^{(t-1)}) \cdot q(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)})}\right\} = \min\left\{1, \frac{f(\boldsymbol{X})}{f(\boldsymbol{X}^{(t-1)})}\right\},$$

as $q(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)}) = g(\boldsymbol{X} - \boldsymbol{X}^{(t-1)}) = g(\boldsymbol{X}^{(t-1)} - \boldsymbol{X}) = q(\boldsymbol{X}^{(t-1)} \mid \boldsymbol{X})$ using the symmetry of g. This yields the following algorithm, a special case of Algorithm 3.3, which is actually the original algorithm proposed by Metropolis et al. (1953).

Algorithm 3.4 (Random walk Metropolis). *Starting with $\boldsymbol{X}^{(0)} := (X_1^{(0)}, \dots, X_p^{(0)})$ and using a symmetric distribution g, iterate for $t = 1, 2, \dots$*

1. *Draw $\boldsymbol{\epsilon} \sim g$ and set $\boldsymbol{X} = \boldsymbol{X}^{(t-1)} + \boldsymbol{\epsilon}$.*
2. *Compute*

$$\alpha(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)}) = \min\left\{1, \frac{f(\boldsymbol{X})}{f(\boldsymbol{X}^{(t-1)})}\right\}.$$

3. *With probability $\alpha(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)})$ set $\boldsymbol{X}^{(t)} = \boldsymbol{X}$, otherwise set $\boldsymbol{X}^{(t)} = \boldsymbol{X}^{(t-1)}$.*

Example 3.7. (Bayesian probit model). *In a medical study on infections resulting from birth by Cæsarean section (taken from Fahrmeir and Tutz 2001) three influence factors have been studied: an indicator whether the Cæsarian was planned or not (z_{i1}), an indicator of whether additional risk factors were present at the time of birth (z_{i2}), and an indicator of whether antibiotics were given as a prophylaxis (z_{i3}). The response Y_i is the number of infections that were observed amongst n_i patients having the same influence factors (covariates). The data is given in Table 3.1.*

Table 3.1: Data used in Example 3.7.

Number of births with infection	total	planned	risk factors	antibiotics
y_i	n_i	z_{i1}	z_{i2}	z_{i3}
11	98	1	1	1
1	18	0	1	1
0	2	0	0	1
23	26	1	1	0
28	58	0	1	0
0	9	1	0	0
8	40	0	0	0

The data can be modelled by assuming that

$$Y_i \sim \mathsf{Bin}(n_i, \pi_i), \qquad \pi = \Phi(\boldsymbol{z}_i'\boldsymbol{\beta}),$$

where $\boldsymbol{z}_i = (1, z_{i1}, z_{i2}, z_{i3})$ *and* $\Phi(\cdot)$ *being the CDF of the* $\mathsf{N}(0,1)$ *distribution. Note that* $\Phi(t) \in [0,1]$ *for all* $t \in \mathbb{R}$. *A suitable prior distribution for the parameter of interest* $\boldsymbol{\beta}$ *is* $\boldsymbol{\beta} \sim N(\mathbf{0}, \mathbb{I}/\lambda)$. *The posterior density of* $\boldsymbol{\beta}$ *is*

$$f(\boldsymbol{\beta} \mid y_1, \ldots, y_n) \propto \left(\prod_{i=1}^{N} \Phi(\boldsymbol{z}_i'\boldsymbol{\beta})^{y_i} \cdot (1 - \Phi(\boldsymbol{z}_i'\boldsymbol{\beta}))^{n_i - y_i} \right) \cdot \exp\left(-\frac{\lambda}{2} \sum_{j=0}^{3} \beta_j^2 \right).$$

We can sample from the above posterior distribution using the following random walk Metropolis algorithm. Starting with any $\boldsymbol{\beta}^{(0)}$ *iterate for* $t = 1, 2, \ldots$*:*

1. *Draw* $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \boldsymbol{\Sigma})$ *and set* $\boldsymbol{\beta} = \boldsymbol{\beta}^{(t-1)} + \boldsymbol{\epsilon}$.
2. *Compute*

$$\alpha(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(t-1)}) = \min\left\{ 1, \frac{f(\boldsymbol{\beta} \mid Y_1, \ldots, Y_n)}{f(\boldsymbol{\beta}^{(t-1)} \mid Y_1, \ldots, Y_n)} \right\}.$$

3. *With probability* $\alpha(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(t-1)})$ *set* $\boldsymbol{\beta}^{(t)} = \boldsymbol{\beta}$, *otherwise set* $\boldsymbol{\beta}^{(t)} = \boldsymbol{\beta}^{(t-1)}$.

To keep things simple, we choose the covariance $\boldsymbol{\Sigma}$ of the proposal to be $0.08 \cdot \mathbb{I}$.

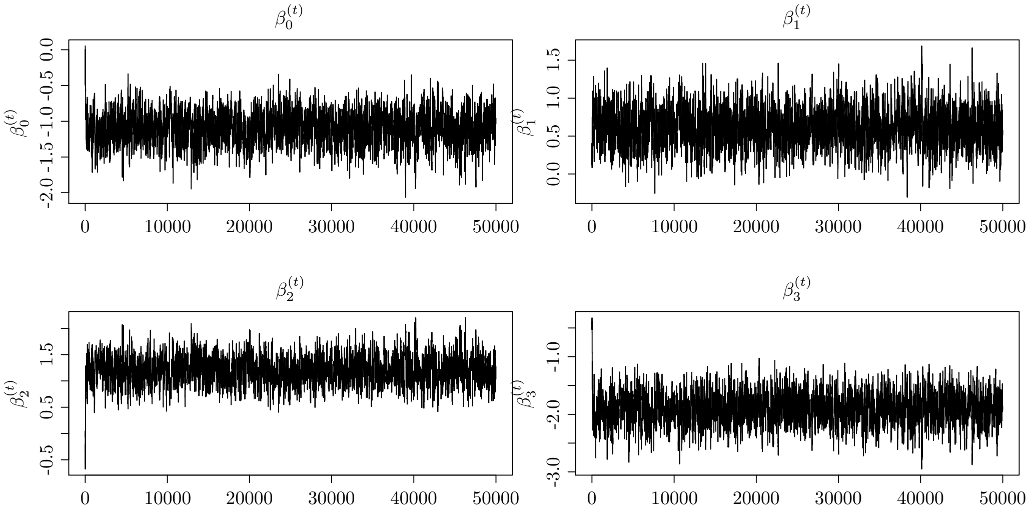

Figure 3.12: Results obtained for the Bayesian probit model: Sample paths of the $\beta_j^{(t)}$.

Table 3.2: Parameter estimates obtained for the Bayesian probit model from Example 3.7.

		Posterior mean	95% credible interval	
intercept	β_0	-1.0952	-1.4646	-0.7333
planned	β_1	0.6201	0.2029	1.0413
risk factors	β_2	1.2000	0.7783	1.6296
antibiotics	β_3	-1.8993	-2.3636	-1.471

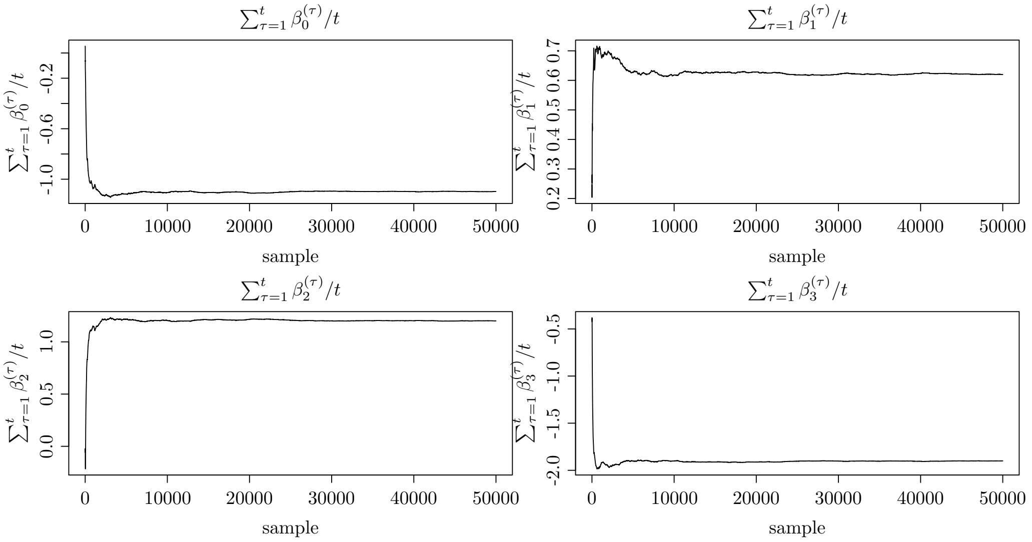

Figure 3.13: Results obtained for the Bayesian probit model: Cumulative averages $\sum_{\tau=1}^{t} \beta_j^{(\tau)}/t$.

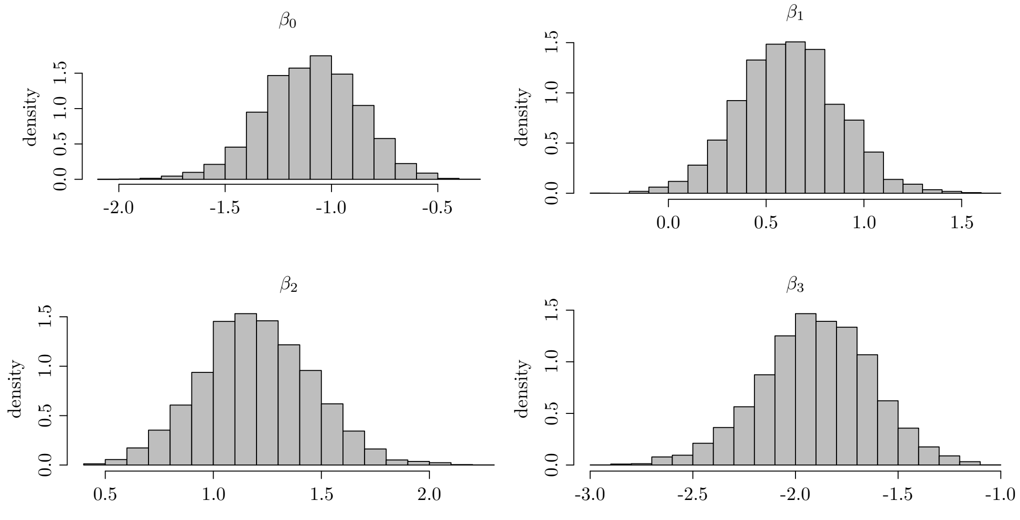

Figure 3.14: Results obtained for the Bayesian probit model: Posterior distributions of the β_j.

Figure 3.12–3.14 and Table 3.2 show the results obtained using 50,000 samples (you might want to consider a longer chain in practice). Note that the convergence of the $\beta_j^{(t)}$ is to a distribution, whereas the cumulative averages $\sum_{\tau=1}^{t} \beta_j^{(\tau)}/t$ converge, as the ergodic theorem implies, to a value. For Figure 3.12–3.14 and Table 3.2 the first 10,000 samples have been discarded ("burn-in").

3.2.4.1 Choosing the proposal distribution

The efficiency of a Metropolis–Hastings sampler depends on the choice of the proposal distribution $q(\cdot \mid \boldsymbol{x}^{(t-1)})$. An ideal choice of proposal would lead to a small correlation of subsequent realisations $\boldsymbol{X}^{(t-1)}$ and $\boldsymbol{X}^{(t)}$. This correlation has two sources:

- the correlation between the current state $\boldsymbol{X}^{(t-1)}$ and the newly proposed value $\boldsymbol{X} \sim q(\cdot \mid \boldsymbol{X}^{(t-1)})$, and
- the correlation introduced by retaining a value $\boldsymbol{X}^{(t)} = \boldsymbol{X}^{(t-1)}$ because the newly generated value $\boldsymbol{X}$ has been rejected.

Thus we would ideally want a proposal distribution that both allows for fast changes in the $\boldsymbol{X}^{(t)}$ and yields a high probability of acceptance. Unfortunately these are two competing goals. If we choose a proposal distribution with a small variance, the probability of acceptance will be high, however the resulting Markov chain will be highly correlated, as the $\boldsymbol{X}^{(t)}$ change only very slowly. If, on the other hand, we choose a proposal distribution with a large variance, the $\boldsymbol{X}^{(t)}$ can potentially move very fast, however the probability of acceptance will be rather low.

Example 3.8. *Assume we want to sample from a $N(0,1)$ distribution using a random walk Metropolis–Hastings algorithm with $\varepsilon \sim N(0,\sigma^2)$. At first sight, we might think that setting $\sigma^2 = 1$ is the optimal choice, this is however not the case. In this example we examine the choices: $\sigma^2 = 0.1$, $\sigma^2 = 1$, $\sigma^2 = 2.38^2$, and $\sigma^2 = 10^2$. Figure 3.15 shows the sample paths of a single run of the corresponding random walk Metropolis–Hastings algorithm. Rejected values are drawn as grey open circles. Table 3.3 shows the average correlation $\rho(X^{(t-1)}, X^{(t)})$ as well as the average probability of acceptance $\alpha(X \mid X^{(t-1)})$ averaged over 100 runs of the algorithm. Choosing σ^2 too small yields a very high probability of acceptance, however at the price of a chain that is hardly moving. Choosing σ^2 too large allows the chain to make large jumps; however, most of the proposed values are rejected, so the chain remains for a long time at each accepted value. The results suggest that $\sigma^2 = 2.38^2$ is the optimal choice. This corresponds to the theoretical results of Gelman, Roberts, and Gilks (1995) and the many papers which have extended the original result.*

Table 3.3: Average correlation $\rho(X^{(t-1)}, X^{(t)})$ and average probability of acceptance $\alpha(X \mid X^{(t-1)})$ found in Example 3.8 for different choices of the proposal variance σ^2.

	Autocorrelation		*Probability of acceptance*	
	$\rho(X^{(t-1)}, X^{(t)})$		$\alpha(X, X^{(t-1)})$	
	Mean	*95% CI*	*Mean*	*95% CI*
$\sigma^2 = 0.1^2$	*0.9901*	*(0.9891,0.9910)*	*0.9694*	*(0.9677,0.9710)*
$\sigma^2 = 1$	*0.7733*	*(0.7676,0.7791)*	*0.7038*	*(0.7014,0.7061)*
$\sigma^2 = 2.38^2$	*0.6225*	*(0.6162,0.6289)*	*0.4426*	*(0.4401,0.4452)*
$\sigma^2 = 10^2$	*0.8360*	*(0.8303,0.8418)*	*0.1255*	*(0.1237,0.1274)*

Finding the ideal proposal distribution $q(\cdot \mid \boldsymbol{x}^{(t-1)})$ is an art. The optimal proposal would be sampling directly from the target distribution. The very reason for using a Metropolis–Hastings algorithm is, however, that we cannot sample directly from the target! This difficulty is the price we have to pay for the generality of the Metropolis–Hastings algorithm. Popular choices for random walk proposals are multivariate Gaussian or t-distributions. The latter have heavier tails, making them a safer choice. The covariance structure of the proposal distribution should ideally reflect the covariance of the target distribution. G. O. Roberts, Gelman, and Gilks (1997) propose to adjust the proposal such that the acceptance rate is around 1/2 for one- or two dimensional target distributions, and around 1/4 for larger dimensions, which is in line with the results we obtained in the above simple example and the guidelines which motivate them. Remember, however, that these are just rough guidelines and there is little to be gained from fine-tuning acceptance rates to several decimal places.

Example 3.9. (Bayesian probit model (continued)). *In the Bayesian probit model we studied in Example 3.7 we*

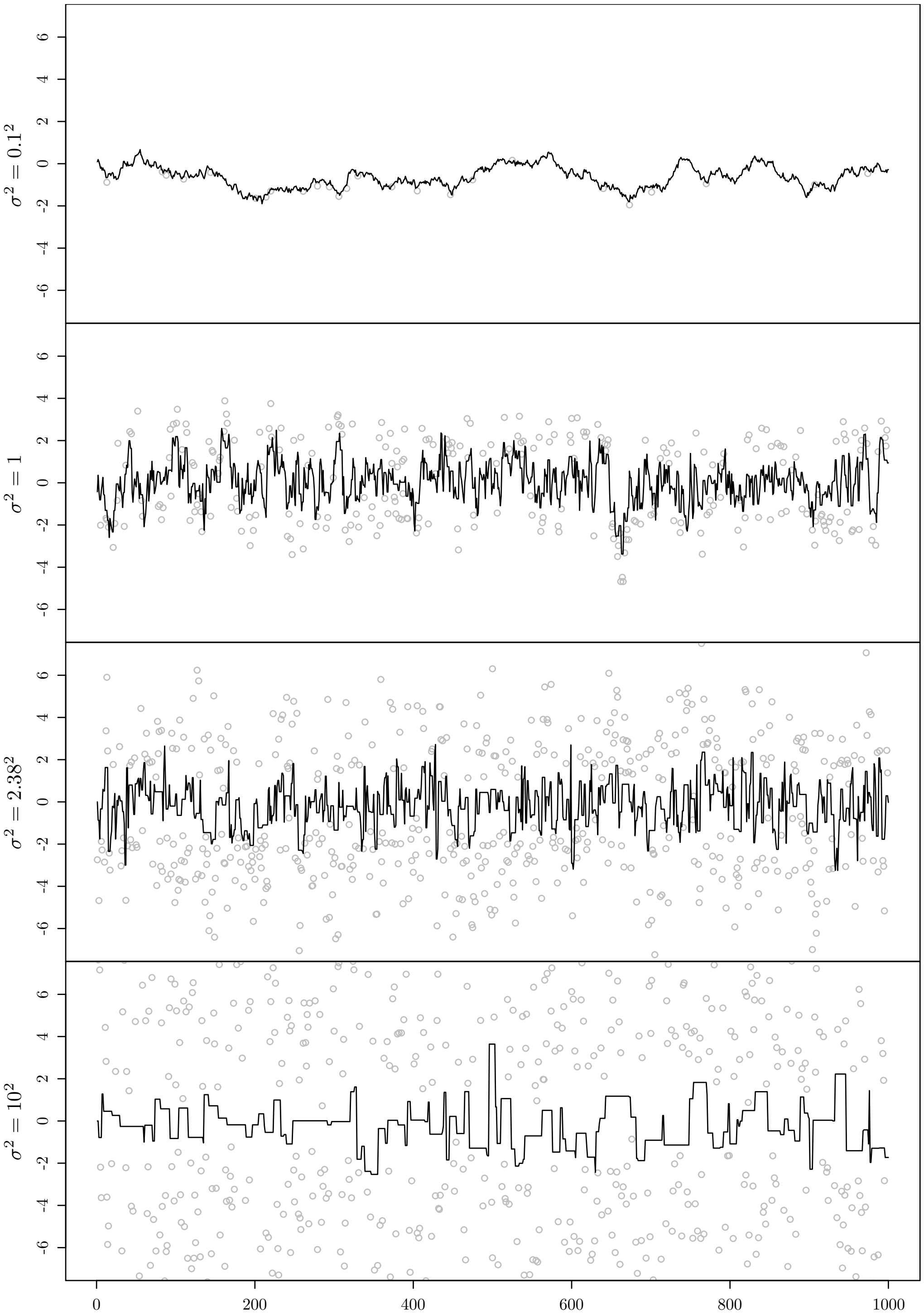

Figure 3.15: Sample paths for the Metropolis–Hastings algoritmhm for simulating from a standard normal distribution using different choices of the proposal variance σ^2. Open grey discs represent rejected values.

drew

$$\boldsymbol{\epsilon} \sim N(\mathbf{0}, \boldsymbol{\Sigma})$$

with $\boldsymbol{\Sigma} = 0.08 \cdot \mathbf{I}$*, i.e. we modelled the components of* $\boldsymbol{\epsilon}$ *to be independent. The proportion of accepted values we obtained was* 13.9%*. Table 3.4 shows the corresponding autocorrelation. The resulting Markov chain can be made faster mixing by using a proposal distribution that represents the covariance structure of the posterior distribution of* $\boldsymbol{\beta}$*. This can be done by resorting to the frequentist theory of generalised linear models (GLM): it suggests that the asymptotic covariance of the maximum likelihood estimate* $\hat{\boldsymbol{\beta}}$ *is* $(\boldsymbol{z}'\boldsymbol{D}\boldsymbol{z})^{-1}$*, where* $\boldsymbol{z}$ *is the matrix of the covariates, and* $\boldsymbol{D}$ *is a suitable diagonal matrix. When using* $\boldsymbol{\Sigma} = 2 \cdot (\boldsymbol{z}'\boldsymbol{D}\boldsymbol{z})^{-1}$ *in the algorithm presented in Example 3.7 we can obtain better mixing performance: the autocorrelation is reduced (see Table 3.4), and the proportion of accepted values obtained increases to 20.0%. Note that the determinant of both choices of* $\boldsymbol{\Sigma}$ *was chosen to be the same, so the improvement of the mixing behaviour is due to a difference in the structure of the covariance.*

Table 3.4: Autocorrelation $\rho(\beta_j^{(t-1)}, \beta_j^{(t)})$ between subsequent samples for two choices of the covariance $\boldsymbol{\Sigma}$.

	β_0	β_1	β_2	β_3
Autocorrelation for $\boldsymbol{\Sigma} = 0.08 \cdot \mathbf{I}$	0.9496	0.9503	0.9562	0.9532
Autocorrelation for $\boldsymbol{\Sigma} = 2 \cdot (\boldsymbol{z}'\boldsymbol{D}\boldsymbol{z})^{-1}$	0.8726	0.8765	0.8741	0.8792

3.2.5 The (Metropolised) Independence Sampler

The random walk proposals considered thus far are appealing because we can in principle employ them without detailed knowledge of the structure of the target distribution nor dedicating considerable effort to their design. However, if we do have information about the target distribution we may wish to use it to design *global* rather than local proposals and hence, we might hope, to reduce the autocorrelation of the chain.

This is indeed possible and if we can construct proposal distributions which have a similar form to the target distribution we can obtain good performance within an MCMC algorithm.

Choosing proposals of the form $q(x^{(t)} \mid x^{(t-1)}) = q(x^{(t)})$ (i.e. which are *independent* of the current state) leads to what is known as the *Metropolised Independence Sampler* or, sometimes, just the *Independence Sampler*. This name is potentially a little misleading, as this algorithm does not yield independent samples; it simply employs proposals which are themselves independent of the current state of the chain.

Algorithm 3.5 (Metropolised Independence Sampler). *Starting with* $\boldsymbol{X}^{(0)} := (X_1^{(0)}, \ldots, X_p^{(0)})$ *iterate for* $t = 1, 2, \ldots$

1. *Draw* $\boldsymbol{X} \sim q(\cdot)$.
2. *Compute*

$$\alpha(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)}) = \min\left\{1, \frac{f(\boldsymbol{X}) \cdot q(\boldsymbol{X}^{(t-1)})}{f(\boldsymbol{X}^{(t-1)}) \cdot q(\boldsymbol{X})}\right\}$$
$$= \min\left\{1, \frac{f(\boldsymbol{X})/q(\boldsymbol{X})}{f(\boldsymbol{X}^{(t-1)})/q(\boldsymbol{X}^{(t-1)})}\right\}. \tag{3.14}$$

3. *With probability* $\alpha(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)})$ *set* $\boldsymbol{X}^{(t)} = \boldsymbol{X}$*, otherwise set* $\boldsymbol{X}^{(t)} = \boldsymbol{X}^{(t-1)}$.

The form of the acceptance probability given in Equation (3.14) is highly suggestive: the ratio within the minimum is exactly a ratio of importance weights. If we sampled independently from q and used those samples to approximate expectations with respect to f by importance sampling, we'd be using exactly the numerator of this ratio as the importance weight. If we used the same strategy within a rejection sampling setting, assuming this ratio to be bounded, then we'd need an acceptance probability proportional to this ratio.

Later we will see that under those conditions in which the independence sampler proposal would be a good rejection sampling proposal it will also be a good proposal within a MCMC setting. First, however, it's interesting to consider the relationship between the independence sampler and its rejection-sampling counterpart.

Proposition 3.5. *(Acceptance Rates).* If $f(x)/q(x) \leq M < \infty$ the acceptance rate of the independence sampler is at least as high as that of the corresponding rejection sampler.

Proof. Simply expanding the acceptance probability at any point x we establish that:

$$a(x) = \int q(y)\alpha(x,y)dy = \int q(y) \min\left(1, \frac{f(y)/q(y)}{f(x)/q(x)}\right) dy = \int \min\left(q(y), \frac{f(y)}{f(x)/q(x)}\right) dy$$
$$\geq \int \min\left(f(y)/M, f(y)/M\right) dy \geq 1/M,$$

and as this holds for any M which bounds f/q, the acceptance rate of the independence sampler is lower bounded by the best possible acceptance rate for any rejection sampler. □

However, this comes at a cost: with rejection sampling one obtains independent samples from the target; with the independence sampler this is not the case.

3.2.5.1 Ergodicity and the Independence Sampler

One method of assessing the convergence of Markov chains is to look at how far away from the invariant distribution it is possible for the marginal distribution of the chain to remain after a certain number of iterations. In order to make such an assessment it is necessary to define a distance on the space of probability measures.

Definition 3.2. (Total Variation). *The **total variation** distance between two probability distributions, f and g, may be defined as:*

$$||f - g||_{TV} := 2 \sup_A \left| \int_A f(x) - g(x) dx \right|.$$

Actually, in the case of probability densities, this is exactly the L_1 distance between those densities and you may find this formulation easier to interpret:

Proposition 3.6. For any pair of probability densities defined on a common space E:

$$||f - g||_{TV} = \int |f(x) - g(x)| dx.$$

Proof. Let $A^\star = \{x : f(x) > g(x)\}$. It is clear that for all A:

$$\left| \int_A (f(x) - g(x)) dx \right| \leq \left| \int_{A^\star} f(x) - g(x) dx \right|.$$

Noting further that $\int_{A \cup A^c} (f(x) - g(x)) dx = 0$, we can establish that for any (measurable) A:

$$\int_A (f(x) - g(x)) dx = - \int_{A^c} (f(x) - g(x)) dx$$

and so

$$2 \left| \int_A f(x) - g(x) dx \right| = \left| \int_A f(x) - g(x) dx \right| + \left| \int_{A^c} f(x) - g(x) dx \right|.$$

On $A^\star$, $f(x) > g(x)$ while on $(A^\star)^c$ the reverse is true, so:

$$\left| \int_{A^\star} f(x) - g(x) dx \right| = \int_{A^\star} f(x) - g(x) dx = \int_{A^\star} |f(x) - g(x)| dx$$

and

$$\left| \int_{(A^\star)^c} f(x) - g(x) dx \right| = - \int_{(A^\star)^c} f(x) - g(x) dx = \int_{(A^\star)^c} |f(x) - g(x)| dx.$$

Combining everything, we establish that:

$$\begin{aligned}||f-g||_{TV} := 2\sup_A \left|\int_A f(x)-g(x)dx\right| &= 2\left|\int_{A^\star} f(x)-g(x)dx\right| \\ &= \left|\int_{A^\star} f(x)-g(x)dx\right| + \left|\int_{(A^\star)^c} f(x)-g(x)dx\right| \\ &= \int_{A^\star} |f(x)-g(x)|dx + \int_{(A^\star)^c} |f(x)-g(x)|dx \\ &= \int |f(x)-g(x)|dx.\end{aligned}$$

□

Having defined total variation, we can define three forms of ergodicity.

Definition 3.3. (Forms of Ergodicity). *An f-invariant Markov kernel, K, is said to be **ergodic** if*

$$\lim_{n\to\infty} ||K^n(x,\cdot)-f(\cdot)||_{TV} = 0$$

where $||K^n(x,\cdot)-f(\cdot)||_{TV} = \int |K^n(x,y)-f(y)|dy$.

If this statement can be strengthened to:

$$||K^n(x,\cdot)-f(\cdot)||_{TV} \leq M(x)\rho^n$$

*for some $M(x) < \infty$ and $\rho < 1$ then the kernel is said to be **geometrically ergodic** and if it can be further strengthened to:*

$$||K^n(x,\cdot)-f(\cdot)||_{TV} \leq M\rho^n$$

*for some $M < \infty$ which does not depend upon x then it is said to be **uniformly geometrically ergodic** or simply **uniformly ergodic**.*

These are useful because they tell us something about the qualitative *rate of convergence* of the Markov chain to stationarity. However, we should bear in mind that if we don't know the constants M and ρ then even a uniformly ergodic chain can in practice converge rather slowly (see G. O. Roberts and Rosenthal (2011) for some examples).

We're now in a position to state and prove one celebrated result about independence samplers.

Proposition 3.7. If an independence sampler uses proposal q and target f and $f(y)/q(y) \leq M < \infty$ then the associated Markov kernel is uniformly ergodic.

Proof. We follow the argument of Robert and Casella (2004, Exercise 7.11). First we show that $f(y)/q(y) \leq M \Rightarrow K(x,y) \geq f(y)/M$:

$$\begin{aligned}K(x,y) = q(y)\alpha(x,y) + (1-a(x))\delta_x(y) &\geq q(y)\alpha(x,y) \\ \geq q(y)\min\left(\frac{f(y)/q(y)}{f(x)/q(x)}, 1\right) &= \min\left(\frac{f(y)}{f(x)/q(x)}, q(y)\right).\end{aligned}$$

Under the assumptions of the proposition we have that $f(x)/q(x) \leq M$ and $q(y) \geq f(y)/M$ and so:

$$K(x,y) \geq \min\left(\frac{f(y)}{M}, f(y)/M\right) = f(y)/M. \tag{3.15}$$

Now we establish a preliminary result, defining $A^\star(x) = \{y : f(y) > K(x,y)\}$:

$$\begin{aligned}\sup_A \left|\int_A K(x,y)-f(y)dy\right| &= \left|\int_{A^\star(x)} K(x,y)-f(y)dy\right| \\ &= \int_{A^\star(x)} f(y)-K(x,y)dy \\ &\leq \int_{A^\star(x)} f(y)-(1/M)f(y)dy = 1-\frac{1}{M},\end{aligned}$$

using Equation (3.15) to bound the negative term from below. We use this as a base case for induction. We have (by substituting this bound into the definition of the total variation norm) for $n = 1$: $||K^n(x, \cdot) - f(\cdot)||_{TV} \leq 2(1-1/M)^n$.

We now turn to the induction step, and assume that the hypothesis $||K^n(x, \cdot) - f(\cdot)||_{TV} \leq 2(1 - 1/M)^n$ holds for some n and write, for any (measurable) A:

$$\int_A (K^{n+1}(x, y) - f(y))dy = \int_A \int_A (K^n(u, y) - f(y))dy(K(x, u) - f(u))du$$

(you can check this by remembering that K is f-invariant and expanding the right hand side explicitly).

The induction hypothesis tells us that the integral over y is bounded by $(1 - 1/M)^n$ (it's easy to establish that the integral over any set of the difference between any pair of probability densities is at most half of the total variation distance between those densities), and a similar argument to that used to prove the base case establishes that the integral over u can then be bounded by $(1 - 1/M)$:

$$\begin{aligned}\int_A (K^{n+1}(x, y) - f(y))dy &= \int_A \int_A (K^n(u, y) - f(y))dy(K(x, u) - f(u))du \\ &\leq \int (1 - 1/M)^n (K(x, u) - f(u))du \leq (1 - 1/M)^{n+1}\end{aligned}$$

Writing the total variation as twice the supremum over A of quantities of this form completes the argument. □

3.3 Composing kernels: Mixtures and Cycles

It can be advantageous, especially in the case of more complex distributions, to combine different Metropolis–Hastings updates into a single algorithm. Each of the different Metropolis–Hastings updates corresponds to a transition kernel $K^{(j)}$. As with the substeps of Gibbs sampler there are two simple and valid ways of combining the transition kernels $K^{(1)}, \ldots, K^{(r)}$:

- As in the systematic scan Gibbs sampler, we can cycle through the kernels in a deterministic order, i.e. first carry out the Metropolis–Hastings update corresponding the the kernel $K^{(1)}$, then carry out the one corresponding to $K^{(2)}$, etc. until we start again with $K^{(1)}$. The transition kernel of this composite chain is

$$K^{\circ}(\boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t)}) = \int \cdots \int K^{(1)}(\boldsymbol{x}^{(t-1)}, \boldsymbol{\xi}^{(1)}) K^{(2)}(\boldsymbol{\xi}^{(1)}, \boldsymbol{\xi}^{(2)}) \cdots K^{(r)}(\boldsymbol{\xi}^{(r-1)}, \boldsymbol{x}^{(t)}) \; d\boldsymbol{\xi}^{(r-1)} \cdots d\boldsymbol{\xi}^{(1)}.$$

  If each of the transition kernels $K^{(j)}$ has the invariant distribution f; i.e.

$$\int f(\boldsymbol{x}^{(t-1)}) K(\boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t)}) \; d\boldsymbol{x}^{(t-1)} = f(\boldsymbol{x}^{(t)}),$$

  then K° has f as invariant distribution, too, as

$$\begin{aligned}&\int f(\boldsymbol{x}^{(t-1)}) K^{\circ}(\boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t)}) \; d\boldsymbol{x}^{(t-1)} \\ &= \int \cdots \underbrace{\underbrace{\int \underbrace{\int K^{(1)}(\boldsymbol{x}^{(t-1)}, \boldsymbol{\xi}^{(1)}) f(\boldsymbol{x}^{(t-1)}) d\boldsymbol{x}^{(t-1)}}_{=f(\boldsymbol{\xi}^{(1)})} K^{(2)}(\boldsymbol{\xi}^{(1)}, \boldsymbol{\xi}^{(2)}) d\boldsymbol{\xi}^{(1)}}_{f(\boldsymbol{\xi}^{(2)})} \cdots d\boldsymbol{\xi}^{(r-2)}}_{=f(\boldsymbol{\xi}^{(r-1)})} K^{(r)}(\boldsymbol{\xi}^{(r-1)}, \boldsymbol{x}^{(t)}) d\boldsymbol{\xi}^{(r-1)} \\ &= f(\boldsymbol{x}^{(t)}).\end{aligned}$$

- Alternatively, we can, as in the random scan Gibbs sampler, choose each time at random which of the kernels should be used, i.e. use the kernel $K^{(j)}$ with probability $w_j > 0$ ($\sum_{\iota=1}^r w_\iota = 1$). The corresponding kernel of the composite chain is the mixture

$$K^{+}(\boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t)}) = \sum_{\iota=1}^{r} w_\iota K^{(\iota)}(\boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t)}).$$

Once again, if each of the transition kernels $K^{(j)}$ has the invariant distribution f, then K^{+} has f as invariant distribution:

$$\int f(\boldsymbol{x}^{(t-1)})K^{+}(\boldsymbol{x}^{(t-1)},\boldsymbol{x}^{(t)})\,d\boldsymbol{x}^{(t-1)} = \sum_{\iota=1}^{r} w_\iota \underbrace{\int f(\boldsymbol{x}^{(t-1)})K^{(\iota)}(\boldsymbol{x}^{(t-1)},\boldsymbol{x}^{(t)})\,d\boldsymbol{x}^{(t-1)}}_{=f(\boldsymbol{x}^{(t)})} = f(\boldsymbol{x}^{(t)}).$$

Example 3.10. (One-at-a-time Metropolis–Hastings). *One example of a method using composite kernels is the so-called* ***one-at-a-time*** *Metropolis–Hastings algorithm. Consider the case of a p-dimensional random variable* $\boldsymbol{X} = (X_1, \ldots, X_p)$*. The Metropolis–Hastings Algorithms 3.3 and 3.4 update all components at a time. It can, however, be difficult to come up with a suitable proposal distribution* $q(\cdot \mid \boldsymbol{x}^{(t-1)})$ *(or g) for all variables. Alternatively, we could, as in the Gibbs sampler, update each component separately. For this we need p proposal distributions* $q_1, \ldots, q_p$ *for updating each of the* X_j*. The j-th proposal* q_j *(and thus the j-th kernel* $K^{(j)}$*) corresponds to updating the* X_j*.*
As mentioned above we can cycle deterministically through the kernels (corresponding to the kernel K°*), yielding the following algorithm. Starting with* $\boldsymbol{X}^{(0)} = (X_1^{(0)}, \ldots, X_p^{(0)})$ *iterate*

- *1.*
    - i. *Draw* $X_1 \sim q_1(\cdot \mid X_2^{(t-1)}, \ldots, X_p^{(t-1)})$.
    - ii. *Compute* $\alpha_1 = \min\left\{1, \frac{f(X_1,X_2^{(t-1)},\ldots,X_p^{(t-1)})\cdot q_1(X_1^{(t-1)}|X_1,X_2^{(t-1)},\ldots,X_p^{(t-1)})}{f(X_1^{(t-1)},X_2^{(t-1)},\ldots,X_p^{(t-1)})\cdot q_1(X_1|X_1^{(t-1)},X_2^{(t-1)},\ldots,X_p^{(t-1)})}\right\}$.
    - iii. *With probability* α_1 *set* $X_1^{(t)} = X_1$*, otherwise set* $X_1^{(t)} = X_1^{(t-1)}$.
- $\vdots$
- *j.*
    - i. *Draw* $X_j \sim q_j(\cdot \mid X_1^{(t)}, \ldots, X_{j-1}^{(t)}, X_j^{(t-1)}, \ldots, X_p^{(t-1)})$.
    - ii. *Compute* $\alpha_j = \min\left\{1, \frac{f(X_1^{(t)},\ldots,X_{j-1}^{(t)},X_j,X_{j+1}^{(t-1)},\ldots,X_p^{(t-1)})\cdot q_j(X_j^{(t-1)}|X_1^{(t)},\ldots,X_{j-1}^{(t)},X_j,X_{j+1}^{(t-1)},\ldots,X_p^{(t-1)})}{f(X_1^{(t)},\ldots,X_{j-1}^{(t)},X_j^{(t-1)},X_{j+1}^{(t-1)},\ldots,X_p^{(t-1)})\cdot q_j(X_j|X_1^{(t)},\ldots,X_{j-1}^{(t)},X_j^{(t-1)},X_{j+1}^{(t-1)},\ldots,X_p^{(t-1)})}\right\}$.
    - iii. *With probability* α_j *set* $X_j^{(t)} = X_j$*, otherwise set* $X_j^{(t)} = X_j^{(t-1)}$.
- $\vdots$
- *p.*
    - i. *Draw* $X_p \sim q_p(\cdot \mid X_1^{(t)}, \ldots, X_{p-1}^{(t)}, X_p^{(t-1)})$.
    - ii. *Compute* $\alpha_p = \min\left\{1, \frac{f(X_1^{(t)},\ldots,X_{p-1}^{(t)},X_p)\cdot q_p(X_p^{(t-1)}|X_1^{(t)},\ldots,X_{p-1}^{(t)},X_p)}{f(X_1^{(t)},\ldots,X_{p-1}^{(t)},X_p^{(t-1)})\cdot q_p(X_p|X_1^{(t)},\ldots,X_{p-1}^{(t)},X_p^{(t-1)})}\right\}$.
    - iii. *With probability* α_p *set* $X_p^{(t)} = X_p$*, otherwise set* $X_p^{(t)} = X_p^{(t-1)}$.

The corresponding random sweep algorithm (corresponding to K^{+}*) is: Starting with* $\boldsymbol{X}^{(0)} = (X_1^{(0)}, \ldots, X_p^{(0)})$ *iterate*

1. *Draw an index j from a distribution on* $\{1, \ldots, p\}$ *(e.g. uniform)*
2. *Draw* $X_j \sim q_j(\cdot \mid X_1^{(t-1)}, \ldots, X_p^{(t-1)})$.
3. *Compute* $\alpha_j = \min\left\{1, \frac{f(X_1^{(t-1)},\ldots,X_{j-1}^{(t-1)},X_j,X_{j+1}^{(t-1)},\ldots,X_p^{(t-1)})\cdot q_j(X_j^{(t-1)}|X_1^{(t-1)},\ldots,X_{j-1}^{(t-1)},X_j,X_{j+1}^{(t-1)},\ldots,X_p^{(t-1)})}{f(X_1^{(t-1)},\ldots,X_{j-1}^{(t-1)},X_j^{(t-1)},X_{j+1}^{(t-1)},\ldots,X_p^{(t-1)})\cdot q_j(X_j|X_1^{(t-1)},\ldots,X_{j-1}^{(t-1)},X_j^{(t-1)},X_{j+1}^{(t-1)},\ldots,X_p^{(t-1)})}\right\}$.
4. *With probability* α_j *set* $X_j^{(t)} = X_j$*, otherwise set* $X_j^{(t)} = X_j^{(t-1)}$.
5. *Set* $X_\iota^{(t)} := X_\iota^{(t-1)}$ *for all* $\iota \neq j$.

Note the similarity to the Gibbs sampler. Indeed, the Gibbs sampler is a special case of a one-at-a-time Metropolis–Hastings algorithm as the following remark shows.

Remark. The Gibbs sampler for a p-dimensional distribution is a special case of a one-at-a-time Metropolis–Hastings algorithm: the (systematic scan) Gibbs sampler (Algorithm 3.1 is a cycle of p kernels, whereas the random scan Gibbs sampler (Algorithm 3.2 is a mixture of these kernels. The proposal q_j corresponding to the j-th kernel consists of drawing $X_j^{(t)} \sim f_{X_j|X_{-j}}$. The corresponding probability of acceptance is uniformly equal to 1.

Proof. The update of the j-th component of the Gibbs sampler consists of sampling from $X_j \mid X_{-j}$, i.e. it has the proposal

$$q_j(x_j \mid \boldsymbol{x}^{(t-1)}) = f_{X_j|X_{-j}}(x_j \mid x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)}).$$

We obtain for the j-th kernel that

$$\begin{aligned}
&\frac{f(x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_j, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)}) q_j(x_j^{(t-1)} \mid x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_j, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)})}{f(x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_j^{(t-1)}, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)}) q_j(x_j \mid x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_j^{(t-1)}, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)})} \\
&= \frac{f(x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_j, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)}) f_{X_j|X_{-j}}(x_j^{(t-1)} \mid x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)})}{f(x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_j^{(t-1)}, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)}) f_{X_j|X_{-j}}(x_j \mid x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)})} \\
&= \frac{f(x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_j, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)}) \frac{f(x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_j^{(t-1)}, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)})}{f(x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)})}}{f(x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_j^{(t-1)}, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)}) \frac{f(x_1^{(t)}, \ldots, x_{j-1}^{(t)}, x_j, x_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)})}{f(x_1^{(t)}, \ldots, x_{j-1}^{(t)}, {}_{j+1}^{(t-1)}, \ldots, x_p^{(t-1)})}} \\
&= 1,
\end{aligned}$$

thus $\alpha_j \equiv 1$. □

As explained above, the composite kernels K^+ and K° have the invariant distribution f, if all kernels $K^{(j)}$ have f as invariant distribution. Similarly, it is sufficient for the irreducibility of the kernels K^+ and K° that all kernels $K^{(j)}$ are irreducible. This is however not a very useful condition, nor is it a necessary condition. Often, some of the kernels $K^{(j)}$ focus on certain subspaces, and thus cannot be irreducible for the entire space. The kernels $K^{(j)}$ corresponding to the Gibbs sampler are *not* irreducible themselves: the j-th Gibbs kernel $K^{(j)}$ only updates X_j, not the other X_ι ($\iota \neq j$).

3.4 Diagnosing Convergence

3.4.1 Practical considerations

The theory of Markov chains guarantees that a Markov chain that is irreducible and has invariant distribution f converges to the invariant distribution. The ergodic theorems allow for approximating expectations $\mathbb{E}_f\left[h(\boldsymbol{X})\right]$ by their corresponding empirical means

$$\frac{1}{T}\sum_{t=1}^{T} h(\boldsymbol{X}^{(t)}) \longrightarrow \mathbb{E}_f\left[h(\boldsymbol{X})\right]$$

using the *entire* chain. In practice, however, often only a subset of the chain $(\boldsymbol{X}^{(t)})_t$ is used:

Burn-in Depending on how $\boldsymbol{X}^{(0)}$ is chosen, the distribution of $(\boldsymbol{X}^{(t)})_t$ for small t might still be far from the stationary distribution f. Thus it might be beneficial to discard the first iterations $\boldsymbol{X}^{(t)}$, $t = 1, \ldots, T_0$. This early stage of the sampling process is often referred to as *burn-in* period. How large T_0 has to be chosen depends on how fast mixing the Markov chain $(\boldsymbol{X}^{(t)})_t$ is. Figure 3.16 illustrates the idea of a burn-in period.

Thinning Markov chain Monte Carlo methods typically yield a Markov chain with positive autocorrelation, i.e. $\rho(X_k^{(t)}, X_k^{(t+\tau)})$ is positive for small τ. This suggests building a subchain by only keeping every m-th value ($m > 1$), i.e. we consider a Markov chain $(\boldsymbol{Y}^{(t)})_t$ with $\boldsymbol{Y}^{(t)} = \boldsymbol{X}^{(m \cdot t)}$ instead of $(\boldsymbol{X}^{(t)})_t$. If the correlation $\rho(\boldsymbol{X}^{(t)}, \boldsymbol{X}^{(t+\tau)})$ decreases monotonically in τ, then

$$\rho(Y_k^{(t)}, Y_k^{(t+\tau)}) = \rho(X_k^{(t)}, X_k^{(t+m\cdot\tau)}) < \rho(X_k^{(t)}, X_k^{(t+\tau)}),$$

i.e. the thinned chain $(\boldsymbol{Y}^{(t)})_t$ exhibits less autocorrelation than the original chain $(\boldsymbol{X}^{(t)})_t$. Thus thinning can be seen as a technique for reducing the autocorrelation, however at the price of yielding a chain $(\boldsymbol{Y}^{(t)})_{t=1,\ldots\lfloor T/m \rfloor}$,

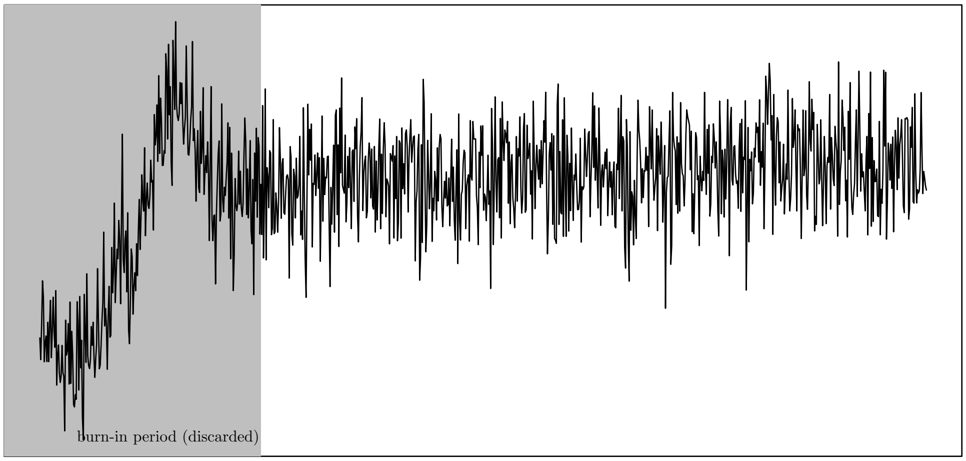

Figure 3.16: Illustration of the idea of a burn-in period.

whose length is reduced to $(1/m)$-th of the length of the original chain $(\boldsymbol{X}^{(t)})_{t=1,\dots,T}$. Even though thinning is very popular, it cannot be justified when the objective is estimating $\mathbb{E}_f\left[h(\boldsymbol{X})\right]$ [excepting special cases in which one can exploit specific features to save computations when working with the thinned chain; Owen (2017)] as the following lemma shows.

Lemma 3.3. Let $(\boldsymbol{X}^{(t)})_{t=1,\dots,T}$ be a the random variables obtained from a Markov chain at stationarity, with $\boldsymbol{X}^{(t)} \sim f$ and $(\boldsymbol{Y}^{(t)})_{t=1,\dots,\lfloor T/m \rfloor}$ a second sequence defined by $\boldsymbol{Y}^{(t)} := \boldsymbol{X}^{(m\cdot t)}$. If $\mathbb{V}\mathrm{ar}_f\left[h(\boldsymbol{X}^{(t)})\right] < \infty$, then

$$\mathbb{V}\mathrm{ar}\left[\frac{1}{T}\sum_{t=1}^{T} h(\boldsymbol{X}^{(t)})\right] \leq \mathbb{V}\mathrm{ar}\left[\frac{1}{\lfloor T/m \rfloor}\sum_{t=1}^{\lfloor T/m \rfloor} h(\boldsymbol{Y}^{(t)})\right].$$

Proof. To simplify the proof we assume that T is divisible by m, i.e. $T/m \in \mathbb{N}$. Using

$$\sum_{t=1}^{T} h(\boldsymbol{X}^{(t)}) = \sum_{\tau=0}^{m-1}\sum_{t=1}^{T/m} h(\boldsymbol{X}^{(t\cdot m+\tau)})$$

and

$$\mathbb{V}\mathrm{ar}\left[\sum_{t=1}^{T/m} h(\boldsymbol{X}^{(t\cdot m+\tau_1)})\right] = \mathbb{V}\mathrm{ar}\left[\sum_{t=1}^{T/m} h(\boldsymbol{X}^{(t\cdot m+\tau_2)})\right]$$

for $\tau_1, \tau_2 \in \{0, \dots, m-1\}$, we obtain that

$$\begin{aligned}\mathbb{V}\mathrm{ar}\left[\sum_{t=1}^{T} h(\boldsymbol{X}^{(t)})\right] &= \mathbb{V}\mathrm{ar}\left[\sum_{\tau=0}^{m-1}\sum_{t=1}^{T/m} h(\boldsymbol{X}^{(t\cdot m+\tau)})\right] \\ &= m\cdot\mathbb{V}\mathrm{ar}\left[\sum_{t=1}^{T/m} h(\boldsymbol{X}^{(t\cdot m)})\right] + \sum_{\eta\neq\tau=0}^{m-1}\underbrace{\mathbb{C}ov\left[\sum_{t=1}^{T/m} h(\boldsymbol{X}^{(t\cdot m+\eta)}), \sum_{t=1}^{T/m} h(\boldsymbol{X}^{(t\cdot m+\tau)})\right]}_{\leq \mathbb{V}\mathrm{ar}\left[\sum_{t=1}^{T/m} h(\boldsymbol{X}^{(t\cdot m)})\right]} \\ &\leq m^2\cdot\mathbb{V}\mathrm{ar}\left[\sum_{t=1}^{T/m} h(\boldsymbol{X}^{(t\cdot m)})\right] = m^2\cdot\mathbb{V}\mathrm{ar}\left[\sum_{t=1}^{T/m} h(\boldsymbol{Y}^{(t)})\right].\end{aligned}$$

Thus

$$\mathbb{V}\text{ar}\left[\frac{1}{T}\sum_{t=1}^{T} h(\boldsymbol{X}^{(t)})\right] = \frac{1}{T^2}\mathbb{V}\text{ar}\left[\sum_{t=1}^{T} h(\boldsymbol{X}^{(t)})\right] \leq \frac{m^2}{T^2}\mathbb{V}\text{ar}\left[\sum_{t=1}^{T/m} h(\boldsymbol{Y}^{(t)})\right] = \mathbb{V}\text{ar}\left[\frac{1}{T/m}\sum_{t=1}^{T/m} h(\boldsymbol{Y}^{(t)})\right].$$

□

The concept of thinning can be useful for other reasons. If storage is limited it may not be possible to store all of an arbitrarily long chain; in this context, it can be much better to store the thinned skeleton of a long chain than to consider the entire sample path of a shorter chain. Furthermore, it can be easier to assess the convergence of the thinned chain $(\boldsymbol{Y}^{(t)})_t$ as opposed to entire chain $(\boldsymbol{X}^{(t)})_t$.

3.4.2 Tools for monitoring convergence

Although the theory presented in the preceding chapters guarantees the convergence of the Markov chains to the required distributions, this does not imply that a *finite* sample from such a chain yields a good approximation to the target distribution. As with all approximating methods this must be confirmed in practice.

This section tries to give a brief overview over various approaches to diagnosing convergence. A more detailed review with many practical examples can be found in (Guihennec-Jouyaux, Mengersen, and Robert 1998) or (Robert and Casella 2004, chap. 12). There are numerous R packages (`CODA` is well known; `mcmcse` provides useful tools for assessing some aspects of performance) that provides a vast selection of tools for diagnosing convergence.

Diagnosing convergence is an art. The techniques presented in the following are no more than exploratory tools that help you judge whether the chain has reached its stationary regime. This section contains several cautionary examples where the different tools for diagnosing convergence fail.

Broadly speaking, convergence assessment can be split into the following three categories, each of which considers the assessment of a different aspect of convergence:

Convergence to the target distribution. The first, and most important, question is whether $(\boldsymbol{X}^{(t)})_t$ yields a sample from the target distribution? In order to answer this question we need to assess ...

- whether $(\boldsymbol{X}^{(t)})_t$ has reached a stationary regime, and
- whether $(\boldsymbol{X}^{(t)})_t$ covers the entire support of the target distribution.

Convergence of the averages. Does $\sum_{t=1}^{T} h(\boldsymbol{X}^{(t)})/T$ provide a good approximation to the expectation $\mathbb{E}_f\left[h(\boldsymbol{X})\right]$ under the target distribution?

Comparison to i.i.d. sampling. How much information is contained in the sample from the Markov chain compared to i.i.d. sampling?

3.4.3 Basic plots

The most basic approach to diagnosing the output of a Markov Chain Monte Carlo algorithm is to plot the sample path $(\boldsymbol{X}^{(t)})_t$. Note that the convergence of $(\boldsymbol{X}^{(t)})_t$ is in distribution, i.e. the sample path is *not* supposed to converge to a single value. Ideally, the plot should be oscillating very fast and show very little structure or trend. In general terms, the smoother such a plot seems, the slower the mixing of the associated chain.

Note however that this plot suffers from the "you've only seen where you've been" problem. It is impossible to see from a plot of the sample path whether the chain has explored the entire support of the distribution (without additional information).

Example 3.11. (A simple mixture of two Gaussians). *Consider sampling from a mixture of two well-separated Gaussians*

$$f(x) = 0.4 \cdot \phi_{(-1,0.2^2)}(x) + 0.6 \cdot \phi_{(2,0.3^2)}(x)$$

(see Figure 3.17 for a plot of the density) using a random walk Metropolis algorithm with an $\mathsf{N}(0, \mathbb{V}ar\left[\varepsilon\right])$ increment distribution. If we choose the proposal variance $\mathbb{V}ar\left[\varepsilon\right]$ too small, we only sample from one component of the mixture, not from the mixture itself. Figures 3.17–3.19 show the sample paths for two choices of $\mathbb{V}ar\left[\varepsilon\right]$: $\mathbb{V}ar\left[\varepsilon\right] = 0.4^2$ and $\mathbb{V}ar\left[\varepsilon\right] = 1.2^2$. The first choice of $\mathbb{V}ar\left[\varepsilon\right]$ is too small: the chain is very likely to remain in one of the two modes of

the distribution. Note that it is impossible to tell from Figure 3.18 alone that the chain has not explored the entire support of the target.

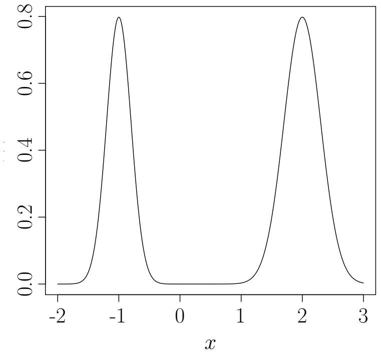

Figure 3.17: Density of a mixture distribution $f(x)$.

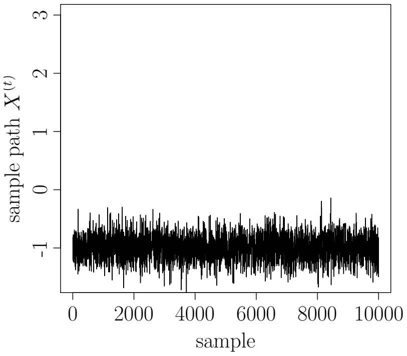

Figure 3.18: Sample path of a random walk Metropolis algorithm with proposal variance 0.4^2 and mixture distribution target $f(x)$.

In order to diagnose the convergence of sample averages, one can look at a plot of the cumulative averages $(\sum_{\tau=1}^{t} h(X^{(\tau)})/t)_t$. Note that the convergence of the cumulative averages is—as the ergodic theorems suggest—to a value $(\mathbb{E}_f [h(\boldsymbol{X})])$. Figure 3.13 shows plots of the cumulative averages. An alternative to plotting the cumulative means is using the so-called CUSUMs

$$\sum_{\tau=1}^{t} \left[h(X_j^{(\tau)}) - \frac{1}{T} \sum_{i=1}^{T} h(X_j^{(i)}) \right],$$

which is none other than the difference between the cumulative sums and the corresponding (pro-rated) estimate of the limit $\mathbb{E}_f [h(\boldsymbol{X})]$. For a fast mixing chain (for which increments might be approximately i.i.d. Normal, say), a CUSUM plot should roughly resemble a Brownian bridge from 0 to 0, i.e. being highly irregular and centred around 0. Slow mixing chains exhibit long excursions away from 0.

Example 3.12. (A pathological generator for the Beta distribution). *The following MCMC algorithm (for details, see Robert and Casella 2004 Problem 7.5) yields a sample from the* $\textsf{Beta}(\alpha, 1)$ *distribution. Starting with any* $X^{(0)}$ *iterate for* $t = 1, 2, \ldots$

1. *With probability* $1 - X^{(t-1)}$, *set* $X^{(t)} = X^{(t-1)}$.
2. *Otherwise draw* $X^{(t)} \sim \textsf{Beta}(\alpha + 1, 1)$.

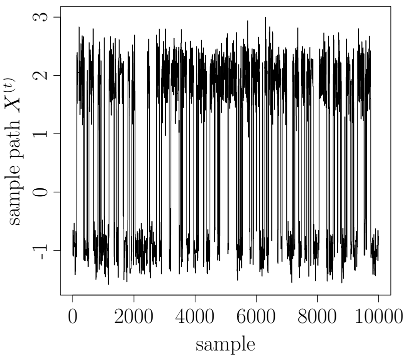

Figure 3.19: Sample path of a random walk Metropolis algorithm with proposal variance 1.2^2 and mixture distribution target.

This algorithm yields a very slowly converging Markov chain, to which no central limit theorem applies. This slow convergence can be seen in a plot of the cumulative means (Figure 3.20–3.21); $\alpha = 1$ *so* $\mathbb{E}[X] = 1/2$*).*

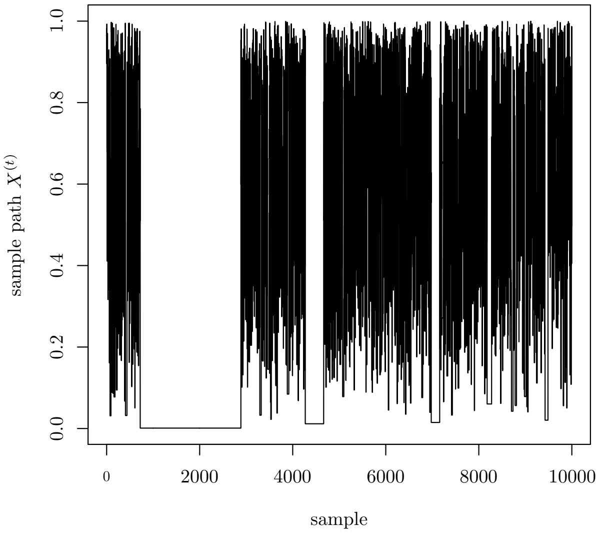

Figure 3.20: Sample path $X^{(t)}$ obtained for the pathological Beta generator, $\alpha = 1$.

Note that it is impossible to tell from a plot of the cumulative means whether the Markov chain has explored the entire support of the target distribution.

3.4.4 Non-parametric tests of stationarity

A variety of nonparametric tests can be employed to establish whether the samples from a Markov chain behave in particular ways. This section presents an illustration of the (informal, approximate) use of the Kolmogorov–Smirnov test to assess whether there is evidence that a Markov chain has not yet reached stationarity.

In its simplest version, it is based on splitting the chain into three parts: $(\boldsymbol{X}^{(t)})_{t=1,\ldots,\lfloor T/3 \rfloor}$,$(\boldsymbol{X}^{(t)})_{t=\lfloor T/3 \rfloor+1,\ldots,2\lfloor T/3 \rfloor}$, and $(\boldsymbol{X}^{(t)})_{t=2\lfloor T/3 \rfloor+1,\ldots,T}$. The first block is considered to be the burn-in period. If the Markov chain has reached its stationary regime after $\lfloor T/3 \rfloor$ iterations, the second and third block should be from the same distribution. Thus

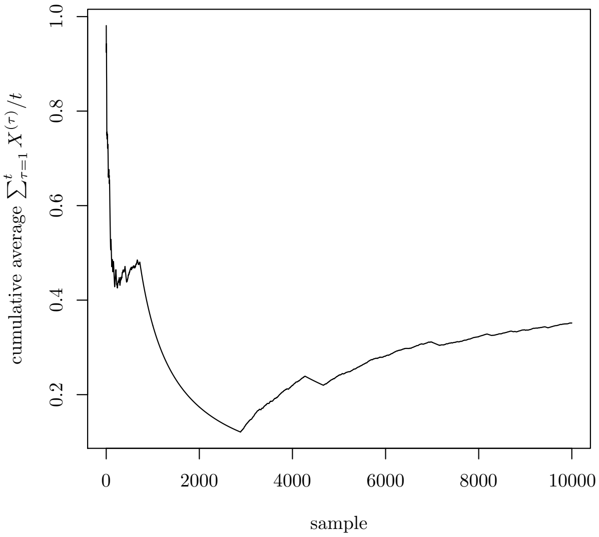

Figure 3.21: Cumulative mean $\sum_{\tau=1}^{t} X^{(\tau)}/t$ obtained for the pathological Beta generator, $\alpha = 1$.

we should be able to tell whether the chain has converged by comparing the distribution of $(\boldsymbol{X}^{(t)})_{t=\lfloor T/3\rfloor+1,\ldots,2\lfloor T/3\rfloor}$ to the distribution of $(\boldsymbol{X}^{(t)})_{t=2\lfloor T/3\rfloor+1,\ldots,T}$ using suitable nonparametric two-sample tests. One such test is the Kolmogorov–Smirnov test.

Definition 3.4. (Kolmogorov–Smirnov Statistic). *The two-sample Kolmogorov–Smirnov test for comparing two i.i.d. samples $Z_{1,1},\ldots,Z_{1,n}$ and $Z_{2,1},\ldots,Z_{2,n}$ is based on comparing their empirical CDFs*

$$\hat{F}_k(z) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{I}_{(-\infty,z]}(Z_{k,i}).$$

The ***Kolmogorov–Smirnov test statistic*** *is the maximum difference between the two empirical CDFs:*

$$K = \sup_{z\in\mathbb{R}} |\hat{F}_1(z) - \hat{F}_2(z)|.$$

For $n \to \infty$ the CDF of $\sqrt{n}\cdot K$ converges to the CDF

$$R(k) = 1 - \sum_{i=1}^{\infty} (-1)^{i-1}\exp(-2i^2k^2).$$

As the Kolmogorov–Smirnov test is designed for i.i.d. samples, we do not apply it to the $(\boldsymbol{X}^{(t)})_t$ directly, but to a thinned chain $(\boldsymbol{Y}^{(t)})_t$ with $\boldsymbol{Y}^{(t)} = \boldsymbol{X}^{(m\cdot t)}$: the thinned chain is less correlated and thus closer to being an i.i.d. sample. This, of course, still formally violates the conditions under which the Kolmogorov–Smirnov test is exact but one can still hope to obtain useful information when the conditions are *close* to being satisfied.

We can now use the Kolmogorov–Smirnov statistics to compare the distribution of the second block, $(\boldsymbol{Y}^{(t)})_{t=\lfloor T/(3m)\rfloor+1,\ldots,2\lfloor T/(3m)\rfloor}$, with that of the third, $(\boldsymbol{Y}^{(t)})_{t=2\lfloor T/(3m)\rfloor+1,\ldots,\lfloor T/m\rfloor}$:

$$K = \sup_{x\in\mathbb{R}} \left| \widehat{F}_{(\boldsymbol{Y}^{(t)})_{t=\lfloor T/(3m)\rfloor+1,\ldots,2\lfloor T/(3m)\rfloor}}(x) - \widehat{F}_{(\boldsymbol{Y}^{(t)})_{t=2\lfloor T/(3m)\rfloor+1,\ldots,\lfloor T/m\rfloor}}(x) \right|.$$

As the thinned chain is not an i.i.d. sample, we cannot use the Kolmogorov–Smirnov test as a formal statistical test (besides, we would run into problems of multiple testing). However, we can use it as an informal tool by monitoring the standardised statistic $\sqrt{t}K_t$ as a function of $t.$, where K_t denotes the Kolmogorov–Smirnov statistic obtained from the sample consisting of the first t observations only. If a significant proportion of the values of this standardised statistic are above the corresponding quantile of the asymptotic distribution, it is safe to assume that the chain has not yet reached its stationary regime.

Example 3.13. (Gibbs sampling from a bivariate Gaussian (continued)). *In this example we consider sampling from a bivariate Gaussian distribution, once with $\rho(X_1, X_2) = 0.3$ and once with $\rho(X_1, X_2) = 0.99$. The former leads to a fast mixing chain, the latter to a very slowly mixing chain. Figure 3.22–3.23 shows the plots of the standardised Kolmogorov–Smirnov statistic. It suggests that the sample size of 10,000 is large enough for the low-correlation setting, but not large enough for the high-correlation setting.*

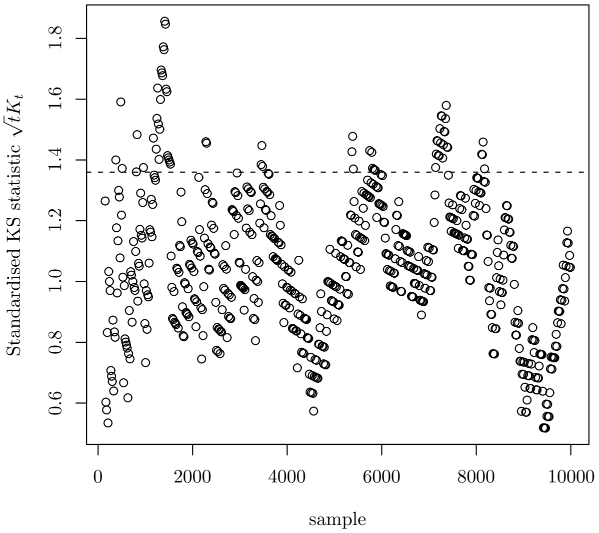

Figure 3.22: Standardised Kolmogorov–Smirnov statistic for $X_1^{(5\cdot t)}$ from the Gibbs sampler from the bivariate Gaussian, $\rho(X_1, X_2) = 0.3$.

Note that this use of the Kolmogorov–Smirnov test suffers from the "you've only seen where you've been" problem, as it is based on comparing $(\boldsymbol{Y}^{(t)})_{t=\lfloor T/(3m)\rfloor+1,\ldots,2\lfloor T/(3m)\rfloor}$ and $(\boldsymbol{Y}^{(t)})_{t=2\lfloor T/(3m)\rfloor+1,\ldots,\lfloor T/m\rfloor}$. A plot of the Kolmogorov–Smirnov statistic for the chain with $\mathbb{V}\text{ar}[\varepsilon] = 0.4$ from Example 3.11 would not reveal anything unusual.

3.4.5 Riemann sums and control variates

A simple tool for diagnosing convergence of a one-dimensional Markov chain can be based on the fact that

$$\int_E f(x)\, dx = 1.$$

We can estimate this integral using the Riemann sum

$$\sum_{t=2}^{T} (X^{[t]} - X^{[t-1]}) f(X^{[t]}), \tag{3.16}$$

where $X^{[1]} \leq \cdots \leq X^{[T]}$ is the ordered sample from the Markov chain. If the Markov chain has explored all the support of f, then (3.16) should be around 1 (as it is an estimate of the integral of a probability density). Note that this method, often referred to as Riemann sums (Philippe and Robert 2001), requires that the density f is known inclusive of normalisation constants (and thus to apply this technique to a univariate marginal of a multivariate problem would require that at least one univariate marginal of the target distributions is known exactly).

Example 3.14. (A simple mixture of two Gaussians (continued)). *In Example 3.11 we considered two random-walk Metropolis algorithms: one ($\mathbb{V}ar[\varepsilon] = 0.4^2$) failed to explore the entire support of the target distribution, whereas the*

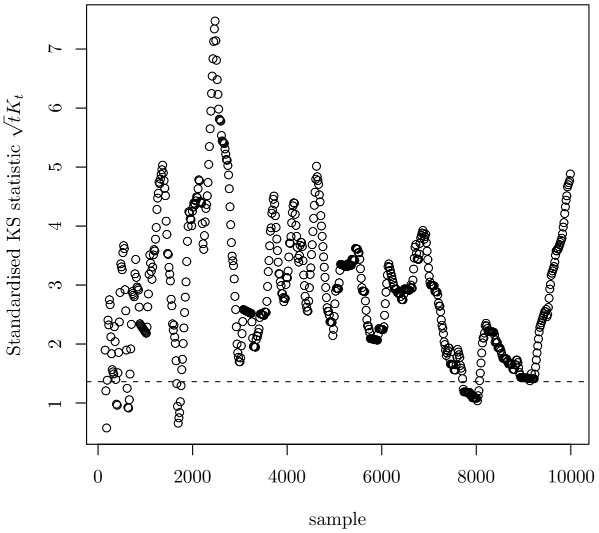

Figure 3.23: Standardised Kolmogorov–Smirnov statistic for $X_1^{(5\cdot t)}$ from the Gibbs sampler from the bivariate Gaussian, $\rho(X_1, X_2) = 0.99$.

other one ($\mathbb{V}ar\left[\varepsilon\right] = 1.2^2$*) managed to. The corresponding Riemann sums are* 0.598 *and* 1.001*, clearly indicating that the first algorithm does not explore the entire support.*

Riemann sums can be seen as a special case of a technique called *control variates.* The idea of control variates is essentially to compare several ways of estimating the same quantity using the same collection of samples. If the different estimates disagree, the chain has not yet converged. Note that the technique of control variates is only useful if the different estimators converge about as fast as the quantity of interest—otherwise we would obtain an overly optimistic, or an overly conservative estimate of whether the chain has converged. In the special case of the Riemann sum we compare two quantities: the constant 1 and the Riemann sum (3.16).

3.4.6 Comparing multiple chains

A family of convergence diagnostics (see e.g. Gelman and Rubin 1992; Brooks and Gelman 1998) is based on running $L > 1$ chains—which we will denote by $(\boldsymbol{X}^{(1,t)})_t, \dots, (\boldsymbol{X}^{(L,t)})_t$—with overdispersed starting values $\boldsymbol{X}^{(1,0)}, \dots, \boldsymbol{X}^{(L,0)}$ (in the sense that the variance of the starting values should be larger than the variance of the target distribution). These starting values should in principle be chosen to give reasonable coverage of the support of the target distribution.

All L chains should converge to the same distribution, so comparing the plots described in Section @ref(#seccdplots) for the L different chains should not reveal any difference. A more formal approach to diagnosing whether the L chains are all from the same distribution can be based on comparing the inter-quantile distances.

We can estimate the inter-quantile distances in two ways. The first consists of estimating the inter-quantile distance for each of the L chains and averaging over these results, i.e. our estimate is $\sum_{l=1}^{L} \delta_\gamma^{(l)}/L$, where $\delta_\gamma^{(l)}$ is the distance between the γ and $(1-\gamma)$ quantile of the l-th chain $(X_k^{(l,t)})_t$. Alternatively, we can pool the data first, and then compute the distance, $\hat{\delta}_\gamma$, between the γ and $(1-\gamma)$ quantile of the pooled data. If all chains are a sample from the same distribution, both estimates should be roughly the same, so their ratio

$$\hat{S}_\gamma^{\text{interval}} = \frac{\sum_{l=1}^{L} \delta_\gamma^{(l)}/L}{\hat{\delta}_\gamma}$$

can be used as a tool to diagnose whether all chains sampled from the same distribution, in which case the ratio should be around 1.

Alternatively, one could compare the variances within the L chains to the pooled estimate of the variance (see Brooks and Gelman 1998 for more details).

Example 3.15. (A simple mixture of two Gaussians (continued)). *In the example of the mixture of two Gaussians we will consider $L = 8$ chains initialised with iid samples from a $\mathsf{N}\left(0, 10^2\right)$ distribution. Figure 3.24–3.25 shows the sample paths of the 8 chains for both choices of $\mathbb{V}ar\left[\varepsilon\right]$. The corresponding values of $\hat{S}_{0.05}^{interval}$ are:*

$$\mathbb{V}ar\left[\varepsilon\right] = 0.4^2 : \hat{S}_{0.05}^{interval} = \frac{0.9789992}{3.630008} = 0.2696962$$
$$\mathbb{V}ar\left[\varepsilon\right] = 1.2^2 : \hat{S}_{0.05}^{interval} = \frac{3.634382}{3.646463} = 0.996687.$$

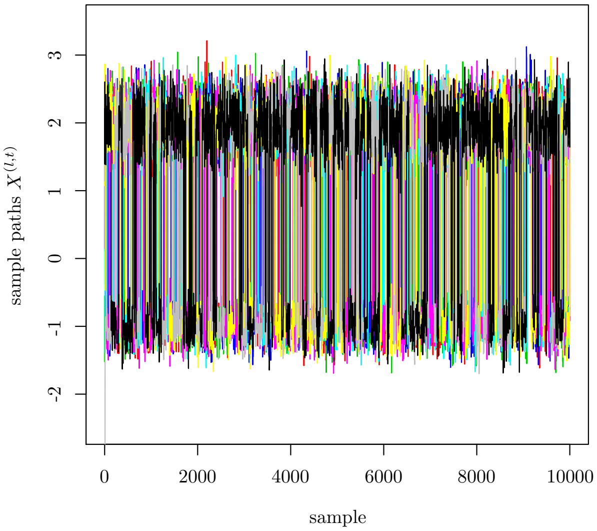

Figure 3.24: Comparison of the sample paths for $L = 8$ chains for the mixture of two Gaussians: $\mathbb{V}\text{ar}\left[\varepsilon\right] = 0.4^2$.

Note that this method depends crucially on the choice of initial values $\boldsymbol{X}^{(1,0)}, \ldots, \boldsymbol{X}^{(L,0)}$, and thus can easily fail, as the following example shows.

Example 3.16. (Witch's hat distribution). *Consider a distribution with the following density:*

$$f(x_1, x_2) \propto \begin{cases} (1-\delta)\phi_{(\boldsymbol{\mu},\sigma^2\cdot\mathbb{I})}(x_1, x_2) + \delta & \text{if } x_1, x_2 \in (0,1), \\ 0 & \text{otherwise,} \end{cases}$$

which is a mixture of a Gaussian and a uniform distribution, both truncated to $(0,1) \times (0,1)$. Figure 3.26 illustrates the density. For very small σ^2, the Gaussian component is concentrated in a very small area around $\boldsymbol{\mu}$.

The conditional distribution of $X_1 \mid X_2$ is

$$f(x_1 \mid x_2) = \begin{cases} (1-\delta_{x_2})\phi_{(\boldsymbol{\mu},\sigma^2\cdot\mathbb{I})}(x_1, x_2) + \delta_{x_2} & \text{for } x_1 \in (0,1) \\ 0 & \text{otherwise} \end{cases}$$

with $\delta_{x_2} = \dfrac{\delta}{\delta + (1-\delta)\phi_{(\mu_2,\sigma^2)}(x_2)}$.

Assume we want to estimate $\mathbb{P}\left(0.49 < X_1, X_2 \leq 0.51\right)$ for $\delta = 10^{-3}$, $\boldsymbol{\mu} = (0.5, 0.5)'$, and $\sigma = 10^{-5}$ using a Gibbs sampler. Note that 99.9% of the mass of the distribution is concentrated in a very small area around $(0.5, 0.5)$, i.e. $\mathbb{P}\left(0.49 < X_1, X_2 \leq 0.51\right) \approx 0.999$.

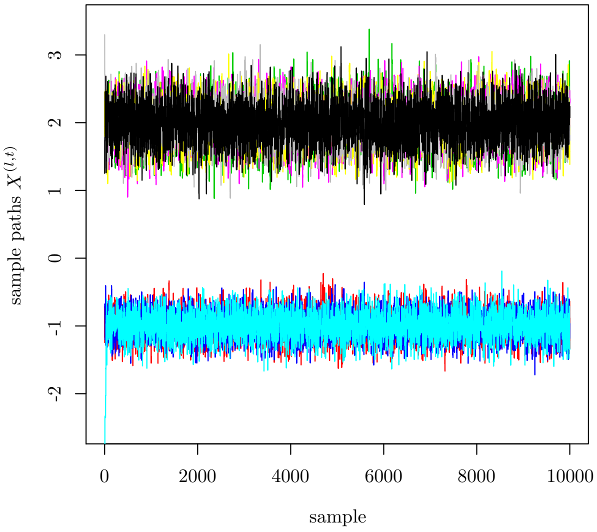

Figure 3.25: Comparison of the sample paths for $L = 8$ chains for the mixture of two Gaussians: $\mathbb{V}\text{ar}\left[\varepsilon\right] = 1.2^2$.

Nonetheless, it is very unlikely that the Gibbs sampler visits this part of the distribution. This is due to the fact that unless x_2 (or x_1) is very close to μ_2 (or μ_1), δ_{x_2} (or δ_{x_1}) is almost 1, i.e. the Gibbs sampler only samples from the uniform component of the distribution. Figure 3.27 shows the samples obtained from 15 runs of the Gibbs sampler (first 100 iterations only) all using different initialisations. On average only 0.04% of the sampled values lie in $(0.49, 0.51) \times (0.49, 0.51)$ *yielding an estimate of* $\hat{\mathbb{P}}(0.49 < X_1, X_2 \leq 0.51) = 0.0004$ *(as opposed to* $\mathbb{P}(0.49 < X_1, X_2 \leq 0.51) = 0.999$*).*

It is however close to impossible to detect this problem with any technique based on multiple initialisations. The Gibbs sampler shows this behaviour for practically all starting values. In Figure 3.27 all 15 starting values yield a Gibbs sampler that is stuck in the "brim" of the witch's hat and thus misses 99.9% of the probability mass of the target distribution.

3.4.7 Comparison to i.i.d. sampling and the effective sample size

MCMC algorithms typically yield a positively correlated sample $(\boldsymbol{X}^{(t)})_{t=1,\ldots,T}$, which contains less information than an i.i.d. sample of size T. If the $(\boldsymbol{X}^{(t)})_{t=1,\ldots,T}$ are positively correlated, then the variance of the average

$$\mathbb{V}\text{ar}\left[\frac{1}{T}\sum_{t=1}^{T} h(\boldsymbol{X}^{(t)})\right] \tag{3.17}$$

is larger than the variance we would obtain from an i.i.d. sample, which is $\mathbb{V}\text{ar}\left[h(\boldsymbol{X}^{(t)})\right]/T$.

The effective sample size (ESS) attempts to quantify the loss of information caused by this positive correlation. The effective sample size is the size an i.i.d. sample would have to have in order to obtain the same variance (3.17) as the estimate from the Markov chain $(\boldsymbol{X}^{(t)})_{t=1,\ldots,T}$.

As the exact computation of this quantity is generally impossible, a number of simplifying approximations are often made in order to obtain a computationally tractable proxy for this quantity. Slightly confusingly, the *approximate* equivalent independent sample size arrived at following this chain of approximations is also referred to as the ESS. More sophisticated approximations are used to obtain better approximations of the ESS in some settings, but we focus here on one simple and widely-applicable option.

In order to compute the variance (3.17) we make the simplifying assumption that $(h(\boldsymbol{X}^{(t)}))_{t=1,\ldots,T}$ is from a

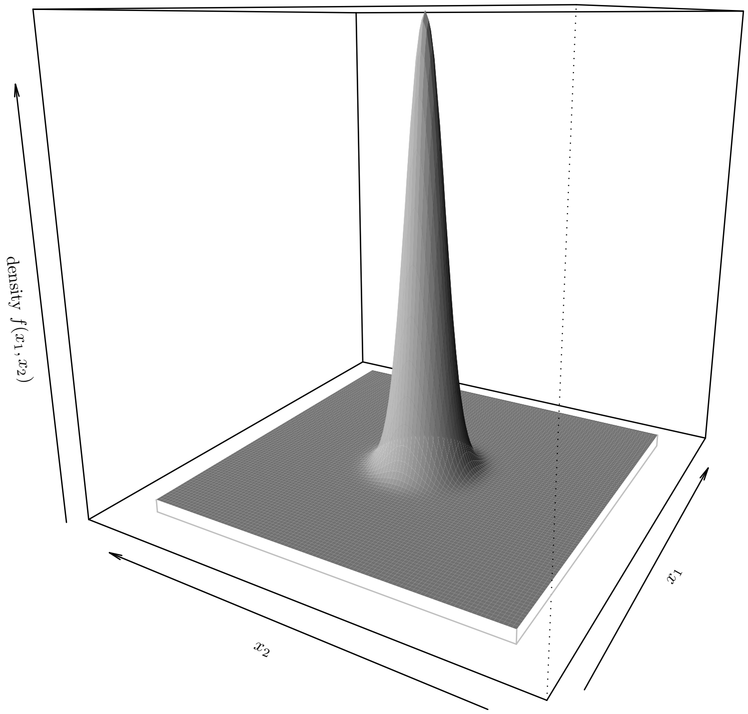

Figure 3.26: Witch's hat distribution: Density for $\delta = 0.2$, $\boldsymbol{\mu} = (0.5, 0.5)'$, and $\sigma = 0.05$

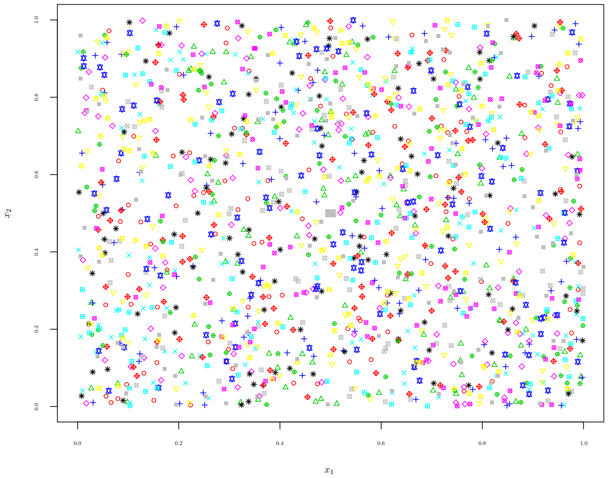

Figure 3.27: Witch's hat distribution: First 100 values from 15 samples using different starting values. $\delta = 10^{-3}$, $\boldsymbol{\mu} = (0.5, 0.5)'$, and $\sigma = 10^{-5}$.

second-order stationary time series, i.e. $\mathbb{V}\text{ar}\left[h(\boldsymbol{X}^{(t)})\right] = \sigma^2$, and $\rho(h(\boldsymbol{X}^{(t)}), h(\boldsymbol{X}^{(t+\tau)})) = \rho(\tau)$. Then

$$\begin{aligned}\mathbb{V}\text{ar}\left[\frac{1}{T}\sum_{t=1}^{T} h(\boldsymbol{X}^{(t)})\right] &= \frac{1}{T^2}\left(\sum_{t=1}^{T}\underbrace{\mathbb{V}\text{ar}\left[h(\boldsymbol{X}^{(t)})\right]}_{=\sigma^2} + 2\sum_{1\le s<t\le T}\underbrace{\mathbb{C}ov\left[h(\boldsymbol{X}^{(s)}), h(\boldsymbol{X}^{(t)})\right]}_{=\sigma^2\cdot\rho(t-s)}\right)\\ &= \frac{\sigma^2}{T^2}\left(T + 2\sum_{\tau=1}^{T-1}(T-\tau)\rho(\tau)\right) = \frac{\sigma^2}{T}\left(1 + 2\sum_{\tau=1}^{T-1}\left(1-\frac{\tau}{T}\right)\rho(\tau)\right).\end{aligned}$$

If $\sum_{\tau=1}^{\infty}|\rho(\tau)| < \infty$, then we can obtain from the dominated convergence theorem (see e.g. Brockwell and Davis 1991 Theorem 7.1.1 for details) that

$$T\cdot\mathbb{V}\text{ar}\left[\frac{1}{T}\sum_{t=1}^{T} h(\boldsymbol{X}^{(t)})\right] \longrightarrow \sigma^2\left(1 + 2\sum_{\tau=1}^{\infty}\rho(\tau)\right)$$

as $T\to\infty$. Note that the variance of the simple Monte Carlo estimate of $\mathbb{E}_f\left[h(X)\right]$ would be σ^2/T_{ESS} if we were to use an i.i.d. sample of size T_{ESS}. We can now obtain the effective sample size T_{ESS} by equating these two variances and solving for T_{ESS}, yielding

$$T_{\text{ESS}} = \frac{1}{1+2\sum_{\tau=1}^{\infty}\rho(\tau)}\cdot T.$$

If we assume that $(h(\boldsymbol{X}^{(t)}))_{t=1,\ldots,T}$ is a first-order autoregressive time series (AR(1)), i.e. $\rho(\tau) = \rho(h(\boldsymbol{X}^{(t)}), h(\boldsymbol{X}^{(t+\tau)})) = \rho^{|\tau|}$, then we obtain using $1 + 2\sum_{\tau=1}^{\infty}\rho^{\tau} = (1+\rho)/(1-\rho)$ that

$$T_{\text{ESS}} = \frac{1-\rho}{1+\rho}\cdot T.$$

Example 3.17. (Gibbs sampling from a bivariate Gaussian (continued)). *In examples 3.4 and 3.5 we obtained for the low-correlation setting that $\rho(X_1^{(t-1)}, X_1^{(t)}) = 0.078$, thus the effective sample size is*

$$T_{ESS} = \frac{1-0.078}{1+0.078}\cdot 10000 = 8547.$$

For the high-correlation setting we obtained $\rho(X_1^{(t-1)}, X_1^{(t)}) = 0.979$, thus the effective sample size is considerably smaller:

$$T_{ESS} = \frac{1-0.979}{1+0.979}\cdot 10000 = 105.$$

Note that there are other more sophisticated ways of estimating ESS which do not require this crude autoregressive approximation. The `mcmcse` package provides an implementation of the method of Gong and Flegal (2016) which has somewhat better properties than the method employed by `coda`.

3.5 Optimisation with MCMC

So far we have studied various methods that allow for approximating expectations $\mathbb{E}\left[h(\boldsymbol{X})\right]$ by ergodic averages $\frac{1}{T}\sum_{t=1}^{T} h(\boldsymbol{X}_i^{(t)})$. This section presents an algorithm for finding the (global) mode(s) of a distribution. For definiteness, in this chapter we define the mode(s) of a distribution to be the set of global maxima of the density, i.e. $\{\boldsymbol{\xi} : f(\boldsymbol{\xi}) \ge f(\boldsymbol{x})\ \forall \boldsymbol{x}\}$. In Section 3.5.1 we will extend this idea to finding global extrema of arbitrary functions.

We could estimate the mode of a distribution by the $\boldsymbol{X}^{(t)}$ with maximal density $f(\boldsymbol{X}^{(t)})$; this is however a not very efficient strategy. A sample from a Markov chain with invariant distribution $f(\cdot)$ samples from the whole distribution and not only from the mode(s).

This suggests modifying the distribution such that it is more concentrated around the mode(s). One way of achieving this is to consider

$$f_{(\beta)}(x) \propto (f(x))^{\beta}$$

for very large values of β.

Example 3.18. (Normal distribution). *Consider the $\mathsf{N}\left(\mu, \sigma^2\right)$ distribution with density*

$$f_{(\mu,\sigma^2)}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \propto \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

It is easy to see that the mode of the $\mathsf{N}\left(\mu, \sigma^2\right)$ distribution is μ. We have that

$$\left(f_{(\mu,\sigma^2)}(x)\right)^\beta \propto \left(\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)\right)^\beta = \exp\left(-\frac{(x-\mu)^2}{2\sigma^2/\beta}\right) \propto f_{(\mu,\sigma^2/\beta)}(x).$$

In other words, the larger β is chosen, the more concentrated the distribution will be around the mode μ. Figure 3.28 illustrates this idea.

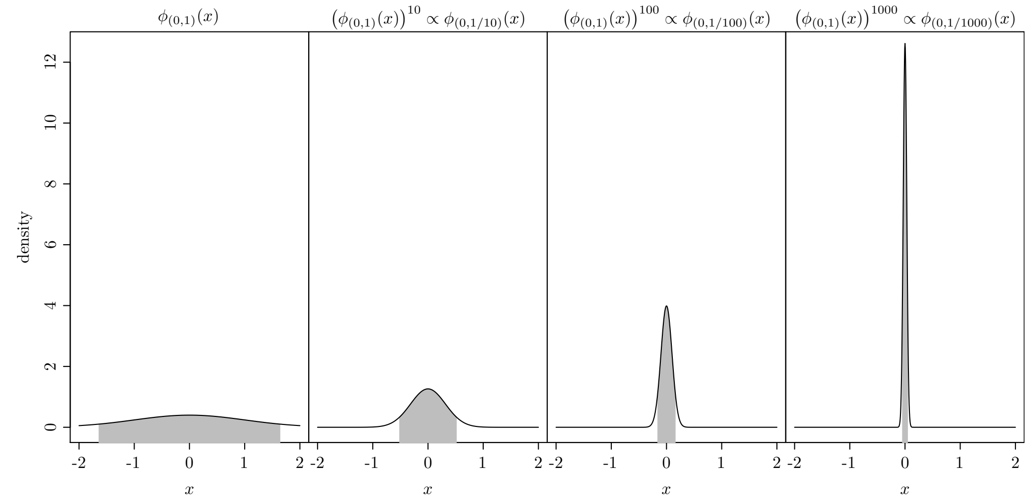

Figure 3.28: Density of the $\mathsf{N}\left(0, 1\right)$ raised to increasing powers. The areas shaded in grey represent 90% of the probability mass.

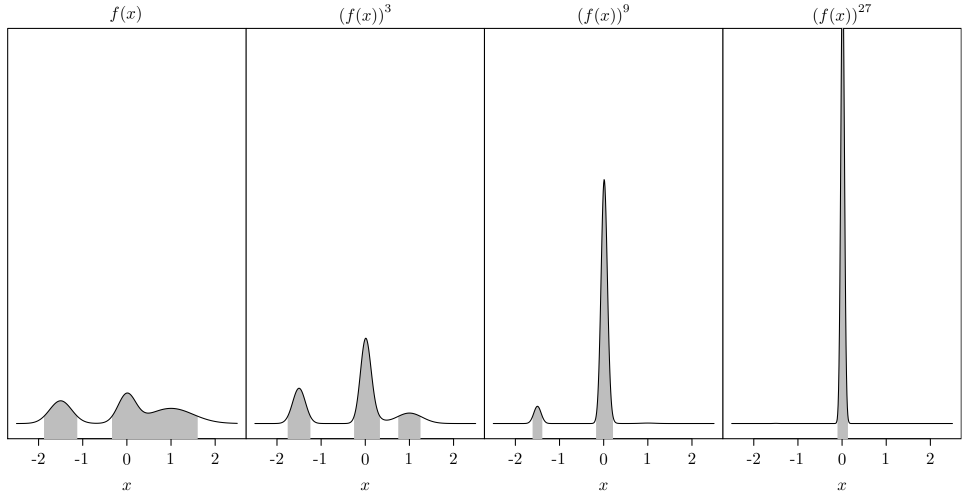

Figure 3.29: An arbitrary multimodal density raised to increasing powers. The areas shaded in grey reach from the 5% to the 95% quantiles.

The result we have obtained for the Gaussian distribution in the above example actually holds in general. For $\beta \to \infty$ the distribution defined by the density $f_{(\beta)}(x)$ converges to a distribution that has all mass on the mode(s) of f (see Figure 3.29 for an example). It is instructive to see informally why this is the case when considering a discrete random variable with probability density function $p(\cdot)$ and finite support E. Denote with E^* the set of modes of p, i.e. $p(\xi) \geq p(x)$ for all $\xi \in E^*$ and $x \in E$, and with $m := p(\xi)$ with $\xi \in E^*$. Then

$$\begin{aligned} p_{(\beta)}(x) &= \frac{(p(x))^\beta}{\sum_{y\in E^*}(p(y))^\beta + \sum_{y\in E\setminus E^*}(p(y))^\beta} \\ &= \frac{(p(x)/m)^\beta}{\sum_{y\in E^*} 1 + \sum_{y\in E\setminus E^*}(p(y)/m)^\beta} \overset{\beta\to\infty}{\longrightarrow} \begin{cases} 1/|E^*| & \text{if } x \in E^* \\ 0 & \text{if } x \notin E^* \end{cases} \end{aligned}$$

In general in the continuous case the distribution is not uniform on the modes and depends on the curvature around each mode (see Hwang 1980 for details). An outline of the argument is (Figure 3.30):

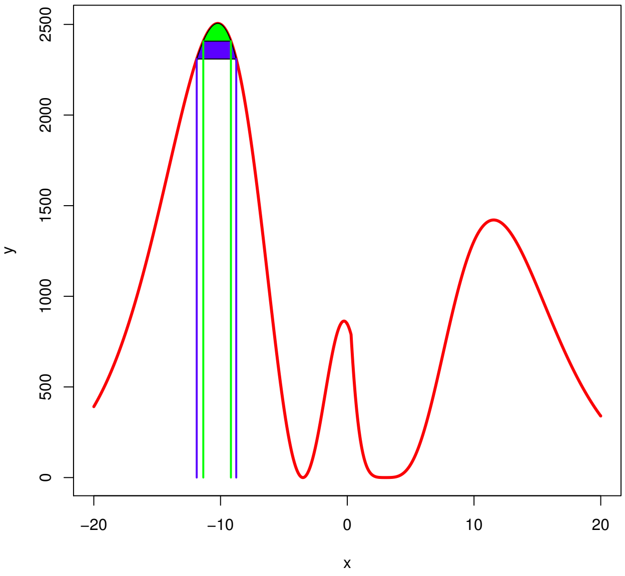

Figure 3.30: Simulated annealing for a continuous density.

- Take $x^\star \in \arg\max_x f(x)$ and let:
- $E_1 = \{x : f(x) \geq f(x^\star) - \epsilon\}$.
- $E_2 = \{x : f(x) \geq f(x^\star) - 2\epsilon\} \setminus E_1$.
- $D = \mathbb{R}^d \setminus (E_1 \cup E_2)$.
- Consider: $\lim_{\beta\to\infty} P_{(\beta)}(E_1 \cup E_2)/P_{(\beta)}(D)$.
- Use: $P_{(\beta)}(D)/P_{(\beta)}(E_1) \to 0$.

Under slightly stronger regularity conditions one can employ an elementary argument along the same lines as the one described in the discrete case above.

We can use a random-walk Metropolis algorithm to sample from $f_{(\beta)}(\cdot)$. The probability of accepting a move from $\boldsymbol{X}^{(t-1)}$ to $\boldsymbol{X}$ would be

$$\min\left\{1, \frac{f_{(\beta)}(\boldsymbol{X})}{f_{(\beta)}(\boldsymbol{X}^{(t-1)})}\right\} = \min\left\{1, \left(\frac{f(\boldsymbol{X})}{f(\boldsymbol{X}^{(t-1)})}\right)^\beta\right\}.$$

Note that this probability does not depend on the (generally unknown) normalisation constant of $f_{(\beta)}(\cdot)$. It is however difficult to directly sample from $f_{(\beta)}$ for large values of β: for $\beta \to \infty$ the probability of accepting a newly proposed X becomes 1 if $f(X) > f(X^{(t-1)})$ and 0 otherwise. Thus $X^{(t)}$ converges to a *local* extrema of the density

f, however not necessarily a mode of f (i.e. a *global* extremum of the density). Whether $X^{(t)}$ gets caught in a local extremum or not, depends on whether we can reach the mode from the *local* extrema of the density within one step. The following example illustrates this problem.

Example 3.19. Consider the following simple optimisation problem of finding the mode of the distribution defined on $\{1, 2, \ldots, 5\}$ by

$$p(x) = \begin{cases} 0.4 & \text{for } x = 2 \\ 0.3 & \text{for } x = 4 \\ 0.1 & \text{for } x = 1, 3, 5. \end{cases}$$

Figure 3.31 illustrates this distribution. Clearly, the (global) mode of $p(x)$ is at $x = 2$. Assume we want to sample from $p_{(\beta)}(x) \propto p(x)^{\beta}$ using a random walk Metropolis algorithm with proposed value $X = X^{(t-1)} + \varepsilon$ with $\mathbb{P}(\varepsilon = \pm 1) = 0.5$ for $X^{(t-1)} \in \{2, 3, 4\}$, $\mathbb{P}(\varepsilon = +1) = 1$ for $X^{(t-1)} = 1$, and $\mathbb{P}(\varepsilon = -1) = 1$ for $X^{(t-1)} = 5$. In other words, we can either move one to the left, stay in the current value (when the proposed value is rejected), or move one to the right. Note that for $\beta \to \infty$ the probability for accepting a move from 4 to 3 converges to 0, as $p(4) > p(3)$. As the Markov of chain can only move from 4 to 2 only via 3, it cannot escape the local extremum at 4 for $\beta \to +\infty$.*

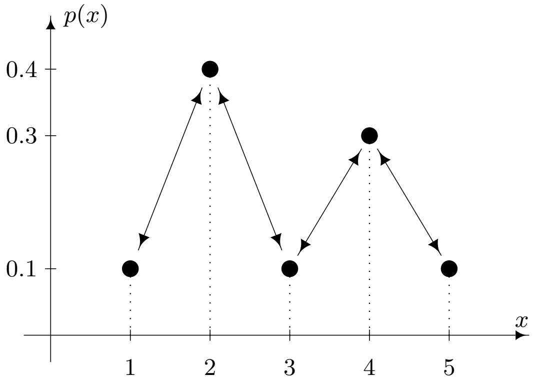

Figure 3.31: Illustration of a random walk Metropolis algorithm for the optimisation problem.

For large β the distribution $f_{(\beta)}(\cdot)$ is concentrated around the modes, however at the price of being difficult to sample from: the resulting Markov chain has very poor mixing properties: for large β the algorithm can hardly move away from a local extremum surrounded by areas of low probability (the density of such a distribution would have many local extrema separated by areas where the density is effectively 0).

The key idea of simulated annealing[4] (Kirkpatrick, Gelatt, and Vecchi 1983) is to sample from a target distribution that changes over time: $f_{(\beta_t)}(\cdot)$ with $\beta_t \to \infty$. Before we consider different strategies for choosing the sequence (β_t), we generalise the framework developed so far to finding the global extrema of arbitrary functions.

3.5.1 Minimising an arbitrary function

Consider that we want to find the global minimum of a function $h : E \to \mathbb{R}$. Finding the global minimum of $H(x)$ is equivalent to finding the mode of a distribution

$$f(x) \propto \exp(-H(x)) \text{ for } x \in E,$$

if such a distribution exists. In this framework, finding the mode of a density f corresponds to finding the minimum of $-\log(f(x))$. As in the previous section we can raise f to large powers to obtain a distribution

$$f_{(\beta_t)}(x) = (f(x))^{\beta_t} \propto \exp(-\beta_t \cdot H(x)) \text{ for } x \in E.$$

We hope to find the (global) minimum of $H(x)$, which is the (global) mode of the distribution defined by $f_{\beta_t}(x)$, by sampling from a Metropolis–Hastings algorithm. As suggested above we let $\beta_t \to \infty$. This yields the following algorithm:

[4] The term **annealing** comes from metallurgy and refers to the technique of melting a metal before allowing that metal to cool down slowly in order to reach a lower energy state and consequently produce a tougher metal. Following this analogy, $1/\beta$ is typically referred to as temperature, β as inverse temperature.

Algorithm 3.6 (Simulated Annealing). *Starting with* $\boldsymbol{X}^{(0)} := (X_1^{(0)}, \dots, X_p^{(0)})$ *and* $\beta^{(0)} > 0$ *iterate for* $t = 1, 2, \dots$

1. *Increase* $\beta^{(t-1)}$ *to* $\beta^{(t)}$ *(see below for different annealing schedules).*
2. *Draw* $\boldsymbol{X} \sim q(\cdot \mid \boldsymbol{X}^{(t-1)})$.
3. *Compute*
$$\alpha(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)}) = \min\left\{1, \exp\left(-\beta_t(H(\boldsymbol{X}) - H(\boldsymbol{X}^{(t-1)}))\right) \cdot \frac{q(\boldsymbol{X}^{(t-1)} \mid \boldsymbol{X})}{q(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)})}\right\}.$$
4. *With probability* $\alpha(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)})$ *set* $\boldsymbol{X}^{(t)} = \boldsymbol{X}$*, otherwise set* $\boldsymbol{X}^{(t)} = \boldsymbol{X}^{(t-1)}$.

If a random walk Metropolis update is used (i.e. $\boldsymbol{X} = \boldsymbol{X}^{(t-1)} + \boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim g(\cdot)$ for a symmetric g), then the probability of acceptance becomes

$$\alpha(\boldsymbol{X} \mid \boldsymbol{X}^{(t-1)}) = \min\left\{1, \exp\left(-\beta_t(H(\boldsymbol{X}) - H(\boldsymbol{X}^{(t-1)}))\right)\right\}.$$

Using the same arguments as in the previous section, it is easy to see that the simulated annealing algorithm converges to a *local* minimum of $H(\cdot)$. Whether it will be able to find the *global* minimum depends on how slowly we let the inverse temperature β go to infinity.

Logarithmic tempering When choosing $\beta_t = \frac{\log(1+t)}{\beta_0}$, the inverse temperature increases slow enough that global convergence results can be established for certain special cases. Hajek (1988) established global convergence when $H(\cdot)$ is optimised over a *finite* set using a proposal which is uniform over E and logarithmic tempering with a suitably large β_0. Andrieu, Breyer, and Doucet (2001) use Foster–Lyapunov type arguments to establish convergence on more general spaces (under appropriate conditions).

Assume we choose $\beta_0 = \Delta H$ with $\Delta H := \max_{x,x' \in E} |H(x) - H(x')|$. Then the probability of reaching state x in the t-th step is

$$\mathbb{P}\left(X^{(t)} = x\right) = \sum_{\xi} \underbrace{\mathbb{P}\left(X^{(t)} = x \mid X^{(t-1)} = \xi\right)}_{\geq \exp(-\beta_t \Delta H)/|E|} \mathbb{P}\left(X^{(t-1)} = \xi\right) \geq \exp(-\beta_t \Delta H)/|E|$$

Using the logarithmic tempering schedule we obtain $\mathbb{P}\left(X^{(t)} = x\right) \geq 1/\left((1+t)|E|\right)$ and thus the expected number of visits to state x is

$$\sum_{t=0}^{\infty} \mathbb{P}\left(X^{(t)} = x\right) \geq \sum_{t=0}^{\infty} [(1+t)|E|]^{-1} = \infty.$$

Thus every state is recurrent. As β increases we however spend an ever increasing amount of time in the global minima of x.

On the one hand visiting very state x infinitely often implies that we can escape from local minima. On the other hand, this implies as well that we visit every state x (regardless of how large $H(x)$ is) infinitely often. In other words, the reason why simulated annealing with logarithmic tempering works, is that it still behaves very much like an exhaustive search. However the only reason why we consider simulated annealing is that exhaustive search would be too slow! For this reason, logarithmic tempering has little practical relevance.

Geometric tempering A popular choice is $\beta_t = \alpha^t \cdot \beta_0$ for some $\alpha > 1$.

Example 3.20. *Assume we want to find the maximum of the function*

$$H(x) = \left((x-1)^2 - 1\right)^2 + 3 \cdot s(11.56 \cdot x^2), \text{ with}$$
$$s(x) = \begin{cases} |x| \mod 2 & \text{for } 2k \leq |x| \leq 2k+1 \\ 2 - |x| \mod 2 & \text{for } 2k+1 \leq |x| \leq 2(k+1) \end{cases}$$

for $k \in \mathbb{N}_0$. *Figure 3.32 shows* $H(x)$ *for* $x \in [-1, 3]$. *The global minimum of* $H(x)$ *is at* $x = 0$. *We simulated annealing with a geometric tempering with* $\beta_0 = 1$ *and* $\beta_t = 1.001\beta_{t-1}$ *and a random walk Metropolis algorithm with* $\varepsilon \sim \text{Cauchy}\left(0, \sqrt{0.1}\right)$. *Figure 3.33 shows the first 1,000 iterations of the Markov chain yielded by the simulated annealing algorithm. Note that when using a Gaussian distribution with small enough a variance the simulated annealing algorithm is very likely to remain in the local minimum at* $x \approx 1.8$.

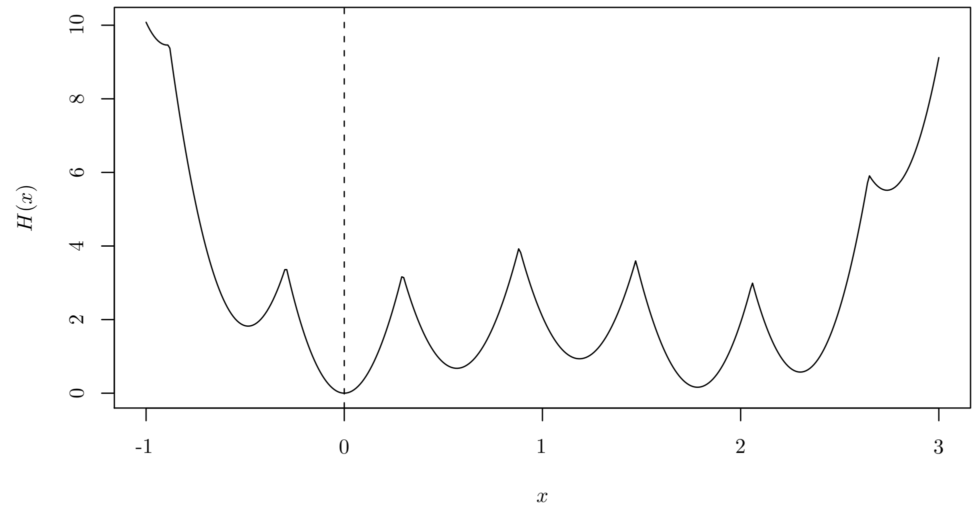

Figure 3.32: Objective function $H(x)$.

Note that there is no guarantee that the simulated annealing algorithm converges to the global minimum of $H(x)$ in finite time. In practice, it would be unrealistic to expect simulated annealing to converge to a *global* minimum, however in most cases it will find a "good" *local* minimum.

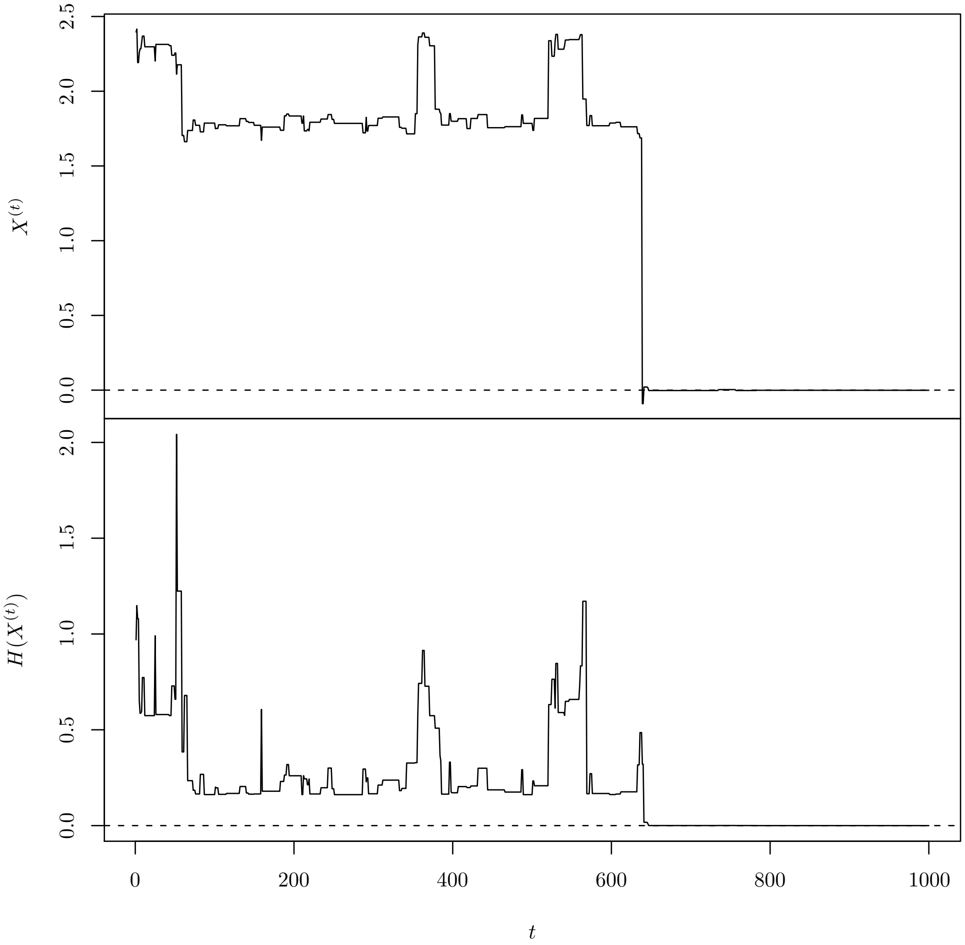

Figure 3.33: First 1,000 iterations of the Markov chain yielded by simulated annealing for finding the maximum of $H(x)$.

Chapter 4

Augmentation: Extending the Space

A very general technique in the field of simulation based inference is to augment the space on which simulation is done with auxiliary variables whose presence makes the problem easier. It may seem counterintuitive that making the space on which one must sample *larger* can make the sampling *easier*, but as we shall see in this chapter their are many techniques in the literature which can be seen as particular cases of this general strategy.

4.1 Composition Sampling

Consider the problem of drawing samples from a mixture distribution, i.e. one with a density of the form

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \sum_{i=1}^{k} w_i f_i(\boldsymbol{x})$$

where $\boldsymbol{w} = (w_1, \ldots, w_k)$ is a vector of non-negative real numbers which sum to one. These correspond to component *weights* and $\{f_i\}_{i=1}^k$ corresponds to a family of k different probability densities.

In principle one can readily develop techniques for sampling from such densities using the ideas described in Section 2.1. However, it's convenient to have a simple generic method which can be used whenever we have techniques for sampling from the f_i individually.

Consider introducing an auxiliary variable Z which has a discrete distribution over $\{1, \ldots, k\}$ with associated vector of probability masses $\boldsymbol{w}$. The joint distribution

$$f_{\boldsymbol{X},Z}(\boldsymbol{x}, z) = \sum_{i=1}^{k} w_i \delta_{i,z} f_z(\boldsymbol{x}),$$

admits the marginal over $\boldsymbol{x}$:

$$\sum_{z=1}^{k} f_{\boldsymbol{X},Z}(\boldsymbol{x}, z) = \sum_{z=1}^{k} \sum_{i=1}^{k} w_i \delta_{i,z} f_z(\boldsymbol{x}) = \sum_{i=1}^{k} w_i \sum_{z=1}^{k} \delta_{i,z} f_z(\boldsymbol{x}) = \sum_{i=1}^{k} w_i f_i(\boldsymbol{x}) = f_{\boldsymbol{X}}(\boldsymbol{x}),$$

as required.

The basis of composition sampling is that $f_{\boldsymbol{X},Z}$ also admits a straightforward marginal distribution over z, by construction, $f_Z(z) = \sum_{i=1}^k w_i \delta_{i,z}$ and the conditional distribution of $\boldsymbol{X}$ given $Z = z$ is simply $f_{\boldsymbol{X}|Z}(\boldsymbol{x}|z) = f_z(\boldsymbol{x})$.

Combining these ideas, we conclude that we can sample from $f_{\boldsymbol{X},Z}$ by sampling $Z \sim \mathsf{Cat}(\boldsymbol{w})$, sampling $\boldsymbol{X}$ from its conditional distribution given the realized value of Z: $\boldsymbol{X}|\{Z = z\} \sim f_z$ and then discarding the auxiliary variable z.

4.2 Rejection Revisited

We can also look at rejection sampling through the lens of spatial extension. Consider the following scenario. We are interested in obtaining samples from $f_{\boldsymbol{X}}$, know that $\sup_{\boldsymbol{x}} f_{\boldsymbol{X}}(\boldsymbol{x})/g_{\boldsymbol{X}}(\boldsymbol{x}) \leq M < \infty$ for some density $g_{\boldsymbol{X}}$ from which we can sample and some real constant M. We extend the space by introducing an additional U which takes its values in $\mathbb{R}_+$ and define the joint distributions:

$$g_{\boldsymbol{X},U}(\boldsymbol{x},u) \propto \mathbb{I}_{\mathsf{G}_M}((\boldsymbol{x},u)), \quad f_{\boldsymbol{X},U}(\boldsymbol{x},u) \propto \mathbb{I}_{\mathsf{F}}((\boldsymbol{x},u)),$$

where $\mathsf{G}_M := \{(\boldsymbol{x},u) \in \mathcal{X} \otimes \mathbb{R}_+ : u \leq Mg(\boldsymbol{x})\}$ and $\mathsf{F} := \{(\boldsymbol{x},u) \in \mathcal{X} \otimes \mathbb{R} : u \leq f(\boldsymbol{x})\}$ are simply the sets of points beneath Mg and f, respectively.

If $g_{\boldsymbol{X}}$ is tractable then we can straightforwardly sample from $g_{\boldsymbol{X},U}$ by sampling $\boldsymbol{X} \sim g_{\boldsymbol{x}}$ and $U|\boldsymbol{X} = \boldsymbol{x} \sim \mathsf{U}[0, Mg(\boldsymbol{x})]$. We could then imagine conducting importance sampling to approximate expectations under $f_{\boldsymbol{X},U}$, which would lead us to an estimator for $I = \int f_{\boldsymbol{X},U}(\boldsymbol{x},u)\varphi(\boldsymbol{x},u)d\boldsymbol{x}du$ of the form

$$\hat{I}^n_\varphi = \frac{\sum_{i=1}^n \frac{\mathbb{I}_{\mathsf{F}}((\boldsymbol{X}_i,U_i))}{\mathbb{I}_{\mathsf{G}_M}((\boldsymbol{X}_i,U_i))}\varphi(\boldsymbol{X}_i)}{\sum_{i=1}^n \frac{\mathbb{I}_{\mathsf{F}}((\boldsymbol{X}_i,U_i))}{\mathbb{I}_{\mathsf{G}_M}((\boldsymbol{X}_i,U_i))}}$$

noting that $\mathsf{F} \subseteq \mathsf{G}_M$ and that the probability (under the sampling mechanism by which we have just described for simulating these random variables which amounts to sampling from the uniform distribution over G_M) that $\boldsymbol{X}_i \notin \mathsf{G}_M$ is 0 allowing us to adopt the convention that $0/0 = 0$ here, we obtain:

$$\hat{I}^n_\varphi = \frac{\sum_{i=1}^n \mathbb{I}_{\mathsf{F}}((\boldsymbol{X}_i,U_i))\varphi(\boldsymbol{X}_i,U)}{\sum_{i=1}^n \mathbb{I}_{\mathsf{F}}((\boldsymbol{X}_i,U_i))} = \frac{\sum_{\{i:(\boldsymbol{X}_i,U_i)\in\mathsf{F}\}}\varphi(\boldsymbol{X}_i,U)}{\sum_{\{i:(\boldsymbol{X}_i,U_i)\in\mathsf{F}\}}1}.$$

Note that this is simply the sample average of the function φ over those points which fell within F. If we restrict our attention to $\varphi(\boldsymbol{x},u) = \varphi(\boldsymbol{x})$ (i.e. we consider functions which depend only upon $\boldsymbol{x}$) it is clear that we've recast the simple Monte Carlo estimate of the expectation of a function under $f_{\boldsymbol{X}}$ using a sample obtained using n proposals from g within a rejection sampler as an importance sampling estimate on an extended space.

The relationship between rejection and importance sampling is well known and has been studied by many authors (Chen 2005; Perron 1999).

4.3 Data Augmentation

Perhaps the most widely known spatial extension technique is that known as data augmentation, introduced by Tanner and Wong (1987).

Consider a *latent variable model*: a statistical model in which one has unknown parameters about which one wishes to performance inference, $\boldsymbol{\theta}$, observations which are known, $\boldsymbol{y}$, and a collection of hidden (latent) variables, $\boldsymbol{z}$. Typically, the joint distribution of all these quantities, say, $f_{\boldsymbol{Y},\boldsymbol{Z},\boldsymbol{\theta}}$ is known but integrating out the latent variables is not feasible. Without access to $f_{\boldsymbol{Y},\boldsymbol{\theta}}$ it's not possible to implement, directly, an MCMC algorithm with the associated marginal posterior distribution $f_{\boldsymbol{\theta}|\boldsymbol{Y}}$ as its target.

The basis of data augmentation is to *augment* the vector of parameters $\boldsymbol{\theta}$ with these latent variables, $\boldsymbol{z}$ and to run an MCMC algorithm (or other Monte Carlo algorithm of your choice) which instead targets the joint posterior distribution $f_{\boldsymbol{\theta},\boldsymbol{Z}|\boldsymbol{Y}}$ noting that this distribution admits as its marginal in $\boldsymbol{\theta}$ exactly the marginal posterior distribution which was the original object of inference. A mixture model is the canonical example of a model which can be susceptible to this approach.

4.4 Multiple Augmentation for Optimisation

A closely related idea used in optimisation is based around "multiple augmentation", introducing several replicates of unobserved quantities in order to allow the maximisation of a marginal quantity which it may not be possible to evaluate. We focus here on the State Augmentation for Maximisation of Expectations algorithm of Doucet, Godsill, and Robert (2002); similar methods are also described by others including Gaetan and Yao (2003), Jacquier, Johannes, and Polson (2007). These schemes all employ MCMC; the alternative of employing a population-based sampling method known as Sequential Monte Carlo was explored by Johansen, Doucet, and Davy (2008).

Two common optimisation problems arise in the evaluation of statistical estimators: *Maximum Likelihood Estimation:* Given $L(\theta;\boldsymbol{x}) = f_{\boldsymbol{x}}(\boldsymbol{x};\theta)$, compute $\hat{\theta}_{\mathrm{ML}} = \arg\max_{\theta\in\Theta} L(\theta;\boldsymbol{x})$ and *Maximum a Posteriori Estimation:* Given $L(\theta;\boldsymbol{x}) = f_{\boldsymbol{x}}(\boldsymbol{x};\theta)$ and prior $f^{\mathrm{prior}}(\theta)$, compute $\hat{\theta}_{\mathrm{MAP}} = \arg\max_{\theta\in\Theta} f^{\mathrm{prior}}(\theta)L(\theta;\boldsymbol{x})$.

Both can fit our simple optimisation framework, and we can see a further illustration of the workings of the annealing method by considering the sequence of distributions obtained for simple problems.

Example 4.1. (Gaussian MAP Estimation).

- *If* $L(\mu;\boldsymbol{x}) = \prod_{i=1}^{n} \phi_{\mu,\sigma^2}(x_i)$ *with* σ^2 *known,*
- *and* $\pi(\mu) = \phi_{\mu_0,\sigma_0^2}(\mu)$*, then*
- *the posterior is*

$$f^{post}(\mu) = \mathsf{N}\left(\mu, \frac{\sigma^2\mu + n\sigma_0^2\bar{x}}{\sigma^2 + n\sigma_0^2}, \frac{\sigma^2\sigma_0^2}{\sigma^2 + n\sigma_0^2}\right),$$

- *and we could aim to sample from*

$$f_{(\beta)}^{MAP}(\mu) \propto (f^{post}(\mu))^{\beta} \propto \mathsf{N}\left(\mu, \frac{\sigma^2\mu + n\sigma_0^2\bar{x}}{\sigma^2 + n\sigma_0^2}, \frac{\sigma^2\sigma_0^2}{\beta(\sigma^2 + n\sigma_0^2)}\right).$$

Example 4.2. (Example: Normal ML Estimation).

- *If* $L(\mu;\boldsymbol{x}) = \prod_{i=1}^{n} \phi_{\mu,\sigma^2}(x_i)$ *with* σ^2 *known,*
- *we could view the likelihood as being proportional to a distribution over* μ*:*

$$f(\mu) = \mathsf{N}\left(\mu; \bar{x}, \sigma^2/n\right),$$

- *and we could aim to sample from*

$$f_{(\beta)}^{MLE}(\mu) \propto (f(\mu))^{\beta} \propto \mathsf{N}\left(\mu; \bar{x}, \sigma^2/\beta n\right).$$

In both of these cases, the sequence of distributions concentrates on the maximiser of the original objective function, and so any algorithm able to sample from these distributions (for large enough β) will provide good approximations of the optimiser of the objective function. These methods involve target distributions that resemble the posterior distribution (either a real posterior, or one obtained using an instrumental prior for the purposes of approximating the MLE) which would have been obtained if there were many copies of the data, so the approach is sometimes referred to as "data cloning".

Two closely related problems often arise when dealing with complicated statistical models. *Marginal Maximum Likelihood Estimation:* Given $L(\theta;\boldsymbol{x}) = \int f_{\boldsymbol{x},\boldsymbol{z}}(\boldsymbol{x},\boldsymbol{z};\theta)d\boldsymbol{z}$, compute $\hat{\theta}_{\text{ML}} = \arg\max_{\theta\in\Theta} L(\theta;\boldsymbol{x})$; and *Marginal Maximum a Posteriori Estimation:* Given $L(\theta;\boldsymbol{x}) = \int f_{\boldsymbol{x},\boldsymbol{z}}(\boldsymbol{x},\boldsymbol{z};\theta)d\boldsymbol{z}$ and prior $f^{\text{prior}}(\theta)$, compute $\hat{\theta}_{\text{MMAP}} = \arg\max_{\theta\in\Theta} f^{\text{prior}}(\theta)L(\theta;\boldsymbol{x})$. Such problems often arise when one can write down a complete generative model for the process by which the data arose in terms of the parameters, but one only observes a subset of the random quantities generated within that model. For example, consider a mixture model in which we don't observe the association of observations with mixture components or a genetic model in which we observe the DNA sequences of only the current generation of individuals: we don't observe the sequences of their ancestors or their *genealogical trees* (Stephens 2007). If it is possible to integrate out the unobserved random quantities then we can proceed as usual, but unfortunately we can't typically evaluate the marginal likelihoods.

Recall the *demarginalisation* technique for sampling from $f_{\boldsymbol{x}}(\boldsymbol{x})$ by defining a convenient joint distribution $f_{\boldsymbol{x},\boldsymbol{z}}(\boldsymbol{x},\boldsymbol{z})$ which admits the distribution of interest as a marginal. In order to do this, we saw that we could introduce a set of auxiliary random variables $Z_1,\ldots,Z_r$ such that $f_{\boldsymbol{x}}$ is the marginal density of $(X_1,\ldots,X_p)$ under the joint distribution of $(X_1,\ldots,X_p,Z_1,\ldots,Z_r)$, i.e.

$$f(x_1,\ldots,x_p) = \int f(x_1,\ldots,x_n,z_1,\ldots,z_r)\, d(z_1,\ldots,z_r).$$

The idea of introducing some auxiliary random variables in such a way that $f_{(\beta)}(\boldsymbol{x})$ is the marginal distribution seems a natural extension of this idea.

In order to do this, we consider

$$L(\boldsymbol{x},\boldsymbol{z}|\theta) = f_{X,Z}(\boldsymbol{x},\boldsymbol{z}|\theta) = f_Z(\boldsymbol{z}|\theta)f_X(\boldsymbol{x}|\boldsymbol{z},\theta),$$

and introduce a whole collection of *vectors* of auxiliary variables:

$$f_\beta^{MMAP}(\theta, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_\beta | \boldsymbol{x}) \propto \prod_{i=1}^{\beta} \left[\pi(\theta) f_Z(\boldsymbol{z}_i) f_X(\boldsymbol{x}|\boldsymbol{z}_i, \theta)\right].$$

We can easily establish that, by exploiting the conditional independence structure of our augmented likelihood:

$$\begin{aligned} f_\beta^{MMAP}(\theta|\boldsymbol{x}) &\propto \int f_\beta^{MMAP}(\theta, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_\beta|\boldsymbol{x}) d\boldsymbol{z}_1, \ldots d\boldsymbol{z}_\beta \\ &\propto \pi(\theta)^\beta f_X(\boldsymbol{x}|\theta)^\beta = f^{\text{post}}(\theta|\boldsymbol{x})^\beta. \end{aligned}$$

This idea is the basis of the *State Augmentation for Maximisation of Expectations* (SAME) algorithm (Doucet, Godsill, and Robert 2002).

In the case of maximising the likelihood rather than the posterior we need to be slightly more careful. The likelihood is a probability density over the data, but need not even be integrable if viewed as a function of the parameters. We can address this problem by introducing an *instrumental* prior distribution (one used exclusively for computational reasons which is not intended to have any influence on the resulting inference.

Considering

$$L(\theta; \boldsymbol{x}, \boldsymbol{z}) = f_{X,Z}(\boldsymbol{x}, \boldsymbol{z}|\theta) = f_Z(\boldsymbol{z}|\theta) f_X(\boldsymbol{x}|\boldsymbol{z}, \theta),$$

we can again consider multiple augmentation—this time for MMLE estimation—by setting

$$f_\beta^{MMLE}(\theta, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_\beta|\boldsymbol{x}) \propto \pi(\theta) \prod_{i=1}^{\beta} \left[f_Z(\boldsymbol{z}_i) f_X(\boldsymbol{x}|\boldsymbol{z}_i, \theta)\right],$$

which ensures that

$$\begin{aligned} f_\beta^{MMLE}(\theta|\boldsymbol{x}) &\propto \int f_\beta^{MMLE}(\theta, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_\beta|\boldsymbol{x}) d\boldsymbol{z}_1, \ldots d\boldsymbol{z}_\beta \\ &\propto \left[\pi(\theta)^{(1/\beta)} f_X(\boldsymbol{x}|\theta)\right]^\beta \approx L(\theta; \boldsymbol{x})^\beta \end{aligned}$$

for large enough β under support and regularity conditions on $\pi(\cdot)$, the *instrumental* prior.

Both of these augmentation strategies can give rise to a sequence of target distributions if we replace β with β_t, a non-decreasing sequence of numbers of replicates of the augmenting variables. (In the SAME case it can be sensible to keep β_t fixed at a particular value for several iterations to give the chain time to reach equilibrium before further increasing it.) And given such a sequence of target distributions, we can apply MCMC kernels for which each is invariant in essentially the same manner as we did when considering simulated annealing. In the particular case in which we can sample from all of the relevant full conditional distributions, this gives rise to Algorithm 4.1; more general cases can be dealt with via obvious extensions.

Algorithm 4.1 (The SAME Gibbs Sampler). *Starting with* $\boldsymbol{\theta}^{(0)}$ *iterate for* $t = 1, 2, \ldots$

1. *Increase* $\beta^{(t-1)}$ *to* $\beta^{(t)}$ *(if necessary).*
2. *For* $k = 1, \ldots, \beta_t$*, sample:*

$$\boldsymbol{z}_k^{(t)} \sim f_Z(\boldsymbol{z}_k^{(t)}|x, \boldsymbol{\theta}^{(t-1)}).$$

3. *Sample:*

$$\boldsymbol{\theta}^{(t)} \sim f_{(\beta_t)}^{\cdots}(\boldsymbol{\theta}|\boldsymbol{x}, \boldsymbol{z}_1^{(t)}, \ldots, \boldsymbol{z}_{\beta_t}^{(t)}).$$

The following toy example shows the SAME Gibbs sampler in action.

Example 4.3. *Consider finding the parameters which maximise the likelihood in a setting in which the likelihood is a student t-distribution of unknown location parameter* θ *with* 0.05 *degrees of freedom. Four observations are available,* $\boldsymbol{x} = (-20, 1, 2, 3)$*.*

In this case, the marginal likelihood is known (and we can use this knowledge to verify that the algorithm works as expected):

$$\log p(\boldsymbol{x}|\theta) = -0.525 \sum_{i=1}^{4} \log\left(0.05 + (x_i - \theta)^2\right).$$

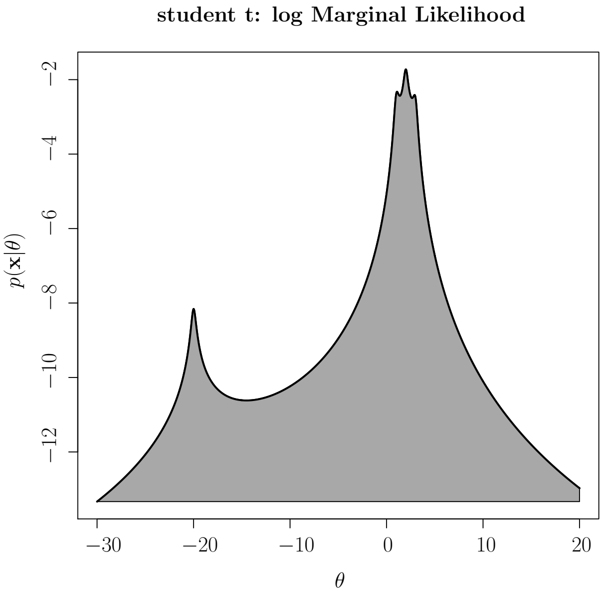

Figure 4.1: The log marginal likelihood for the location parameter of the student t-distribution.

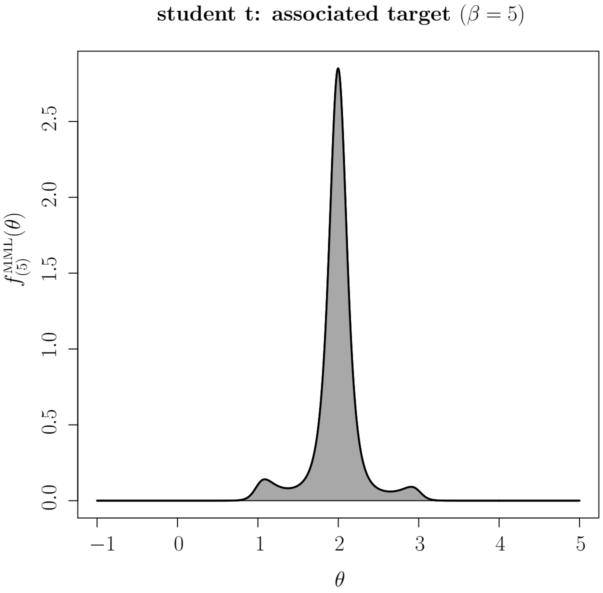

Figure 4.2: The target distribution obtained by the annealing approach at $\beta = 5$ (right) for the location parameter of the student t-distribution.

This marginal likelihood is illustrated in Figure 4.1–4.2.

However, it is also possible to write down an augmented complete likelihood admitting this as a marginal distribution, by exploiting the fact that the student t-distribution may be written as a ***scale mixture*** *of normal densities:*

$$\log p(\boldsymbol{x}, \boldsymbol{z}|\theta) = -\sum_{i=1}^{4} \left[0.475 \log z_i + 0.025 z_i + 0.5 z_i (x_i - \theta)^2\right],$$

$$p_{(\beta_t)}(\boldsymbol{z}_{1:\beta_t}|\theta, \boldsymbol{x}) = \prod_{i=1}^{\beta_t} \prod_{j=1}^{4} \mathsf{Gamma}\left(z_{i,j}; 0.525, 0.025 + \frac{(x_j - \theta)^2}{2}\right),$$

$$p_{(\beta_t)}(\theta|\boldsymbol{z}_{1:\beta_t}) \propto \mathsf{N}\left(\theta; \mu_t^{(\theta)}, \Sigma_t^{(\theta)}\right),$$

where the parameters are

$$\Sigma_t^{(\theta)} = \left[\sum_{i=1}^{\beta_t} \sum_{j=1}^{4} z_{i,j}\right]^{-1}, \quad \mu_t^{(\theta)} = \Sigma_t^{(\theta)} \sum_{i=1}^{\beta_t} y^T z_i.$$

We can straightforwardly implement the SAME Gibbs sampler for this problem..

It is perhaps more interesting to return to the familiar mixture model for which we have already considered several forms of inference. Lets apply the data augmentation approach to the problem of maximising the posterior density. (Note that one cannot use maximum likelihood estimation, at least directly, in this setting as the likelihood is not bounded above: if a cluster mean coincides exactly with an observation then making the variance of that component arbitrarily small leads to an arbitrarily high likelihood!)

Example 4.4. (MAP Estimation for a Gaussian Mixture Model). *Consider again the Gaussian mixture model in which we assume that the density of y_i is a mixture of Gaussians*

$$f(y_i|\pi_1, \ldots, \pi_k, \mu_1, \ldots, \mu_k, \tau_1, \ldots, \tau_k) = \sum_{\kappa=1}^{k} \pi_\kappa \phi_{(\mu_\kappa, 1/\tau_\kappa)}(y_i).$$

Suitable prior distributions are a Dirichlet distribution for $(\pi_1, \ldots, \pi_k)$, a Gaussian for μ_κ, and a Gamma distribution for τ_κ. In order to ensure identifiability we assume the μ_κ are ordered, i.e. $\mu_1 < \cdots < \mu_k$ and make the corresponding change to the posterior density (in order to compensate for setting the density to zero for all configurations which fail to satisfy the ordering constraint, the density of all configurations compatible with the constraint must be increased by a factor of $k!$). Here we assume that k is known, and have:

- *n iid observations, $x_1, \ldots, x_n$.*
- *Likelihood $f_{X,Z}(x_i, z_i|\omega, \mu, \sigma) = \omega_{z_i} \mathsf{N}\left(x_i; \mu_{z_i}, \sigma_{z_i}^2\right)$.*
- *Marginal likelihood $f_X(x_i|\omega, \mu, \sigma) = \sum\limits_{j=1}^{K} \omega_j \mathsf{N}\left(x_i; \mu_j, \sigma_j^2\right)$.*
- *Diffuse conjugate priors:*

$$\omega \sim \mathsf{Dirichlet}(\chi, \ldots, \chi),$$
$$\sigma_i^2 \sim \mathsf{IG}\left(\frac{\lambda_i + 3}{2}, \frac{b_i}{2}\right),$$
$$\mu_i|\sigma_i^2 \sim \mathsf{N}\left(a_i, \sigma_i^2/\lambda_i\right).$$

All full conditional distributions of interest are available, which allows us to use our Gibbs sampling strategy. This gives rise to an iterative algorithm in which step t comprises the following steps:

- *Sample:*

$$\omega \leftarrow \mathsf{Dirichlet}(\beta_t(\chi - 1) + 1 + n_1(\beta_t), \ldots, \beta_t(\chi - 1) + 1 + n_K(\beta_t)),$$
$$\sigma_i^2 \leftarrow \mathsf{IG}(A_i, B_i),$$
$$\mu_i|\sigma_i^2 \leftarrow \mathsf{N}\left(\frac{\beta_t \lambda_i a_i + \bar{\boldsymbol{x}}_i^{\beta_t}}{\beta_t \lambda_i + n_i^{\beta_t}}, \frac{\sigma_i^2}{\beta_t \lambda_i + n_i^{\beta_t}}\right),$$

where

$$n_i^{\beta_t} = \sum_{l=1}^{\beta_t}\sum_{p=1}^{n} \mathbb{I}_i(Z_{l,p}^{(t-1)}), \quad \bar{\boldsymbol{x}}_i^{\beta_t} = \sum_{l=1}^{\beta_t}\sum_{p=1}^{n} \mathbb{I}_i(Z_{l,p}^{(t-1)})x_j, \quad \overline{\boldsymbol{x}^2}_i^{\beta_t} = \sum_{l=1}^{\beta_t}\sum_{p=1}^{n} \mathbb{I}_i(Z_{l,p})x_j^2,$$

- *and*

$$A_i = \frac{\beta_t(\lambda_i + 1) + n_i^{\beta_t}}{2} + 1,$$
$$B_i = \frac{1}{2}\left(\beta_t(b_i + \lambda_i a_i^2) + \bar{\boldsymbol{x}^2}_i^{\beta_t} - \sum_{g=1}^{\beta_t} \frac{(\bar{\boldsymbol{x}}_i^g - \bar{\boldsymbol{x}}_i^{g-1} + \lambda_i a_i)^2}{\lambda_i + n_i^g - n_i^{g-1}}\right).$$

- *Sample, for $j = 1, \ldots, \beta_t$:*

$$\boldsymbol{z}_j^{(t)} \sim f^{posterior}(\boldsymbol{z}|\boldsymbol{x}, \pi^{(t)}, \sigma^{(t)}, \mu^{(t)}).$$

The marginal posterior can be calculated (which means that we don't ***need*** *such a complicated algorithm to deal with this problem, although it does perform well; the advantage of using such an example is that it allows us to assess the performance of the algorithm).*

First we compare the performance of 50 runs of the algorithm with 50 (differently initialised) runs of a deterministic algorithm (expectation maximisation; EM) which is widely used to deal with problems of this type. ***Cost*** *gives a rough indication of the computational cost of running each algorithm once.*

Algorithm	*T*	***Cost***	***Mean***	***Std. Dev.***	***Min***	***Max***
EM	*500*	*500*	*-158.06*	*3.23*	*-166.39*	*-153.85*
EM	*5000*	*5000*	*-157.73*	*3.83*	*-165.81*	*-153.83*
SAME(6)	*4250*	*8755*	*-155.32*	*0.87*	*-157.35*	*-154.03*
SAME(50)	*4250*	*112522*	*-155.05*	*0.82*	*-156.11*	*-153.98*

Two different sequences of the annealing parameter were considered:

SAME(6): *Set $\beta_t = 1$ for the first half of the iterations and then increasing linearly to a final maximum value of 6.*

SAME(50): *Set $\beta_t = 1$ for the first 250 iterations, and then increasing linearly to 50.*

The log posterior density of the generating parameters was -155.87. These parameters were:

$$\pi = [0.2, 0.3, 0.5], \quad \mu = [0, 2, 3], \quad \textit{and } \sigma = \left[1, \frac{1}{4}, \frac{1}{16}\right].$$

Although the EM algorithm occasionally produces good results, for this clean simulated data, some runs of the algorithm totally fail to find anything close to the global mode. The SAME algorithm is computationally more costly, but does behave more robustly. In real marginal optimisation problems, one typically cannot evaluate the objective function and so robust methods that can be relied upon to produce good solutions are required.

Next we turn to the much celebrated Galaxy data set of Roeder (1990). This data set consists of the velocities of 82 galaxies, and it has been suggested that it consists of a mixture of between 3 and 7 distinct components—for example, see Roeder and Wasserman (1997) and Escobar and West (1995). For our purposes we have estimated the parameters of a 3 component Gaussian mixture model from which we assume the data was drawn. The following table summarises the marginal posterior of the solutions found by 50 runs of each algorithm, comparing the same algorithm with the EM algorithm. ***Cost*** *gives a rough indication of the computational cost of running each algorithm once.*

Algorithm	*T*	***Cost***	***Mean***	***Std. Dev.***	***Min***	***Max***
EM	*500*	*500*	*-46.54*	*2.92*	*-54.12*	*-44.32*
EM	*5000*	*5000*	*-46.91*	*3.00*	*-56.68*	*-44.34*
SAME(6)	*4250*	*8755*	*-45.18*	*0.54*	*-46.61*	*-44.17*
SAME(50)	*4250*	*112522*	*-44.93*	*0.21*	*-45.52*	*-44.47*

Algorithm	*T*	*Cost*	*Mean*	*Std. Dev.*	*Min*	*Max*

Again, two different sequences of annealing schedule were considered:

SAME(6) *set $\beta_t = 1$ for the first half of the iterations and then increasing linearly to a final maximum value of 6,*

SAME(50) *set $\beta_t = 1$ for the first 250 iterations, and then increasing linearly to 50,*

and again, good robustness is demonstrated by the same algorithm. A slightly more sophisticated algorithm (Johansen, Doucet, and Davy 2008) finds solutions broadly consistent with the -43.96 ± 0.03 found here.

4.5 Approximate Bayesian Computation

The Approximate Bayesian Computation (ABC) approach to inference has become extremely popular for performing inference for models whose likelihood is not tractable (either in the sense that we can't evaluate it pointwise or that such evaluation is prohibitively expensive). Such models abound in some areas, such as ecology and phylogenetics, and these methods have consequently received a great deal of attention in recent years.

It was Pritchard et al. (1999) who introduced the method, although there are some connections to earlier work such as Diggle and Gratton (1984) and Tavaré et al. (1997). It is not always viewed as a spatial extension technique, but it can be quite helpful to think about it in these terms.

Before moving on to consider ABC itself, think about a simple case in which one has a target distribution $f_{\boldsymbol{X}|\boldsymbol{Y}}(\boldsymbol{x}|\boldsymbol{y})$ which will typically be a Bayesian posterior distribution (and $\boldsymbol{y}$ the observed data). This distribution is written, via Bayes rule as:

$$f_{\boldsymbol{X}|\boldsymbol{Y}}(\boldsymbol{x}|\boldsymbol{y}) = \frac{f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{y}|\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x})}{f_{\boldsymbol{Y}}(\boldsymbol{y})}.$$

If both $f_{\boldsymbol{X}}(\boldsymbol{x})$ and $f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{y}|\boldsymbol{x})$ can be evaluated pointwise then we can use standard simulation techniques to obtain samples which we can use to approximate our target distribution, and to approximate expectations with respect to it.

If we *cannot* evaluate $f_{\boldsymbol{Y}|\boldsymbol{X}}$ even pointwise, then we *can't* directly use the techniques which we've described previously. To address this, we can invoke a clever data augmentation trick which requires only that we can *sample* from $f_{\boldsymbol{Y}|\boldsymbol{X}}$. First let's consider the case in which $\boldsymbol{Y}$ is a *discrete* random variable. We can define the extended distribution, with $\boldsymbol{Z}$ taking it values in the space space as $\boldsymbol{Y}$:

$$f_{\boldsymbol{X},\boldsymbol{Z}|\boldsymbol{Y}}(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{y}) \propto f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{z}|\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \delta_{\boldsymbol{y},\boldsymbol{z}}$$

and note that it has as a marginal distribution, our target:

$$\sum_{\boldsymbol{z}} f_{\boldsymbol{X},\boldsymbol{Z}|\boldsymbol{Y}}(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{y}) \propto \sum_{\boldsymbol{z}} f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{z}|\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}) \delta_{\boldsymbol{y},\boldsymbol{z}} = f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{y}|\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}).$$

In the simplest case, we can sample $(\boldsymbol{X}, \boldsymbol{Z}) \sim f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{z}|\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x})$ using this as a rejection sampling proposal for our target distribution, keeping samples with probability proportional to

$$f_{\boldsymbol{X},\boldsymbol{Z}|\boldsymbol{Y}}(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{y}) / f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{z}|\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x}).$$

This probability can easily be seen to be proportional to $\delta_{\boldsymbol{y},\boldsymbol{z}}$. So this rejection sampling algorithm amounts to sampling $\boldsymbol{X}$ from its prior distribution; sampling an artificial set of data from the model $f_{\boldsymbol{Y}|\boldsymbol{X}}$ and keeping the sample as a sample from the posterior only if the artificial data set exactly matches the observed one.

Thus far, this clever algorithm has made no approximations. However, the probability of a sample being accepted is exactly the probability that a data set drawn by sampling a parameter value from the prior and a data set from the data-generating model with that parameter value *exactly* matches the observed data. In the case of very small discrete data sets this might be acceptable, but typically it will be vanishingly small. That's why approximation becomes necessary.

The approximate part of ABC arises first of all by relaxing the requirement that the simulated data *exactly* matches the observed data and keeping any sample for which the simulated data falls within some tolerance, ϵ, of the observed data. This leads to a *different* target distribution:

$$f^{\mathrm{ABC}}_{\boldsymbol{X},\boldsymbol{Z}|\boldsymbol{Y}} \propto f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{z}|\boldsymbol{x})f_{\boldsymbol{X}}(\boldsymbol{x})\mathbb{I}_{B(\boldsymbol{y},\epsilon)}(\boldsymbol{z}),$$

where $B(\boldsymbol{y},\epsilon) := \{\boldsymbol{x} : |\boldsymbol{x}-\boldsymbol{y}| \leq \epsilon\}$, for which the marginal is *no longer correct* but may be approximately so under regularity conditions:

$$\begin{aligned} f^{\mathrm{ABC}}_{\boldsymbol{x}|\boldsymbol{Y}} &\propto \int f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{z}|\boldsymbol{x})f_{\boldsymbol{X}}(\boldsymbol{x})\mathbb{I}_{B(\boldsymbol{y},\epsilon)}(\boldsymbol{z})d\boldsymbol{z} \\ &\propto f_{\boldsymbol{X}}(\boldsymbol{x})\int f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{z}|\boldsymbol{x})\mathbb{I}_{B(\boldsymbol{y},\epsilon)}(\boldsymbol{z})d\boldsymbol{z} \propto f_{\boldsymbol{X}}(\boldsymbol{x})\int_{\boldsymbol{z}\in B(\boldsymbol{y},\epsilon)} f_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{z}|\boldsymbol{x})d\boldsymbol{z}. \end{aligned}$$

This approximation amounts to a *smoothing* of the likelihood function.

Often a further approximation is introduced by considering not the data itself but some low dimensional summary. If that low dimensional summary does not constitute a sufficient statistic for the inferential task at hand, this induces an additional approximation error which doesn't vanish even if the tolerance parameter, ϵ, is reduced to zero. We won't consider this further here, but it is important to be aware of the impact of such approximation if you employ this type of technique in real inferential situations.

Using simple rejection sampling with such a target distribution leads to the ABC algorithm of Pritchard et al. (1999), while using this target distribution within a standard MCMC algorithm was proposed by Marjoram et al. (2003) with various approaches based around other Monte Carlo schemes, especially Sequential Monte Carlo also being proposed by various authors including Sisson, Fan, and Tanaka (2007), Del Moral, Doucet, and Jasra (2012), and Peters, Fan, and Sisson (2012).

It is important to be aware that ABC makes use of both finite tolerances and summary statistics. If the latter lack the sufficiency property this also introduces approximation error which does not go away with increased simulation effort and which can be extremely difficult to quantify or understand. Although the method is appealing in its simplicity and broad applicability, as statisticians we should be careful to understand any approximations involved in our computations.

There is work on the use of ABC within a model selection context. Early algorithms include those of Del Moral, Doucet, and Jasra (2012). Characterisation of sufficient statistics for model choice by ABC can be found in Grelaud et al. (2009), Didelot et al. (2011), and Robert et al. (2011), while Marin et al. (2014) characterises the properties required in order for *insufficient* summary statistics to provide asymptotically consistent Bayes factor (and hence model selection).

Chapter 5

Current and Future Directions

Monte Carlo methodology is being actively developed. Indeed it is is likely that many of the students attending this module will themselves be working on aspects of *computer intensive statistics*. No module of this sort can hope to start from the beginning of the discipline and reach all of the frontiers of current research. This chapter contains a very few words about some current research directions and attempts to provide references so that the interested reader can easily find out more. It isn't an exhaustive summary of interesting directions in this area, but we have attempted to provide a little information about at least the most widespread such topics (and, of course, those in which we are particularly interested).

5.1 Ensemble-based Methods and Sequential Monte Carlo

Ensemble- (or *particle-*) based methods use a *collection* of samples to approximate a distribution within an algorithm. Operations are performed on the ensemble rather than considering only a single sample at a time as for MCMC algorithms. Many of these methods come originally from the signal-processing literature, in which a class of algorithms known as *particle filters* were introduced by Gordon, Salmond, and Smith (1993) to approximate the solution of the discrete time filtering problem. Consider the problem of inferring the location of an object using noisy radar measurements taken at regular times. The filtering problem in this setting asks for the posterior distribution of the true location of the object (which is treated as a hidden or latent variable) at a time t given the noisy observations up to time t. A particle filter approximates this distribution by simulating a collection of particles whose weighted empirical distribution hopefully provides a good approximation to this *filtering distribution*. Particles are evolved through time and reweighted in order to update the filtering distribution as new observations arrive. See Doucet and Johansen (2011) for a survey of these and some related techniques.

Amongst others, R. M. Neal (2001) and Chopin (2001) proposed algoriothms based around this type of methodology (from quite different perspectives) which are applicable to more general problems. The particle filtering framework can be generalised to a rather more abstract problem of sampling from a growing collection of measures, in which case the algorithm is known as *Sequential Monte Carlo*. See Del Moral, Doucet, and Jasra (2006) for an important special case known as *Sequential Monte Carlo samplers*, and Del Moral (2004), Del Moral (2013), for book-length studies of the theoretical behaviour of this type of algorithm.

5.2 Pseudomarginal Methods and Particle MCMC

One area which has attracted a lot of attention is that in which one has access to a joint distribution but is interested in inference for only a (relatively low-dimensional) marginal of that distribution. It was demonstrated in Beaumont (2003) that with a clever spatial-extension scheme one could justify an approximation to the ideal marginal scheme. Such an approach was further analysed by Andrieu and Roberts (2009) (and there is a trail of more recent work) who termed them *pseudo-marginal* methods. The idea is motivated as follows. Suppose you wanted to run an MCMC algorithm on a model parameter but you could not compute the acceptance probability at each step because the likelihoods appearing in the numerator and denominator of the Metropolis–Hastings ratio are intractable. However, at some additional cost you are able to obtain a Monte Carlo estimate for each likelihood. If you were to simply plug in those estimates into the Metropolis–Hastings ratio, you have no guarantees that the MCMC algorithm still has the correct target distribution. The essence of the pseudo-marginal method is to view the act of doing Monte Carlo estimation as an instance of data augmentation; the augmentation comes from all the extra randomness you need to do the Monte Carlo step. By writing down this augmentation explicitly and treating the randomness associated with the Monte Carlo steps carefully, one is able to show that the MCMC algorithm still has the correct target.

A closely related idea is the particle MCMC (PMCMC) approach of Andrieu, Doucet, and Holenstein (2010). Here, SMC algorithms are used within an MCMC algorithm to integrate out large collections of latent variables. A number of schemes can be justified based upon a common extended-space view of these algorithms.

5.3 Approximate Approximate Methods

Pseudomarginal and related methods are often referred to as *exact approximate* methods: they are approximate in the sense that they emulate an idealised algorithm by invoking additional randomness, but are exact in the sense that they retain the correct target distribution. In recent years there has been some interest in making still further approximations and considering approximation schemes which also affect the target distribution. There is some weak theoretical support that such methods *can* work under certain circumstances (Alquier et al. 2016; Medina-Aguayo, Lee, and Roberts 2016; Everitt et al. 2017 for example) but it remains difficult to justify their use in realistic settings.

5.4 Quasi-Monte Carlo

In Section 2.1.1 the idea of quasi-random number generators was briefly mentioned. The use of quasi-random numbers within simulation procedures has received a burst of recent attention in large part due to the paper of Mathieu Gerber and Chopin (2015) which presented an elegant approach to their incorporation within the SMC framework. See also M. Gerber, Chopin, and Whiteley (2019) for further connections between quasi-Monte Carlo and SMC.

5.5 Hamiltonian/Hybrid MCMC

Hamiltonian Monte Carlo (HMC) is another approach to constructing MCMC chains which gives the evolution of those chains a physical interpretation. We augment the state space to incorporate a *momentum* variable and use some clever numerical technology borrowed from the physics literature—like the HMC method itself—in order to produce transitions which can exhibit much better mixing than the small steps allowed by Metropolis–Hastings type algorithms in high dimensions. Techniques adapted to a statistical setting in order to make use of higher order spatial structure was developed by Girolami and Calderhead (2011). See R. Neal (2011) for an introduction to these methods.

5.6 Methods for Big Data

An enormous research effort is currently being dedicated to the development of methods which scale sufficiently well with the size of a set of data that they allow inference with truly enormous data sets. This is too large, and too specialised, an area to dedicate much space to here, but Bardenet, Doucet, and Holmes (2017) provide an excellent comparative summary of the current state of the art.

Standard MCMC algorithms suffer in high-dimensions partly because their reversible nature slows down exploration of the state space. There has been a flurry of recent work developing *non-reversible* MCMC algorithms with good scaling properties. Often they are best expressed as continuous-time algorithms in contrast to the discrete updates of standard MCMC. This has something of the flavour of HMC but they rely heavily on proposals driven by piecewise deterministic Markov processes; see Fearnhead et al. (2018) for an introduction.

Appendix A

Some Markov Chain Concepts

This appendix is provided largely to make these notes self contained and to provide a little context and some details for those who want them. The notion of a stochastic process in general and Markov chains in particular are, of course, explored in more depth during the *Applied Stochastic Processes* module. No significant amount of lecture time will be dedicated to this material, and if this is all unfamiliar to you then you'll be able to engage with the lectures and the module without becoming intimately acquainted with the fine details of this material.

I've attempted to balance the need for technical rigour with accessibility and have avoided making much explicit reference to the theory of measure. If you aren't familiar with measure theory then you should be able to read this appendix by simply ignoring any reference to *measurability* but be aware that should you go on to use these concepts in the wild that we do need to be careful about such things.

A.1 Stochastic Processes

For our purposes we can define an E-valued *process* as a function $\xi : \mathcal{I} \to E$ which maps values in some index set $\mathcal{I}$ to some other space E. The evolution of the process is described by considering the variation of $\xi(i)$ with i. An E-valued *stochastic process* (or *random process*) can be viewed as a process in which, for each $i \in \mathcal{I}$, $\xi(i)$ is a random variable taking values in E.

Although a rich literature on more general situations exists, we will consider only the case of *discrete time stochastic processes* in which the index set $\mathcal{I}$ is $\mathbb{N}$ (of course, any index set isomorphic to $\mathbb{N}$ can be used in the same framework by simple relabeling). We will use the notation ξ_i to indicate the value of the process at *time* i (note that there need be no connection between the index set and *real* time, but this terminology is both convenient and standard).

We will begin with an extremely brief description of a general stochastic process, before moving on to discuss the particular classes of process in which we will be interested. In order to characterise a stochastic process of the sort in which we are interested, it is sufficient to know all of its *finite-dimensional distributions*, the joint distributions of the process at any collection of finitely many times. For any collection of times $i_1, i_2, \ldots, i_t$ and any *measurable* collection of subsets of E, $A_{i_1}, A_{i_2}, \ldots, A_{i_t}$ we are interested in the probability:

$$\mathbb{P}\left(\xi_{i_1} \in A_{i_1}, \xi_{i_2} \in A_{i_2}, \ldots, \xi_{i_t} \in A_{i_t}\right).$$

For such a collection of probabilities to define a stochastic process, we require that they meet a certain *consistency* criterion. We require the marginal distribution of the values taken by the process at any collection of times to be the same under any finite-dimensional distribution which includes the process at those time points, so, defining any second collection of times $j_1, \ldots, j_s$ with the property that $j_k \neq i_l$ for any $k \leq t, l \leq s$, we must have that:

$$\mathbb{P}\left(\xi_{i_1} \in A_{i_1}, \xi_{i_2} \in A_{i_2}, \ldots, \xi_{i_t} \in A_{i_t}\right) = \mathbb{P}\left(\xi_{i_1} \in A_{i_1}, \xi_{i_2} \in A_{i_2}, \ldots, \xi_{i_t} \in A_{i_t}, \xi_{j_1} \in E, \ldots, \xi_{j_s} \in E\right).$$

This is just an expression of the intuitive concept that any finite-dimensional distribution which describes the process at the times of interest should provide the same description if we neglect any information it provides about the process at other times. Or, to put it another way, they must all be marginal distributions of *the same* distribution.

In the case of real-valued stochastic processes, in which $E = \mathbb{R}$, we may express this concept in terms of the joint distribution functions (the multivariate analogue of the distribution function). Defining the joint distribution functions according to:

$$F_{i_1,\dots,i_t}(x_1, x_2, \dots, x_t) = \mathbb{P}\left(\xi_{i_1} \leq x_1, \xi_{i_2} \leq x_2, \dots, \xi_{i_t} \leq x_t\right),$$

our consistency requirement may now be expressed as:

$$F_{i_1,\dots,i_t,j_1,\dots,j_s}(x_1, x_2, \dots, x_t, \infty, \dots, \infty) = F_{i_1,\dots,i_t}(x_1, x_2, \dots, x_t).$$

Having established that we can specify a stochastic process if we are able to specify its finite-dimensional distributions, we might wonder how to specify these distributions. In the next two sections, we proceed to describe a class of stochastic processes which can be described constructively and whose finite-dimensional distributions may be easily established. The *Markov processes* which we are about to introduce represent the most widely used class of stochastic processes, and the ones which will be of most interest in the context of Monte Carlo methods.

A.2 Discrete State Space Markov Chains

A.2.1 Basic Notions

We begin by turning our attention to the discrete state space case which is somewhat easier to deal with than the general case which will be of interest later. In the case of discrete state spaces, in which $|E|$ is either finite, or countably infinite, we can work with the actual probability of the process having a particular value at any time (you'll recall that in the case of continuous random variables more subtlety is generally required as the probability of any continuous random variable defined by a density (with respect to Lebesgue measure, in particular) taking any particular value is zero). This simplifies things considerably, and we can consider defining the distribution of the process of interest over the first t time points by employing the following decomposition:

$$\begin{aligned}\mathbb{P}\left(\xi_1 = x_1, \xi_2 = x_2, \dots, \xi_t = x_t\right) =& \mathbb{P}\left(\xi_1 = x_1, \xi_2 = x_2, \dots, \xi_{t-1} = x_{t-1}\right) \\ & \times \mathbb{P}\left(\xi_t = x_t | \xi_1 = x_1, \dots, \xi_{t-1} = x_{t-1}\right).\end{aligned}$$

Looking at this decomposition, it's clear that we could construct all of the distributions of interest from an initial distribution from which ξ_1 is assumed to be drawn and then a sequence of conditional distributions for each t, leading us to the specification:

$$\mathbb{P}\left(\xi_1 = x_1, \xi_2 = x_2, \dots, \xi_t = x_t\right) = \mathbb{P}\left(\xi_1 = x_1\right) \prod_{i=2}^{t} \mathbb{P}\left(\xi_i = x_i | \xi_1 = x_1, \dots, \xi_{i-1} = x_{i-1}\right). \tag{A.1}$$

From this specification we can trivially construct all of the finite-dimensional distributions using no more than the sum and product rules of probability.

So, we have a method for constructing finite-dimensional distributions for a discrete state space stochastic process, but it remains a little formal as the conditional distributions seem likely to become increasingly complex as the time index increases. The conditioning present in decomposition (A.1) is needed to capture any relationship between the distribution at time t and *any* previous time. In many situations of interest, we might expect interactions to exist on only a much shorter time-scale. Indeed, one could envisage a *memoryless* process in which the distribution of the state at time $t+1$ depends only upon its state at time t, ξ_t, regardless of the path by which it reached ξ_t. Formally, we could define such a process as:

$$\mathbb{P}\left(\xi_1 = x_1, \xi_2 = x_2, \dots, \xi_t = x_t\right) = \mathbb{P}\left(\xi_1 = x_1\right) \prod_{i=2}^{t} \mathbb{P}\left(\xi_i = x_i | \xi_{i-1} = x_{i-1}\right). \tag{A.2}$$

It is clear that (A.2) is a particular case of (A.1) in which this lack of memory property is captured explicitly, as:

$$\mathbb{P}\left(\xi_t = x_t | \xi_1 = x_1, \dots, \xi_{t-1} = x_{t-1}\right) = \mathbb{P}\left(\xi_t = x_t | \xi_{t-1} = x_{t-1}\right).$$

We will take this as the defining property of a collection of processes which we will refer to as discrete time *Markov processes* or, as they are more commonly termed in the Monte Carlo literature, *Markov chains*. There is some debate in the literature as to whether the term “Markov chain” should be reserved for those Markov processes which take place on a discrete state space, those which have a discrete index set (the only case we will consider here) or both. As is common in the field of Monte Carlo simulation, we will use the terms Markov chain and Markov process interchangeably.

When dealing with discrete state spaces, it is convenient to associate a row vector[1] with any probability distribution. We assume, without loss of generality, that the state space, E, is $\mathbb{N}$. Now, given a random variable X on E, we say that X has distribution μ, often written as $X \sim \mu$ for some vector μ with the property that:

$$\forall x \in E : \ \mathbb{P}\left(X = x\right) = \mu_x.$$

A.2.1.1 Homogeneous Markov Chains

The term *homogeneous Markov Chain* is used to describe a Markov process of the sort just described with the additional caveat that the conditional probabilities do not depend explicitly on the time index, so:

$$\forall m \in \mathbb{N} : \ \mathbb{P}\left(\xi_t = y | \xi_{t-1} = x\right) \equiv \mathbb{P}\left(\xi_{t+m} = y | \xi_{t+m-1} = x\right).$$

In this setting, it is particular convenient to define a function corresponding to the *transition probability* (as the probability distribution at time $t+1$ conditional upon the state of the process at time t) or *kernel* as it is often known, which may be written as a two argument function or, in the discrete case as a matrix, $K(i,j) = K_{ij} = \mathbb{P}\left(\xi_t = j | \xi_{t-1} = i\right)$.

Having so expressed things, we are able to describe the dynamic structure of a discrete state space, discrete time Markov chain in a particularly simple form. If we allow μ_t to describe the distribution of the chain at time t, so that $\mu_{t,i} = \mathbb{P}\left(\xi_t = i\right)$, then we have by applying the sum and product rules of probability, that:

$$\mu_{t+1,j} = \sum_i \mu_{t,i} K_{ij}.$$

We may recognise this as standard vector-matrix multiplication and write simply that $\mu_{t+1} = \mu_t K$ and, proceeding inductively it’s straightforward to verify that $\mu_{t+m} = \mu_t K^m$ where K^m denotes the usual m^{th} matrix power of K. We will make some use of this object, as it characterises the m-step ahead condition distribution:

$$K^m_{ij} := (K^m)_{ij} = \mathbb{P}\left(\xi_{t+m} = j | \xi_t = i\right).$$

In fact, the initial distribution μ_1, together with K tells us the full distribution of the chain over any finite time horizon:

$$\mathbb{P}\left(\xi_1 = x_1, \ldots, \xi_t = x_t\right) = \mu_{1,x_1} \prod_{i=2}^{t} K_{x_{i-1} x_i}.$$

A general stochastic processes is said to possess the *weak Markov property* if, for any deterministic time, t and any finite integer p, we may write that for any integrable function $\varphi : E \to \mathbb{R}$:

$$\mathbb{E}\left[\varphi(\xi_{t+p}) | \xi_1 = x_1, \ldots \xi_t = x_t\right] = \mathbb{E}\left[\varphi(\xi_{t+p}) | \xi_t = x_t\right].$$

A.2.1.2 Inhomogeneous Markov Chains

Note that it is perfectly possible to define Markov Chains whose behaviour does depend explicitly upon the time index. Although such processes are more complex to analyse than their homogeneous counterparts, they do play a rôle in Monte Carlo methodology—in both established algorithms such as simulated annealing (see Section 3.5 and in more recent developments such as adaptive Markov Chain Monte Carlo and the State Augmentation for Maximising Expectations (SAME) algorithm of Doucet, Godsill, and Robert (2002). In the interests of simplicity, what follows is presented for homogeneous Markov Chains.

[1] Formally, much of the time this will be an infinite-dimensional vector but this need not concern us here.

A.2.1.3 Examples

Before moving on to introduce some theoretical properties of discrete state space Markov chains we will present a few simple examples. Whilst there are innumerable examples of homogeneous discrete state space Markov chains, we confined ourselves here to some particular simple cases which will be used to illustrate some properties below, and which will probably be familiar to you.

We begin with an example which is apparently simple, and rather well known, but which exhibits some interesting properties.

Example A.1. (Simple random walk over the integers). *Given a process ξ_t whose value at time $t+1$ is $\xi_t + 1$ with probability p_+ and ξ_{t-1} with probability $p_- = 1 - p_+$, we obtain the familiar random walk. We may write this as a Markov chain by setting $E = \mathbb{Z}$ and noting that the transition kernel may be written as:*

$$K_{ij} = \begin{cases} p_- & \textit{if } j = i-1 \\ p_+ & \textit{if } j = i+1 \\ 0 & \textit{otherwise.} \end{cases}$$

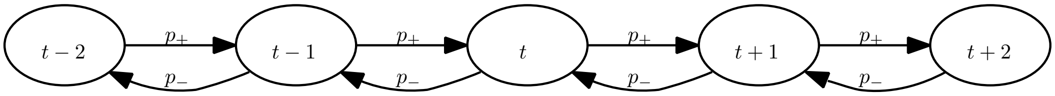

Figure A.1: A simple random walk on $\mathbb{Z}$.

Example A.2. *It will be interesting to look at a slight extension of this random walk, in which there is some probability p_0 of remaining in the present state at the next time step, so $p_+ + p_- < 0$ and $p_0 = 1 - (p_+ + p_-)$. In this case we may write the transition kernel as:*

$$K_{ij} = \begin{cases} p_- & \textit{if } j = i-1 \\ p_0 & \textit{if } j = i \\ p_+ & \textit{if } j = i+1 \\ 0 & \textit{otherwise.} \end{cases}$$

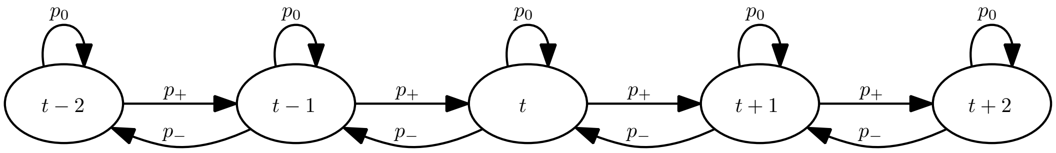

Figure A.2: A random walk on $\mathbb{Z}$ with $K_{tt} > 0$.

Example A.3. (Random Walk on a Triangle). *A third example which we will consider below could be termed a "random walk on a triangle". In this case, we set $E = \{1, 2, 3\}$ and define a transition kernel of the form:*

$$K = \begin{bmatrix} 0 & p_+ & p_- \\ p_- & 0 & p_+ \\ p_+ & p_- & 0 \end{bmatrix}.$$

Example A.4. (One-sided Random Walk). *Finally, we consider the rather one-sided random walk on the positive integers, illustrated in Figure A.4, and defined by transition kernel:*

$$K_{ij} = \begin{cases} p_0 & \textit{if } j = i \\ p_+ = 1 - p_0 & \textit{if } j = i+1 \\ 0 & \textit{otherwise.} \end{cases}$$

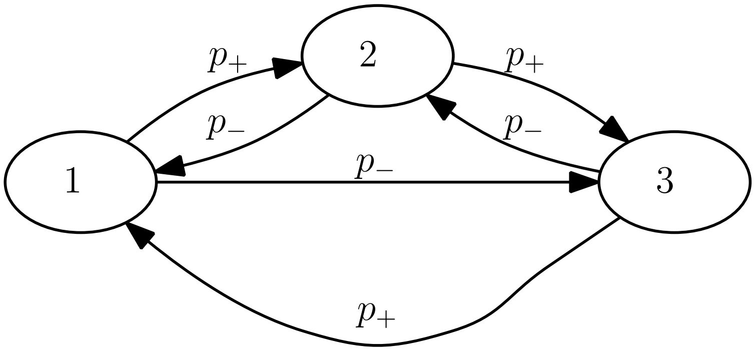

Figure A.3: A random walk on a triangle.

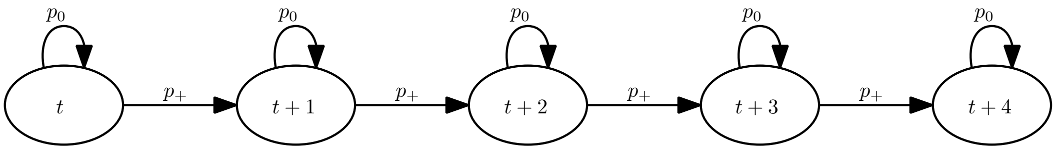

Figure A.4: A random walk on the positive integers.

A.2.2 Important Properties

In this section we introduce some important properties in the context of discrete state space Markov chains and attempt to illustrate their importance within the field of Monte Carlo simulation. As is the usual practice when dealing with this material, we will restrict our study to the homogeneous case. As you will notice, it is the transition kernel which is most important in characterising a Markov chain.

We begin by considering how the various states that a Markov chain may be reached from one another. In particular, the notion of states which *communicate* is at the heart of the study of Markov chains.

Definition A.1. (Accessibility). *A state y is* ***accessible*** *from a state x, sometimes written as $x \to y$ if, for a discrete state space Markov chain,*

$$\inf \{t : \mathbb{P}(\xi_t = y | \xi_1 = x) > 0\} < \infty.$$

We can alternatively write this condition in terms of the transition matrix as $\inf \left\{t : K^t_{xy} > 0\right\} < \infty$.

This concept tells us which states one can reach at some finite time in the future, if one starts from a particular state and then moves, at each time, according to the transition kernel, K. That is, if $x \to y$, then there is a positive probability of reaching y at some finite time in the future, if we start from a state x and then "move" according to the Markov kernel K. It is now useful to consider cases in which one can traverse the entire space, or some subset of it, starting from any point.

Definition A.2. (Communication). *Two states $x, y \in E$ are said to* ***communicate*** *(written, by some authors as $x \leftrightarrow y$) if each is accessible from the other, that is:*

$$x \leftrightarrow y \Leftrightarrow x \to y \textit{ and } y \to x.$$

We're now in a position to describe the relationship, under the action of a Markov kernel, between two states. This allows us to characterise something known as the **communication structure** of the associated Markov chain to some degree, noting which points its possible to travel both to and back from. We now go on to introduce a concept which will allow us to describe the properties of the full state space, or significant parts of it, rather than individual states.

Definition A.3. (Irreducibility). *A Markov Chain is said to be* ***irreducible*** *if all states communicate, so $\forall x, y \in E : x \to y$. Given a distribution ϕ on E, the term ϕ-irreducible is used to describe a Markov chain for which every state with positive probability under ϕ communicates with every other such state:*

$$\forall x, y \in supp(\phi) : x \to y$$

where the support of the discrete distribution ϕ is defined as supp$(\phi) = \{x \in E : \phi(x) > 0\}$. It is said to be strongly irreducible if any state can be reached from any point in the space in a single step and strongly ϕ-irreducible if all states (except for a collection with probability 0 under ϕ) may be reached in a single step.

This will prove to be important for the study of Monte Carlo methods based upon Markov chains as a chain with this property can somehow explore the entire space rather than being confined to some portion of it, perhaps one which depends upon the initial state.

It is also important to consider the type of routes which it is possible to take between a state, x, and itself as this will tell us something about the presence of long-range correlation between the states of the chain.

Definition A.4. (Period). *A state x in a discrete state space Markov chain has period $d(x)$ defined as:*

$$d(x) = \gcd \left\{s \geq 1 : K^s_{xx} > 0\right\},$$

where gcd *denotes the greatest common denominator. A chain possessing such a state is said to have a **cycle of length** d.*

Proposition A.1. All states which communicate have the same period and hence, in an irreducible Markov chain, all states have the same period.

Proof. Assume that $x \leftrightarrow y$. Let there exist paths of lengths r, s and t, respectively from $x \to y$, $y \to x$ and $y \to y$, respectively.

There are paths of length $r + s$ and $r + s + t$ from x to x, hence $d(x)$ must be a divisor of $r + s$ and $r + s + t$ and consequently of their difference, t. This holds for any t corresponding to a path from $y \to y$ and so $d(x)$ is a divisor of the length of any path from $y \to y$: as $d(y)$ is the greatest common divisor of all such paths, we have that $d(x) \leq d(y)$.

By symmetry, we also have that $d(y) \leq d(x)$, and this completes the proof. □

In the context of irreducible Markov chains, the term *periodic* is used to describe those chains whose states have some common period great than 1, whilst those chains whose period is 1 are termed *aperiodic*.

One further quantity needs to be characterised in order to study the Markov chains which will arise later. Some way of describing *how many times* a state is visited if a Markov chain is allowed to run for infinite time still seems required. In order to do this it is useful to define an additional random quantity, the number of times that a state is visited:

$$\eta_x := \sum_{k=1}^{\infty} \mathbb{I}_x(\xi_k).$$

We will also adopt the convention, common in the Markov chain literature that, given any function of the path of a Markov chain, φ, $\mathbb{E}_x[\varphi]$ is the expectation of that function under the law of the Markov chain initialised with $\xi_1 = x$. Similarly, if μ is some distribution over E, then $\mathbb{E}_\mu[\varphi]$ should be interpreted as the expectation of ϕ under the law of the process initialised with $\xi_1 \sim \mu$.

Definition A.5. (Transience and Recurrence). *In the context of discrete state space Markov chains, we describe a state, x, as **transient** if:*

$$\mathbb{E}_x[\eta_x] < \infty,$$

whilst, if we have that,

$$\mathbb{E}_x[\eta_x] = \infty,$$

*then that state will be termed **recurrent**.*

In the case of irreducible Markov chains, transience and recurrence are properties of the chain itself, rather than its individual states: if any state is transient (or recurrent) then all states have that property. Indeed, for an irreducible Markov chain either all states are recurrent or all are transient.

We will be particularly concerned in this course with Markov kernels which admit an invariant distribution.

Definition A.6. (Invariant Distribution). *A distribution, μ is said to be **invariant** or **stationary** for a Markov kernel, K, if $\mu K = \mu$.*

If a Markov chain has any single time marginal distribution which corresponds to its stationary distribution, $\xi_t \sim \mu$, then all of its future time marginals are the same as, $\xi_{t+s} \sim \mu K^s = \mu$. A Markov chain is said to be in its stationary regime once this has occurred. Note that this tells us nothing about the correlation between the states or their joint distribution. One can also think of the invariant distribution μ of a Markov kernel, K as the *left eigenvector* with unit eigenvalue.

Definition A.7. (Reversibility). *A stationary stochastic process is said to be reversible if the statistics of the time-reversed version of the process match those of the process in the forward distribution, so that reversing time makes no discernible difference to the sequence of distributions which are obtained, that is the distribution of any collection of future states given any past history must match the conditional distribution of the past conditional upon the future being the reversal of that history.*

Reversibility is a condition which, if met, simplifies the analysis of Markov chains. It is normally verified by checking the detailed balance condition, (A.3). If this condition holds for a distribution, then it also tells us that this distribution is the stationary distribution of the chain, another property which we will be interested in.

Proposition A.2. If a Markov kernel satisfies the **detailed balance condition** for some distribution μ,

$$\forall x, y \in E : \ \mu_x K_{xy} = \mu_y K_{yx} \tag{A.3}$$

then:

1. μ is the invariant distribution of the chain.
2. The chain is reversible with respect to μ.

Proof. To demonstrate that K is μ-invariant, consider summing both sides of the detailed balance equation over x:

$$\sum_{x \in E} \mu_x K_{xy} = \sum_{x \in E} \mu_y K_{yx},$$
$$(\mu K)_y = \mu_y,$$

and as this holds for all y, we have $\mu K = \mu$.

In order to verify that the chain is reversible we proceed directly:

$$\begin{aligned}\mathbb{P}\left(\xi_t = x | \xi_{t+1} = y\right) &= \frac{\mathbb{P}\left(\xi_t = x, \xi_{t+1} = y\right)}{\mathbb{P}\left(\xi_{t+1} = y\right)} = \frac{\mathbb{P}\left(\xi_t = x\right) K_{xy}}{\mathbb{P}\left(\xi_{t+1} = y\right)} = \frac{\mu_x K_{xy}}{\mu_y} = \frac{\mu_y K_{yx}}{\mu_y} = K_{yx} \\ &= \mathbb{P}\left(\xi_t = x | \xi_{t-1} = y\right),\end{aligned}$$

in the case of a Markov chain it is clear that if the transitions are time-reversible then the process must be time reversible. □

A.3 General State Space Markov Chains

A.3.1 Basic Concepts

The study of general state space Markov chains is a complex and intricate business. To do so entirely rigorously requires a degree of technical sophistication which lies somewhat outside the scope of this course. Here, we will content ourselves with explaining how the concepts introduced in the context of discrete state spaces in the previous section might be extended to continuous domains via the use of probability densities. We will not consider more complex cases—such as mixed continuous and discrete spaces, or distributions over uncountable spaces which may not be described by a density. Nor will we provide proofs of results for this case, but will provide suitable references for the interested reader.

Although the guiding principles are the same, the study of Markov chains with continuous state spaces requires considerably more subtlety as it is necessary to introduce concepts which correspond to those which we introduced in the discrete case, describe the same properties and are motivated by the same intuition but which remain meaningful when we are dealing with densities rather than probabilities. As always, the principal complication is that the probability of any random variable distributed according to a non-degenerate density on a continuous state space taking any particular value is formally zero.

We will begin by considering how to emulate the decomposition we used to define a Markov chain on a discrete state space, Equation (A.2), when E is a continuous state space. In this case, what we essentially require is that the probability of any range of possible values, given the entire history of the process depends only upon its most recent value in the sense that, for any measurable $A_t \subseteq E$:

$$\mathbb{P}\left(\xi_t \in A_t | \xi_1 = x_1, \ldots, \xi_{t-1} = x_{t-1}\right) = \mathbb{P}\left(\xi_t \in A_t | \xi_{t-1} = x_{t-1}\right).$$

In the case which we are considering, it is convenient to describe the distribution of a random variable over E in terms of some probability density, $\mu : E \to \mathbb{R}$ which has the property that, if integrated over any measurable set, it tells us the probability that the random variable in question lies within that set, i.e. if $X \sim \mu$, we have that for any measurable set A that:

$$\mathbb{P}\left(X \in A\right) = \int\limits_A \mu(x) dx.$$

We will consider only the homogeneous case here, although the generalisation to inhomogeneous Markov chains follows in the continuous setting in precisely the same manner as the discrete one. In this context, we may describe the conditional probabilities of interest as a function $K : E \times E \to \mathbb{R}$ which has the property that for all measurable sets $A \subseteq E$ and all points $x \in E$:

$$\mathbb{P}\left(\xi_t \in A | X_{t-1} = x\right) = \int\limits_A K(x, y) dy.$$

We note that, as in the discrete case the law of a Markov chain evaluated at any finite number of points may be completely specified by the initial distribution, call it μ, and a transition kernel, K. We have, for any suitable collection of sets $A_1, \ldots$, that the following holds:

$$\mathbb{P}\left(\xi_1 \in A_1, \ldots, \xi_t \in A_t\right) = \int\limits_{A_1 \times \cdots \times A_t} \mu(x_1) \prod_{k=2}^{t} K_k(x_{k-1}, x_k) dx_1 \ldots dx_t.$$

And, again, it is useful to be able to consider the s-step ahead conditional distributions,

$$\mathbb{P}\left(\xi_{t+s} \in A | \xi_t = x_t\right) = \int\limits_{E^{s-1} \times A} \prod_{k=t+1}^{k=t+s} K(x_{k-1}, x_k) dx_{t+1} \ldots dx_{t+s},$$

and it is useful to define an s-step ahead transition kernel in the same manner as it is in the discrete case, here matrix multiplication is replaced by a convolution operation but the intuition remains the same. Defining

$$K^s(x_t, x_{t+s}) := \int\limits_{E^{s-1}} \prod_{k=t+1}^{k=t+s} K(x_{k-1}, x_k) dx_{t+1} \ldots dx_{t+s-1},$$

we are able to write

$$\mathbb{P}\left(\xi_{t+s} \in A | \xi_t = x_t\right) = \int\limits_A K^s(x_t, x_{t+s}) dx_{t+s}.$$

A.3.2 Important Properties

In this section we will introduce properties which fulfill the same rôle in context of continuous state spaces as those introduced in Section @ref(#secmcdiscreteproperties) do in the discrete setting.

Whilst it is possible to define concepts similar to communication and accessibility in a continuous state space context, this isn't especially productive. We are more interested in the property of *irreducibility*: we want some way of determining what class of states are reachable from one another and hence what part of E might be explored, with positive probability, starting from a point within such a class. We will proceed directly to a continuous state space definition of this concept.

Definition A.8. (Irreducibility). *Given a distribution, μ, over E, a Markov chain is said to be **μ-irreducible** if for all points $x \in E$ and all measurable sets A such that $\mu(A) > 0$ there exists some t such that:*

$$\int_A K^t(x,y)dy > 0.$$

*If this condition holds with $t = 1$, then the chain is said to be **strongly μ-irreducible**.*

This definition has the same character as that employed in the discrete case, previously, but is well defined for more general state spaces. It still tells us whether a chain is likely to be satisfactory if we are interested in approximation of some property of a measure μ by using a sample of the evolution of that chain: if it is *not* μ-irreducible then there are some points in the space from which we cannot reach all of the support of μ, and this is likely to be a problem. In the sequel we will be interested more of less exclusively with Markov chains which are irreducible with respect to some measure of interest.

We need a little more subtlety in extending some of the concepts introduced in the case of discrete Markov chains to the present context. In order to do this, it will be necessary to introduce the concept of the *small set*; these function as a replacement for the individual states of a discrete space Markov chain as we will see shortly.

A first attempt might be to consider the following sets which have the property that the distribution of taken by the Markov chain at time $t + 1$ is the same if it starts at any point in this set—so the conditional distribution function is constant over this set.

Definition A.9. (Atoms). *A Markov chain with transition kernel K is said to have an atom, $\alpha \subseteq E$, if there is some probability distribution, ν, such that:*

$$\forall x \in \alpha, A \subseteq E : \quad \int_A K(x,y)dy = \int_A \nu(y)dy.$$

*If the Markov chain in question is ν-irreducible, then α is termed an **accessible atom**.*

Whilst the concept of *atoms* starts to allow us to introduce some sort of structure similar to that seen in discrete chains—it provides us with a set of positive probability which, if the chain ever enters it, we know the distribution of the subsequent state. Note that this is much stronger than knowledge of the transition kernel, K, as in general all points in the space have zero probability. Most interesting continuous state spaces do not possess atoms. The condition that the distribution of the next state is precisely the same, wherever the current state is rather strong. Another approach would be to require only that the conditional distribution has a common component, and that is the intuition behind a much more useful concept which underlies much of the analysis of general state space Markov chains.

Definition A.10. (Small Sets). *A set, $C \subseteq E$, is termed **small** for a given Markov chain (or, when one is being precise, (ν, s, ϵ)-small) if there exists some positive integer s, some $\epsilon > 0$ and some non-trivial probability distribution, ν, such that:*

$$\forall x \in C, A \subseteq E : \quad \int_A K^s(x,y)dy \geq \epsilon \int_A \nu(y)dy.$$

This tells us that the distribution s-steps after the chain enters the small set has a component of size at least ϵ of the distribution ν, wherever it was within that set. In this sense, small sets are not "too big": there is potentially some commonality of all paths emerging from them. Although we have not proved that such sets exist for any particular class of Markov chains it is, in fact, the case that they do for many interesting Markov chain classes and their existence allows for a number of sophisticated analytic techniques to be applied.

In order to define cycles (and hence the notion of periodicity) in the general case, we require the existence of a small set. We need some group of "sufficiently similar" points in the state space which have a finite probability of being reached. We then treat this collection of points in the same manner as an individual state in the discrete case, leading to the following definitions.

Definition A.11. (Cycles). *A μ-irreducible Markov chain has a **cycle of length** d if there exists a (ν, M, ϵ)-small set C, such that:*

$$d = \gcd \left\{ s \geq 1 : C \textit{ is } (\nu, s, \delta_s \epsilon)\textit{-small for some } \delta_s > 0 \right\}.$$

This provides a reasonable concept of periodicity within a general state space Markov chain as it gives us a way of characterising the existence of regions of the space with the property that, wherever you start within that region you have positive probability of returning to that set after any multiple of d steps and this *does not* hold for any number of steps which is not a multiple of d. We are able to define periodicity and aperiodicity in the same manner as for discrete chains, but using this definition of a cycle. As in the discrete space, all states within the support of μ in a μ-irreducible chain must have the same period (see Proposition A.1) although we will not prove this here.

Considering periodicity from a different viewpoint, we are able to characterise it in a manner which is rather easier to interpret but somewhat difficult to verify in practice. The following definition of period is equivalent to that given above (Nummelin 1984): a Markov chain has a period d if there exists some partition of the state space, $E_1, \ldots, E_d$ with the properties that:

- $\forall i \neq j : \; E_i \cap E_j = \emptyset$,
- $\bigcup\limits_{i=1}^{d} E_i = E$,
- $\forall i, j, t, s : \; \mathbb{P}\left(X_{t+s} \in E_j | X_t \in E_i\right) = \begin{cases} 1 & j = i + s \bmod d \\ 0 & \text{otherwise.} \end{cases}$

What this actually tells us is that a Markov chain with a period of d has associated with it a disjoint partition of the state space, $E_1, \ldots, E_d$ and that we know that the chain moves with probability 1 from set E_1 to E_2, E_2 to E_3, E_{d-1} to E_d and E_d to E_1 (assuming that $d \geq 3$, of course). Hence the chain will visit a particular element of the partition with a period of d.

We also require some way of characterising how often a continuous state space Markov chain visits any particular region of the state space in order to obtain concepts analogous to those of transience and recurrence in the discrete setting. In order to do this we define a collection of random variables η_A for any subset A of E, which correspond to the number of times the set A is visited, i.e. $\eta_A := \sum_{l=1}^{\infty} \mathbb{I}_A(\xi_k)$ and, once again we use $\mathbb{E}_x\left[\cdot\right]$ to denote the expectation under the law of the Markov chain with initial state x. We note that if a chain is not μ-irreducible for some distribution μ, then there is no guarantee that it is either transient or recurrent, however, the following definitions do hold:

Definition A.12. (Transience and Recurrence). *We begin by defining* ***uniform transience*** *and* ***recurrence*** *for sets $A \subseteq E$ for μ-irreducible general state space Markov chains. Such a set is recurrent if:*

$$\forall x \in A : \; \mathbb{E}_x\left[\eta_A\right] = \infty.$$

A set is ***uniformly transient*** *if there exists some $M < \infty$ such that:*

$$\forall x \in A : \; \mathbb{E}_x\left[\eta_A\right] \leq M.$$

The weaker concept of transience of a set may then be introduced. A set, $A \subseteq E$, is ***transient*** *if it may be expressed as a countable union of uniformly transient sets, i.e.:*

$$\begin{aligned} \exists \left\{B_i \subseteq E\right\}_{i=1}^{\infty} : & \quad A \subseteq \bigcup_{i=1}^{\infty} B_i \\ \forall i \in \mathbb{N}, \forall x \in B_i : & \quad \mathbb{E}_x\left[\eta_{B_i}\right] \leq M_i < \infty. \end{aligned}$$

A general state space Markov chain is ***recurrent*** *if the following two conditions are satisfied:*

- *The chain is μ-irreducible for some distribution μ,*
- *For every measurable set $A \subseteq E$ such that $\int_A \mu(y)dy > 0$, $\mathbb{E}_x\left[\eta_A\right] = \infty$ for every $x \in A$;*

whilst it is ***transient*** *if it is μ-irreducible for some distribution μ and the entire space is transient.*

As in the discrete setting, in the case of irreducible chains, transience and recurrence are properties of the chain rather than individual states: all states within the support of the irreducibility distribution are either transient or recurrent. It is useful to note that any μ-irreducible Markov chain which has stationary distribution μ is positive recurrent (Tierney 1994).

A slightly stronger form of recurrence is widely employed in the proof of many theoretical results which underlie many applications of Markov chains to statistical problems, this form of recurrence is known as Harris recurrence and may be defined as follows:

Definition A.13. (Harris Recurrence). *A set $A \subseteq E$ is* ***Harris recurrent*** *if $\mathbb{P}_x(\eta_A = \infty) = 1$ for every $x \in A$.*

A Markov chain is ***Harris recurrent*** *if there exists some distribution μ with respect to which it is irreducible and every set A such that $\int_A \mu(x)dx > 0$ is Harris recurrent.*

The concepts of invariant distribution, reversibility and detailed balance are essentially unchanged from the discrete setting. It's necessary to consider integrals with respect to densities rather than sums over probability distributions, but no fundamental differences arise here.

A.4 Selected Theoretical Results

The probabilistic study of Markov chains dates back more than fifty years and comprises an enormous literature, much of it rather technically sophisticated. We don't intend to summarise that literature here, nor to provide proofs of the results which we present here. This section serves only to motivate the material presented in the subsequent chapters.

These two theorems fill the rôle which the law of large numbers and the central limit theorem for independent, identically distributed random variables fill in the case of simple Monte Carlo methods. They tell us, roughly speaking, that if we take the sample averages of a function at the points of a Markov chain which satisfies suitable regularity conditions and possesses the correct invariant distribution, then we have convergence of those averages to the integral of the function of interest under the invariant distribution and, furthermore, under stronger regularity conditions we can obtain a rate of convergence.

There are two levels of strength of law of large numbers which it is useful to be aware of. The first tells us that for most starting points of the chain a law of large numbers will hold. Under slightly stronger conditions (which it may be difficult to verify in practice) it is possible to show the same result holds for *all* starting points.

Theorem A.1. *(A Simple Ergodic Theorem).* If $(\xi_i)_{i\in\mathbb{N}}$ is a μ-irreducible, recurrent $\mathbb{R}^d$-valued Markov chain which admits μ as a stationary distribution, then the following strong law of large numbers holds (convergence is with probability 1) for any integrable function $\varphi : \mathbb{R}^d \to \mathbb{R}$:

$$\lim_{t\to\infty} \frac{1}{t}\sum_{i=1}^{t} \varphi(\xi_i) \overset{a.s.}{=} \int \varphi(x)\mu(x)dx.$$

for almost every starting value x (i.e. for any x except perhaps for a set of bad starting value, $\mathcal{N}$, which has the property that $\int_{\mathcal{N}} \mu(x)dx = 0$).

An outline of the proof of this theorem is provided by G. O. Roberts and Rosenthal (2004, Fact 5).

Theorem A.2. *(A Stronger Ergodic Theorem).* If $(\xi_i)_{i\in\mathbb{N}}$ is a μ-invariant, Harris recurrent Markov chain, then the following strong law of large numbers holds (convergence is with probability 1) for any integrable function $\varphi : E \to \mathbb{R}$:

$$\lim_{t\to\infty} \frac{1}{t}\sum_{i=1}^{t} \varphi(\xi_i) \overset{a.s.}{=} \int \varphi(x)\mu(x)dx.$$

A proof of this result is beyond the scope of the course. This is a particular case of Robert and Casella (2004, p241, Theorem 6.63), and a proof of the general theorem is given there. The same theorem is also presented with proof in Meyn and Tweedie (1993, p433, Theorem 17.3.2).

Theorem A.3. *(A Central Limit Theorem).* Under technical regularity conditions (see Jones (2004) for a summary of various combinations of conditions) it is possible to obtain a central limit theorem for the ergodic averages of a Harris recurrent, μ-invariant Markov chain, and a function $\varphi : E \to \mathbb{R}$ which has at least two finite moments (depending upon the combination of regularity conditions assumed, it may be necessary to have a finite moment of order $2 + \delta$).

$$\lim_{t\to\infty} \sqrt{t}\left[\frac{1}{t}\sum_{i=1}^{t} \varphi(\xi_i) - \int \varphi(x)\mu(x)dx\right] \overset{\mathcal{D}}{=} \mathsf{N}\left(0, \sigma^2(\varphi)\right),$$

$$\sigma^2(\varphi) = \mathbb{E}\left[(f(\xi_1) - \bar{\varphi})^2\right] + 2\sum_{k=2}^{\infty} \mathbb{E}\left[(\varphi(\xi_1) - \bar{\varphi})(\varphi(\xi_k) - \bar{\varphi})\right],$$

where $\bar{\varphi} = \int \varphi(x)\mu(x)dx$.

A.5 Further Reading

We conclude this chapter by noting that innumerable tutorials on the subject of Markov chains have been written, particularly with reference to their use in the field of Monte Carlo simulation. Some which might be of interest include the following:

- G. Roberts (1996) provides an elementary introduction to some Markov chain concepts required to understand their use in Monte Carlo algorithms.
- In the same volume, Tierney (1996) provides a more technical look at the same concepts; a more in-depth, but similar approach is taken by the earlier paper Tierney (1994).
- An alternative, elementary formulation of some of the material presented here together with some additional background material, aimed at an engineering audience, can be found in Johansen (2009).
- Robert and Casella (2004, chap. 6). This is a reasonably theoretical treatment intended for those interest in Markov chain Monte Carlo; it is reasonably technical in content, without dwelling on proofs. Those familiar with measure theoretic probability might find this a reasonably convenient place to start.
- Those of you interested in technical details might like to consult Meyn and Tweedie (1993). This is the definitive reference work on stability, convergence and theoretical analysis of Markov chains and it is now possible to download it, free of charge from the website of one of the authors.
- A less detailed, but more general and equally rigorous, look at Markov chains is provided by the seminal work of Nummelin (1984). This covers some material outside of the field of probability, but remains a concise work and presents only a few of the simpler results. It is perhaps a less intimidating starting point than Meyn and Tweedie (1993), although opinions on this vary.
- A survey of theoretical results relevant to Monte Carlo is provided by G. O. Roberts and Rosenthal (2004). Again, this is necessarily somewhat technical.

Alquier, P., N. Friel, R. Everitt, and A. Boland. 2016. "Noisy Monte Carlo: Convergence of Markov Chains with Approximate Transition Kernels." *Statistics and Computing* 26 (1): 29–47.

Andrieu, C., L. Breyer, and A. Doucet. 2001. "Convergence of Simulated Annealing Using Foster-Lyapunov Criteria." *Journal of Applied Probability* 38 (4): 975–94.

Andrieu, C., A. Doucet, and R. Holenstein. 2010. "Particle Markov Chain Monte Carlo." *Journal of the Royal Statistical Society B* 72 (3): 269–342.

Andrieu, C., and G. O. Roberts. 2009. "The Pseudo-Marginal Approach for Efficient Monte Carlo Computations." *Annals of Statistics* 37 (2): 697–725.

Bardenet, R., A. Doucet, and C. Holmes. 2017. "On Markov chain Monte Carlo Methods for Tall Data." *The Journal of Machine Learning Research* 18 (1): 1515–57.

Barnard, G. A. 1963. "Discussion of Prof. Bartlett's Paper." *Journal of the Royal Statistical Society B* 25 (2): 294.

Beaumont, M. 2003. "Estimation of Population Growth or Decline in Genetically Monitored Populations." *Genetics* 164 (3): 1139–60.

Besag, J., and P. Diggle. 1977. "Simple Monte Carlo Tests for Spatial Pattern." *Journal of the Royal Statistical Society C* 26 (3): 327–33.

Box, G. E. P., and M. E. Muller. 1958. "A Note on the Generation of Normal Random Deviates." *Annals of Mathematical Statistics* 29 (2): 610–11.

Brockwell, P. J., and R. A. Davis. 1991. *Time Series: Theory and Methods.* 2nd ed. New York: Springer.

Brooks, S., and A. Gelman. 1998. "General Methods for Monitoring Convergence of Iterative Simulations." *Journal of Computational and Graphical Statistics* 7 (4): 434–55.

Brooks, S., A. Gelman, G. L. Jones, and X.-L. Meng, eds. 2011. *Handbook of Markov Chain Monte Carlo.* CRC Press.

Chen, Y. 2005. "Another Look at Rejection Sampling Through Importance Sampling." *Statistics & Probability Letters* 72 (4): 277–83.

Chopin, N. 2001. "Sequential Inference and State Number Determination for Discrete State-Space Models Through Particle Filtering." Working Paper 2001-34. Laboratoire de Statistique, CREST, INSEE, Timbre J120, 75675 Paris cedex 14, France: CREST.

Davison, A. C., D. V. Hinkley, and G. A. Young. 2003. "Recent Developments in Bootstrap Methodology." *Statistical Science* 18 (2): 141–57.

Del Moral, P. 2004. *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*. Probability and Its Applications. New York: Springer Verlag.

———. 2013. *Mean Field Integration*. Chapman Hall.

Del Moral, P., A. Doucet, and A. Jasra. 2006. "Sequential Monte Carlo Samplers." *Journal of the Royal Statistical Society B* 63 (3): 411–36.

———. 2012. "An Adaptive Sequential Monte Carlo Method for Approximate Bayesian Computation." *Statistics and Computing* 22 (5): 1009–20.

Devroye, L. 1986. *Non-Uniform Random Variate Generation*. New York: Springer Verlag.

Didelot, X., R. G. Everitt, A. M. Johansen, and D. J. Lawson. 2011. "Likelihood-Free Estimation of Model Evidence." *Bayesian Analysis* 6 (1): 49–76.

Diggle, P. J., and R. J. Gratton. 1984. "Monte Carlo Methods of Inference for Implicit Statistical Models." *Journal of the Royal Statistical Society B* 46: 193–227.

Doucet, A., S. J. Godsill, and C. P. Robert. 2002. "Marginal Maximum a Posteriori Estimation Using Markov Chain Monte Carlo." *Statistics and Computing* 12: 77–84.

Doucet, A., and A. M. Johansen. 2011. "A Tutorial on Particle Filtering and Smoothing: Fiteen Years Later." In *The Oxford Handbook of Nonlinear Filtering*, edited by D. Crisan and B. Rozovsky, 656–704. Oxford University Press.

Escobar, M. D., and M. West. 1995. "Bayesian Density Estimation and Inference Using Mixtures." *Journal of the American Statistical Association* 90 (430): 577–88.

Everitt, R. G., A. M. Johansen, E. Rowing, and M. Evdemon-Hogan. 2017. "Bayesian Model Selection with Un-Normalised Likelihoods." *Statistics and Computing* 27 (2): 403–22.

Fahrmeir, L., and G. Tutz. 2001. *Multivariate Statistical Modelling Based on Generalised Linear Models*. 2nd ed. New York: Springer.

Fearnhead, P., J. Bierkens, M. Pollock, and G. O. Roberts. 2018. "Piecewise Deterministic Markov Processes for Continuous-Time Monte Carlo." *Statistical Science* 33 (3): 386–412.

Gaetan, C., and J.-F. Yao. 2003. "A Multiple-Imputation Metropolis Version of the EM Algorithm." *Biometrika* 90 (3): 643–54.

Gelfand, A. E., and A. F. M. Smith. 1990. "Sampling Based Approaches to Calculating Marginal Densities." *Journal of the American Statistical Association* 85: 398–409.

Gelman, A., G. O. Roberts, and W. R. Gilks. 1995. "Efficient Metropolis Jumping Rules." In *Bayesian Statistics*, edited by J. M. Bernado, J. Berger, A. Dawid, and A. Smith, 5:599–607. Oxford: Oxford University Press.

Gelman, A., and B. D. Rubin. 1992. "Inference from Iterative Simulation Using Multiple Sequences." *Statistical Science* 7: 457–72.

Geman, S., and D. Geman. 1984. "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 6 (6): 721–41.

Gerber, Mathieu, and Nicolas Chopin. 2015. "Sequential Quasi Monte Carlo." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 77 (3): 509–79.

Gerber, M., N. Chopin, and N. Whiteley. 2019. "Negative Association, Ordering and Convergence of Resampling Methods." *Annals of Statistics* 47 (4): 2236–60.

Geweke, J. 1989. "Bayesian Inference in Econometrics Models Using Monte Carlo Integration." *Econometrica* 57 (6): 1317–39.

Girolami, M., and B. Calderhead. 2011. "Riemann Manifold Langevin and Hamiltonian Monte Carlo Methods." *Journal of the Royal Statistical Society B* 73 (2): 123–214.

Goertzel, G. 1949. "Quota Sampling and Importance Functions in Stochastic Solution of Particle Problems." Technical Report ORNL-434. Oak Ridge National Laboratory, Tennessee, USA: Oak Ridge National Laboratory.

Gong, L., and J. M. Flegal. 2016. "A Practical Sequential Stopping Rule for High- Dimensional Markov Chain Monte Carlo." *Journal of Computational and Graphical Statistics* 25 (3): 684–700.

Gordon, N. J., S. J. Salmond, and A. F. M. Smith. 1993. "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation." *IEE Proceedings-F* 140 (2): 107–13.

Grelaud, A., C. P. Robert, J.-M. Marin, F. Rodolphe, and J.-F. Taly. 2009. "ABC Likelihood-Free Methodology for Model Choice in Gibbs Random Fields." *Bayesian Analysis* 4 (2): 317–36.

Guihennec-Jouyaux, C., K. L. Mengersen, and C. P. Robert. 1998. "MCMC Convergence Diagnostics: A 'Reviewww'." 9816. Institut National de la Statistique et des Etudes Economiques.

Hajek, B. 1988. "Cooling Schedules for Optimal Annealing." *Mathematics of Operations Research* 13 (2): 311–29.

Hall, Peter. 1986. "On the Bootstrap and Confidence Intervals." *The Annals of Statistics*, 1431–52.

Halton, J. H. 1970. "A Retrospective and Prospective Survey of the Monte Carlo Method." *SIAM Review* 12 (1): 1–63.

Hastings, W. K. 1970. "Monte Carlo Sampling Methods Using Markov Chains and Their Applications." *Biometrika* 52: 97–109.

Hwang, C.-R. 1980. "Laplace's Method Revisited: Weak Convergence of Probability Measures." *Annals of Probability* 8 (6): 1177–82.

Jacquier, E., M. Johannes, and N. Polson. 2007. "MCMC Maximum Likelihood for Latent State Models." *Journal of Econometrics* 137 (2): 615–40.

Johansen, A. M. 2009. "Markov Chains." In *Encyclopaedia of Computer Science and Engineering*, edited by Benjamin W. Wah, 4:1800–1808. 111 River Street, MS 8-02, Hoboken, NJ 07030-5774: John Wiley; Sons, Inc.

Johansen, A. M., A. Doucet, and M. Davy. 2008. "Particle Methods for Maximum Likelihood Parameter Estimation in Latent Variable Models." *Statistics and Computing* 18 (1): 47–57.

Jones, G. L. 2004. "On the Markov Chain Central Limit Theorem." *Probability Surveys* 1: 299–320.

Kirkpatrick, S., C. D. Gelatt Jr., and M. P. Vecchi. 1983. "Optimization by Simulated Annealing." *Science* 270 (4598): 671–80.

Liu, J. S. 2001. *Monte Carlo Strategies in Scientific Computing.* Springer Series in Statistics. New York: Springer Verlag.

Liu, J. S., W. H. Wong, and A. Kong. 1995. "Covariance Structure and Convergence Rate of the Gibbs Sampler with Various Scans." *Journal of the Royal Statistical Society B* 57 (1): 157–69.

Marin, J.-M., N. Pillai, C. P. Robert, and J. Rousseau. 2014. "Relevant Statistics for Bayesian Model Choice." *Journal of the Royal Statistical Society B.*

Marjoram, P., J. Molitor, V. Plagnol, and S. Tavaré. 2003. "Markov chain Monte Carlo without likelihoods." *Proceedings of the National Academy of Sciences (U.S.A.)* 100 (26): 15324–28.

Matsumoto, M., and T. Nishimura. 1998. "Mersenne Twister: A 623-Dimensionally Equidistributed Uniform Pseudo-Random Number Generator." *ACM Transactions on Modeling and Computer Simulation* 8 (1): 3–30.

Medina-Aguayo, F. J., A. Lee, and G. O. Roberts. 2016. "Stability of Noisy Metropolis-Hastings." *Statistics and Computing* 26 (6): 1187–1211.

Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, and A. H. Teller. 1953. "Equation of State Calculations by Fast Computing Machines." *Journal of Chemical Physics* 21: 1087–92.

Meyn, S. P., and R. L. Tweedie. 1993. *Markov Chains and Stochastic Stability.* Springer Verlag. http://black.csl.uiuc.edu/~meyn/pages/TOC.html.

Morokoff, W. J., and R. E. Caflisch. 1995. "Quasi-Monte Carlo Integration." *J. Comp. Phys.* 122: 218–30.

Neal, R. 2011. "MCMC Using Hamiltonian Dynamics." In *Handbook of Markov Chain Monte Carlo*, edited by S. Brooks, A. Gelman, G. L. Jones, and X.-L. Meng, 113–62. CRC Press.

Neal, R. M. 2001. "Annealed Importance Sampling." *Statistics and Computing* 11: 125–39.

Niederreiter, H. 1992. *Random Number Generation and Quasi-Monte Carlo Methods.* Society for Industrial; Applied Mathematics.

Nummelin, E. 1984. *General Irreducible Markov Chains and Non-Negative Operators.* 1st Paperback. Cambridge Tracts in Mathematics 83. Cambridge University Press.

Owen, A. O. 2017. "Statistically Efficient Thinning of a Markov chain Sampler." *Journal of Computational and Graphical Statistics* 26 (3): 738–44.

Perron, F. 1999. "Beyond Accept-Reject Sampling." *Biometrika* 86 (4): 803–13.

Peters, G. W., Y. Fan, and S. Sisson. 2012. "On Sequential Monte Carlo, Partial Rejection Control and Approximate Bayesian Computation." *Statistics and Computing* 22 (6): 1209–22.

Philippe, A., and C. P. Robert. 2001. "Riemann Sums for MCMC Estimation and Convergence Monitoring." *Statistics and Computing* 11 (2): 103–15.

Pritchard, J. K., M. T. Seielstad, A. Perez-Lezaun, and M. W. Feldman. 1999. "Population growth of human Y chromosomes: a study of Y chromosome microsatellites." *Mol Biol Evol* 16 (12): 1791–98.

R Core Team. 2013. *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing. http://www.R-project.org/.

Robert, C. P., and G. Casella. 2004. *Monte Carlo Statistical Methods.* Second. New York: Springer Verlag.

Robert, C. P., J. M. Cornuet, J. M. Marin, and N. S. Pillai. 2011. "Lack of Confidence in Approximate Bayesian Computational (ABC) Model Choice." *Proceedings of the National Academy of Science, USA* 108 (37): 15112–17.

Roberts, G. 1996. "Markov Chain Concepts Related to Sampling Algorithms." In *Markov Chain Monte Carlo in Practice*, edited by W. R. Gilks, S. Richardson, and D. J. Spieghalter, first, 45–54. Chapman; Hall.

Roberts, G. O., A. Gelman, and W. Gilks. 1997. “Weak Convergence and Optimal Scaling of Random Walk Metropolis Algorithms.” *Annals of Applied Probability* 7 (1): 110–20.

Roberts, G. O., and J. S. Rosenthal. 2004. “General State Space Markov Chains and MCMC Algorithms.” *Probability Surveys* 1: 20–71.

———. 2011. “Quantitative Non-Geometric Convergence Bounds for Independence Samplers.” *Methodology and Computing in Applied Probability* 13: 391–403.

Roberts, G., and R. Tweedie. 1996. “Geometric Convergence and Central Limit Theorems for Multivariate Hastings and Metropolis Algorithms.” *Biometrika* 83: 95–110.

Roeder, K. 1990. “Density Estimation with Cofidence Sets Exemplified by Superclusters and Voids in Galaxies.” *Journal of the American Statistical Association* 85 (411): 617–24.

Roeder, K., and L. Wasserman. 1997. “Practical Bayesian Density Estimation Using Mixtures of Normals.” *Journal of the American Statistical Association* 92 (439): 894–902.

Salmon, J. K., M. A. Moraes, R. O. Dror, and D. E. Shaw. 2011. “Parallel Random Numbers: As Easy as 1, 2, 3.” In *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*.

Sisson, S. A., Y. Fan, and M. M. Tanaka. 2007. “Sequential Monte Carlo Without Likelihoods.” *Proceedings of the National Academy of Science, USA* 104 (4): 1760–65.

Stephens, M. 2007. “Inference Under the Coalescent.” In *Handbook of Statistical Genetics*, edited by D. Balding, M. Bishop, and C. Cannings, 878–908. Wiley, Chichester, UK.

Tanner, M. A., and W. H. Wong. 1987. “The Calculation of Posterior Distributions by Data Augmentation.” *Journal of the American Statistical Association* 82 (398): 528–40.

Tavaré, S., D. J. Balding, R. C. Griffiths, and P. Donnelly. 1997. “Inferring Coalescence Times from DNA Sequence Data.” *Geneetics* 145: 505–18.

Tierney, L. 1994. “Markov Chains for Exploring Posterior Distributions.” *Annals of Statistics* 22: 1701–62.

———. 1996. “Introduction to General State Space Markov Chain Theory.” In *Markov Chain Monte Carlo in Practice*, edited by W. R. Gilks, S. Richardson, and D. J. Spieghalter, first, 59–74. Chapman; Hall.

Voss, J. 2013. *An Introduction to Statistical Computing: A Simulation-Based Approach*. Wiley.

Young, G. A. 1994. “Bootstrap: More Than a Stab in the Dark?” *Statistical Science* 9 (3): 382–95.