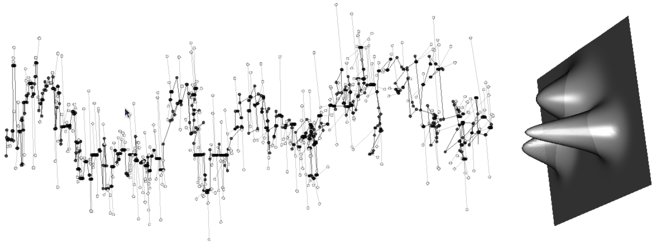
Ioana A. Cosma and Ludger Evers

Markov Chains and Monte Carlo Methods

Lecture Notes

March 12, 2010



African Institute for Mathematical Sciences

AIMS

UNIVERSITY OF CAMBRIDGE | Department of Pure Mathematics and Mathematical Statistics

University of Glasgow | Department of Statistics

These notes are based on notes written originally by Adam M. Johansen and Ludger Evers for a lecture course on Monte Carlo Methods given at the University of Bristol in 2007.

The LaTeX source files of this document can be obtained from the authors.

Table of Contents

Chapter 1

Markov Chains

This chapter introduces Markov chains[1], a special kind of random process which is said to have "no memory": the evolution of the process in the future depends only on the present state and not on where it has been in the past. In order to be able to study Markov chains, we first need to introduce the concept of a stochastic process.

1.1 Stochastic processes

Definition 1.1 (Stochastic process). *A* stochastic process X *is a family* $\{X_t : \; t \in T\}$ *of random variables* $X_t : \Omega \to S$. T *is hereby called the* index set *("time") and* S *is called the* state space.

We will soon focus on stochastic *processes in discrete time*, i.e. we assume that $T \subset \mathbb{N}$ or $T \subset \mathbb{Z}$. Other choices would be $T = [0, \infty)$ or $T = \mathbb{R}$ (*processes in continuous time*) or $T = \mathbb{R} \times \mathbb{R}$ (*spatial process*).

An example of a stochastic process in discrete time would be the sequence of temperatures recorded every morning at Braemar in the Scottish Highlands. Another example would be the price of a share recorded at the opening of the market every day. During the day we can trace the share price continuously, which would constitute a stochastic process in continuous time.

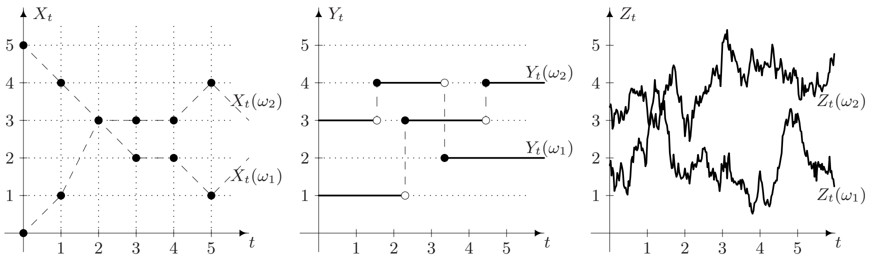
We can distinguish between processes not only based on their index set T, but also based on their state space S, which gives the "range" of possible values the process can take. An important special case arises if the state space S is a countable set. We shall then call X a *discrete process*. The reasons for treating discrete processes separately are the same as for treating discrete random variables separately: we can assume without loss of generality that the state space are the natural numbers. This special case will turn out to be much simpler than the case of a general state space.

Definition 1.2 (Sample Path). *For a given realisation* $\omega \in \Omega$ *the collection* $\{X_t(\omega) : \; t \in T\}$ *is called the* sample path *of* X *at* ω.

If $T = \mathbb{N}_0$ (discrete time) the sample path is a sequence; if $T = \mathbb{R}$ (continuous time) the sample path is a function from $\mathbb{R}$ to S.

Figure 1.1 shows sample paths both of a stochastic process in discrete time (panel (a)), and of two stochastic processes in continuous time (panels (b) and (c)). The process in panel (b) has a discrete state space, whereas the process in panel (c) has the real numbers as its state space ("continuous state space"). Note that whilst certain stochastic processes have sample paths that are (almost surely) continuous or differentiable, this does not need to be the case.

[1] named after the Andrey Andreyevich Markov (1856–1922), a Russian mathematician.



(a) Two sample paths of a discrete process in discrete time.

(b) Two sample paths of a discrete process in continuous time.

(c) Two sample paths of a continuous process in continuous time.

Figure 1.1. Examples of sample paths of stochastic processes.

A stochastic process is not only characterised by the marginal distributions of X_t, but also by the dependency structure of the process. This dependency structure can be expressed by the *finite-dimensional distributions* of the process:

$$\mathbb{P}(X_{t_1} \in A_1, \dots, X_{t_k} \in A_k)$$

where $t_1, \dots, t_k \in T$, $k \in \mathbb{N}$, and $A_1, \dots, A_k$ are measurable subsets of S. In the case of $S \subset \mathbb{R}$ the finite-dimensional distributions can be represented using their joint distribution functions

$$F_{(t_1,\dots,t_k)}(x_1, \dots, x_k) = \mathbb{P}(X_{t_1} \in (-\infty, x_1], \dots, X_{t_k} \in (-\infty, x_k]).$$

This raises the question whether a stochastic process X is fully described by its finite dimensional distributions. The answer to this is given by Kolmogorov's existence theorem. However, in order to be able to formulate the theorem, we need to introduce the concept of a consistent family of finite-dimensional distributions. To keep things simple, we will formulate this condition using distributions functions. We shall call a family of finite dimensional distribution functions *consistent* if for any collection of times $t_1, \dots t_k$, for all $j \in \{1, \dots, k\}$

$$F_{(t_1,\dots,t_{j-1},t_j,t_{j+1},\dots,t_k)}(x_1, \dots, x_{j-1}, +\infty, x_{j+1}, \dots, x_k) = F_{(t_1,\dots,t_{j-1},t_{j+1},\dots,t_k)}(x_1, \dots, x_{j-1}, x_{j+1}, \dots, x_k) \tag{1.1}$$

This consistency condition says nothing else than that lower-dimensional members of the family have to be the marginal distributions of the higher-dimensional members of the family. For a discrete state space, (1.1) corresponds to

$$\sum_{x_j} p_{(t_1,\dots,t_{j-1},t_j,t_{j+1},\dots,t_k)}(x_1, \dots, x_{j-1}, x_j, x_{j+1}, \dots, x_k) = p_{(t_1,\dots,t_{j-1},t_{j+1},\dots,t_k)}(x_1, \dots, x_{j-1}, x_{j+1}, \dots, x_k),$$

where $p_{(\dots)}(\cdot)$ are the joint probability mass functions (p.m.f.). For a continuous state space, (1.1) corresponds to

$$\int f_{(t_1,\dots,t_{j-1},t_j,t_{j+1},\dots,t_k)}(x_1, \dots, x_{j-1}, x_j, x_{j+1}, \dots, x_k) \; dx_j = f_{(t_1,\dots,t_{j-1},t_{j+1},\dots,t_k)}(x_1, \dots, x_{j-1}, x_{j+1}, \dots, x_k)$$

where $f_{(\dots)}(\cdot)$ are the joint probability density functions (p.d.f.).

Without the consistency condition we could obtain different results when computing the same probability using different members of the family.

Theorem 1.3 (Kolmogorov). *Let* $F_{(t_1,\dots,t_k)}$ *be a family of consistent finite-dimensional distribution functions. Then there exists a probability space and a stochastic process* X, *such that*

$$F_{(t_1,\dots,t_k)}(x_1, \dots, x_k) = \mathbb{P}(X_{t_1} \in (-\infty, x_1], \dots, X_{t_k} \in (-\infty, x_k]).$$

Proof. The proof of this theorem can be for example found in (Gihman and Skohorod, 1974). □

Thus we can specify the distribution of a stochastic process by writing down its finite-dimensional distributions. Note that the stochastic process X is not necessarily uniquely determined by its finite-dimensional distributions. However, the finite dimensional distributions uniquely determine all probabilities relating to events involving an at most countable collection of random variables. This is however, at least as far as this course is concerned, all that we are interested in.

In what follows we will only consider the case of a stochastic process in discrete time i.e. $T = \mathbb{N}_0$ (or $\mathbb{Z}$). Initially, we will also assume that the state space is discrete.

1.2 Discrete Markov chains

1.2.1 Introduction

In this section we will define Markov chains, however we will focus on the special case that the state space S is (at most) countable. Thus we can assume without loss of generality that the state space S is the set of natural numbers $\mathbb{N}$ (or a subset of it): there exists a bijection that uniquely maps each element to S to a natural number, thus we can relabel the states $1, 2, 3, \ldots$.

Definition 1.4 (Discrete Markov chain). *Let X be a stochastic process in discrete time with countable ("discrete") state space. X is called a* Markov chain (with discrete state space) *if X satisfies the* Markov property

$$\mathbb{P}(X_{t+1} = x_{t+1} | X_t = x_t, \ldots, X_0 = x_0) = \mathbb{P}(X_{t+1} = x_{t+1} | X_t = x_t)$$

This definition formalises the idea of the process depending on the past only through the present. If we know the current state X_t, then the next state X_{t+1} is independent of the past states $X_0, \ldots X_{t-1}$. Figure 1.2 illustrates this idea.[2]

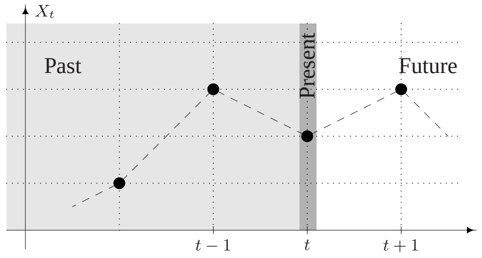


Figure 1.2. Past, present, and future of a Markov chain at t.

Proposition 1.5. *The Markov property is equivalent to assuming that for all $k \in \mathbb{N}$ and all $t_1 < \ldots < t_k \leq t$*

$$\mathbb{P}(X_{t+1} = x_{t+1} | X_{t_k} = x_{t_k}, \ldots, X_{t_1} = x_{t_1}) = \mathbb{P}(X_{t+1} = x_{t+1} | X_{t_k} = x_{t_k}).$$

Proof. (homework) □

Example 1.1 (Phone line). Consider the simple example of a phone line. It can either be busy (we shall call this state 1) or free (which we shall call 0). If we record its state every minute we obtain a stochastic process $\{X_t : t \in \mathbb{N}_0\}$.

[2] A similar concept (*Markov processes*) exists for processes in continuous time. See e.g. `http://en.wikipedia.org/wiki/Markov_process`.

If we assume that $\{X_t : t \in \mathbb{N}_0\}$ is a Markov chain, we assume that probability of a new phone call being ended is independent of how long the phone call has already lasted. Similarly the Markov assumption implies that the probability of a new phone call being made is independent of how long the phone has been out of use before.
The Markov assumption is compatible with assuming that the usage pattern changes over time. We can assume that the phone is more likely to be used during the day and more likely to be free during the night. ◁

Example 1.2 (Random walk on $\mathbb{Z}$). Consider a so-called *random walk* on $\mathbb{Z}$ starting at $X_0 = 0$. At every time, we can either stay in the state or move to the next smaller or next larger number. Suppose that independently of the current state, the probability of staying in the current state is $1-\alpha-\beta$, the probability of moving to the next smaller number is α and that the probability of moving to the next larger number is β, where $\alpha, \beta \geq 0$ with $\alpha + \beta \leq 1$. Figure 1.3 illustrates this idea. To analyse this process in more detail we write X_{t+1} as

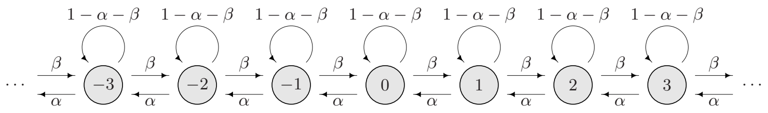


Figure 1.3. Illustration ("Markov graph") of the random walk on $\mathbb{Z}$.

$$X_{t+1} = X_t + E_t,$$

with the E_t being independent and for all t

$$\mathbb{P}(E_t = -1) = \alpha \qquad \mathbb{P}(E_t = 0) = 1 - \alpha - \beta \qquad \mathbb{P}(E_t = 1) = \beta.$$

It is easy to see that

$$\mathbb{P}(X_{t+1} = x_t - 1 | X_t = x_t) = \alpha \quad \mathbb{P}(X_{t+1} = x_t | X_t = x_t) = 1 - \alpha - \beta \quad \mathbb{P}(X_{t+1} = x_t + 1 | X_t = x_t) = \beta$$

Most importantly, these probabilities do not change when we condition additionally on the past $\{X_{t-1} = x_{t-1}, \ldots, X_0 = x_0\}$:

$$\begin{aligned} & \mathbb{P}(X_{t+1} = x_{t+1} | X_t = x_t, X_{t-1} = x_{t-1} \ldots, X_0 = x_0) \\ = \; & \mathbb{P}(E_t = x_{t+1} - x_t | E_{t-1} = x_t - x_{t-1}, \ldots, E_0 = x_1 - x_0, X_0 = x_o) \\ \overset{E_s \perp E_t}{=} \; & \mathbb{P}(E_t = x_{t+1} - x_t) = \mathbb{P}(X_{t+1} = x_{t+1} | X_t = x_t) \end{aligned}$$

Thus $\{X_t : t \in \mathbb{N}_0\}$ is a Markov chain. ◁

The distribution of a Markov chain is fully specified by its *initial distribution* $\mathbb{P}(X_0 = x_0)$ and the *transition probabilities* $\mathbb{P}(X_{t+1} = x_{t+1} | X_t = x_t)$, as the following proposition shows.

Proposition 1.6. *For a discrete Markov chain $\{X_t : t \in \mathbb{N}_0\}$ we have that*

$$\mathbb{P}(X_t = x_t, X_{t-1} = x_{t-1}, \ldots, X_0 = x_0) = \mathbb{P}(X_0 = x_0) \cdot \prod_{\tau=0}^{t-1} \mathbb{P}(X_{\tau+1} = x_{\tau+1} | X_\tau = x_\tau).$$

Proof. From the definition of conditional probabilities we can derive that

$$\begin{aligned}\mathbb{P}(X_t = x_t, X_{t-1} = x_{t-1}, \ldots, X_0 = x_0) = \mathbb{P}(X_0 = x_0)\\ \cdot\, \mathbb{P}(X_1 = x_1 | X_0 = x_0)\\ \cdot\, \underbrace{\mathbb{P}(X_2 = x_2 | X_1 = x_1, X_0 = x_0)}_{=\mathbb{P}(X_2=x_2|X_1=x_1)}\\ \ldots\\ \cdot\, \underbrace{\mathbb{P}(X_t = x_t | X_{t-1} = x_{t-1}, \ldots, X_0 = x_0)}_{=\mathbb{P}(X_t=x_t|X_{t-1}=x_{t-1})}\\ = \prod_{\tau=0}^{t-1} \mathbb{P}(X_{\tau+1} = x_{\tau+1} | X_\tau = x_\tau).\end{aligned}$$ □

Comparing the equation in proposition 1.6 to the first equation of the proof (which holds for any sequence of random variables) illustrates how powerful the Markovian assumption is.

To simplify things even further we will introduce the concept of a *homogeneous Markov chain*, which is a Markov chains whose behaviour does not change over time.

Definition 1.7 (Homogeneous Markov Chain). *A Markov chain* $\{X_t : t \in \mathbb{N}_0\}$ *is said to be* homogeneous *if*

$$\mathbb{P}(X_{t+1} = j | X_t = i) = p_{ij}$$

for all $i, j \in S$*, and independent of* $t \in \mathbb{N}_0$.

In the following we will assume that all Markov chains are homogeneous.

Definition 1.8 (Transition kernel). *The matrix* $\mathbf{K} = (k_{ij})_{ij}$ *with* $k_{ij} = \mathbb{P}(X_{t+1} = j | X_t = i)$ *is called the* transition kernel *(or* transition matrix*) of the homogeneous Markov chain* X.

We will see that together with the initial distribution, which we might write as a vector $\boldsymbol{\lambda}_0 = (\mathbb{P}(X_0 = i))_{(i \in S)}$, the transition kernel $\mathbf{K}$ fully specifies the distribution of a homogeneous Markov chain.

However, we start by stating two basic properties of the transition kernel $\mathbf{K}$:

- The entries of the transition kernel are non-negative (they are probabilities).
- Each row of the transition kernel sums to 1, as

$$\sum_j k_{ij} = \sum_j \mathbb{P}(X_{t+1} = j | X_t = i) = \mathbb{P}(X_{t+1} \in S | X_t = i) = 1$$

Example 1.3 (Phone line (continued)). Suppose that in the example of the phone line the probability that someone makes a new call (if the phone is currently unused) is 10% and the probability that someone terminates an active phone call is 30%. If we denote the states by 0 (phone not in use) and 1 (phone in use). Then

$$\begin{aligned}\mathbb{P}(X_{t+1} = 0 | X_t = 0) &= 0.9 &\qquad \mathbb{P}(X_{t+1} = 1 | X_t = 0) &= 0.1\\ \mathbb{P}(X_{t+1} = 0 | X_t = 1) &= 0.3 &\qquad \mathbb{P}(X_{t+1} = 1 | X_t = 1) &= 0.7,\end{aligned}$$

and the transition kernel is

$$\mathbf{K} = \begin{pmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{pmatrix}.$$

The transition probabilities are often illustrated using a so-called Markov graph. The Markov graph for this example is shown in figure 1.4. Note that knowing $\mathbf{K}$ alone is not enough to find the distribution of the states: for this we also need to know the initial distribution $\boldsymbol{\lambda}_0$. ◁

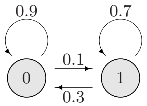


Figure 1.4. Markov graph for the phone line example.

Example 1.4 (Random walk on $\mathbb{Z}$ (continued)). The transition kernel for the random walk on $\mathbb{Z}$ is a Toeplitz matrix with an infinite number of rows and columns:

$$\mathbf{K} = \begin{pmatrix} \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ \ddots & \alpha & 1-\alpha-\beta & \beta & 0 & 0 & 0 & \ddots \\ \ddots & 0 & \alpha & 1-\alpha-\beta & \beta & 0 & 0 & \ddots \\ \ddots & 0 & 0 & \alpha & 1-\alpha-\beta & \beta & 0 & \ddots \\ \ddots & 0 & 0 & 0 & \alpha & 1-\alpha-\beta & \beta & \ddots \\ \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \end{pmatrix}$$

The Markov graph for this Markov chain was given in figure 1.3. ◁

We will now generalise the concept of the transition kernel, which contains the probabilities of moving from state i to step j in one step, to the m-step transition kernel, which contains the probabilities of moving from state i to step j in m steps:

Definition 1.9 (m-step transition kernel). *The matrix* $\mathbf{K}^{(m)} = (k_{ij}^{(m)})_{ij}$ *with* $k_{ij}^{(m)} = \mathbb{P}(X_{t+m} = j | X_t = i)$ *is called the* m-step transition kernel *of the homogeneous Markov chain* X.

We will now show that the m-step transition kernel is nothing other than the m-power of the transition kernel.

Proposition 1.10. *Let* X *be a homogeneous Markov chain, then*

i. $\mathbf{K}^{(m)} = \mathbf{K}^m$*, and*
ii. $\mathbb{P}(X_m = j) = (\boldsymbol{\lambda}_0' \mathbf{K}^{(m)})_j$.

Proof. i. We will first show that for $m_1, m_2 \in \mathbb{N}$ we have that $\mathbf{K}^{(m_1+m_2)} = \mathbf{K}^{(m_1)} \cdot \mathbf{K}^{(m_2)}$:

$$\begin{aligned}\mathbb{P}(X_{t+m_1+m_2} = k | X_t = i) &= \sum_j \mathbb{P}(X_{t+m_1+m_2} = k, X_{t+m_1} = j | X_t = i)\\ &= \sum_j \underbrace{\mathbb{P}(X_{t+m_1+m_2} = k | X_{t+m_1} = j, X_t = i)}_{=\mathbb{P}(X_{t+m_1+m_2}=k|X_{t+m_1}=j)=\mathbb{P}(X_{t+m_2}=k|X_t=j)} \mathbb{P}(X_{t+m_1} = j | X_t = i)\\ &= \sum_j \mathbb{P}(X_{t+m_2} = k | X_t = j)\mathbb{P}(X_{t+m_1} = j | X_t = i)\\ &= \sum_j \mathbf{K}_{ij}^{(m_1)} \mathbf{K}_{jk}^{(m_2)} = \left(\mathbf{K}^{(m_1)} \mathbf{K}^{(m_2)}\right)_{i,k}\end{aligned}$$

Thus $\mathbf{K}^{(2)} = \mathbf{K} \cdot \mathbf{K} = \mathbf{K}^2$, and by induction $\mathbf{K}^{(m)} = \mathbf{K}^m$.

ii. $\mathbb{P}(X_m = j) = \sum_i \mathbb{P}(X_m = j, X_0 = i) = \sum_i \underbrace{\mathbb{P}(X_m = j | X_0 = i)}_{=\mathbf{K}_{ij}^{(m)}} \underbrace{\mathbb{P}(X_0 = i)}_{=(\boldsymbol{\lambda}_0)_i} = (\boldsymbol{\lambda}_0' \mathbf{K}^m)_j$ □

Example 1.5 (Phone line (continued)). In the phone-line example, the transition kernel is

$$\mathbf{K} = \begin{pmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{pmatrix}.$$

The m-step transition kernel is

$$\mathbf{K}^{(m)} = \mathbf{K}^m = \begin{pmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{pmatrix}^m = \begin{pmatrix} \frac{3+\left(\frac{3}{5}\right)^m}{4} & \frac{1-\left(\frac{3}{5}\right)^m}{4} \\ \frac{1+\left(\frac{3}{5}\right)^m}{4} & \frac{3-\left(\frac{3}{5}\right)^m}{4} \end{pmatrix}.$$

Thus the probability that the phone is free given that it was free 10 hours ago is $\mathbb{P}(X_{t+10} = 0|X_t = 0) = K_{0,0}^{(10)} = \frac{3+\left(\frac{3}{5}\right)^{10}}{4} = \frac{7338981}{9765625} = 0.7515$. ◁

1.2.2 Classification of states

Definition 1.11 (Classification of states). *(a) A state i is said to* lead to *a state j ("$i \rightsquigarrow j$") if there is an $m \geq 0$ such that there is a positive probability of getting from state i to state j in m steps, i.e.*

$$k_{ij}^{(m)} = \mathbb{P}(X_{t+m} = j|X_t = i) > 0.$$

(b) Two states i and j are said to communicate *("$i \sim j$") if $i \rightsquigarrow j$ and $j \rightsquigarrow i$.*

From the definition we can derive for states $i, j, k \in S$:

- $i \rightsquigarrow i$ (as $k_{ii}^{(0)} = \mathbb{P}(X_{t+0} = i|X_t = i) = 1 > 0$), thus $i \sim i$.
- If $i \sim j$, then also $j \sim i$.
- If $i \rightsquigarrow j$ and $j \rightsquigarrow k$, then there exist $m_{ij}, m_2 \geq 0$ such that the probability of getting from state i to state j in m_{ij} steps is positive, i.e. $k_{ij}^{(m_{ij})} = \mathbb{P}(X_{t+m_{ij}} = j|X_t = i) > 0$, as well as the probability of getting from state j to state k in m_{jk} steps, i.e. $k_{jk}^{(m_{jk})} = \mathbb{P}(X_{t+m_{jk}} = k|X_t = j) = \mathbb{P}(X_{t+m_{ij}+m_{jk}} = k|X_{t+m_{ij}} = j) > 0$. Thus we can get (with positive probability) from state i to state k in $m_{ij} + m_{jk}$ steps:

$$\begin{aligned} k_{ik}^{(m_{ij}+m_{jk})} &= \mathbb{P}(X_{t+m_{ij}+m_{jk}} = k|X_t = i) = \sum_{\iota} \mathbb{P}(X_{t+m_{ij}+m_{jk}} = k|X_{t+m_{ij}} = \iota)\mathbb{P}(X_{t+m_{ij}} = \iota|X_t = i) \\ &\geq \underbrace{\mathbb{P}(X_{t+m_{ij}+m_{jk}} = k|X_{t+m_{ij}} = j)}_{>0} \underbrace{\mathbb{P}(X_{t+m_{ij}} = j|X_t = i)}_{>0} > 0 \end{aligned}$$

  Thus $i \rightsquigarrow j$ and $j \rightsquigarrow k$ imply $i \rightsquigarrow k$. Thus $i \sim j$ and $j \sim k$ also imply $i \sim k$

Thus $\sim$ is an equivalence relation and we can partition the state space S into *communicating classes*, such that all states in one class communicate and no larger classes can be formed. A class C is called *closed* if there are no paths going out of C, i.e. for all $i \in C$ we have that $i \rightsquigarrow j$ implies that $j \in C$.

We will see that states within one class have many properties in common.

Example 1.6. Consider a Markov chain with transition kernel

$$\mathbf{K} = \begin{pmatrix} \frac{1}{2} & \frac{1}{4} & 0 & \frac{1}{4} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{3}{4} & 0 & 0 & \frac{1}{4} \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & \frac{3}{4} & 0 & 0 & 0 & \frac{1}{4} \\ 0 & 0 & \frac{1}{2} & 0 & 0 & \frac{1}{2} \end{pmatrix}$$

The Markov graph is shown in figure 1.5. We have that $2 \sim 4$, $2 \sim 5$, $3 \sim 6$, $4 \sim 5$. Thus the communicating classes are $\{1\}$, $\{2, 4, 5\}$, and $\{3, 6\}$. Only the class $\{3, 6\}$ is closed. ◁

Finally, we will introduce the notion of an *irreducible chain*. This concept will become important when we analyse the limiting behaviour of the Markov chain.

Definition 1.12 (Irreducibility). *A Markov chain is called* irreducible *if it only consists of a single class, i.e. all states communicate.*

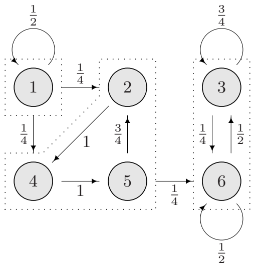


Figure 1.5. Markov graph of the chain of example 1.6. The communicating classes are $\{1\}$, $\{2, 4, 5\}$, and $\{3, 6\}$.

The Markov chain considered in the phone-line example (examples 1.1,1.3, and 1.5) and the random walk on $\mathbb{Z}$ (examples 1.2 and 1.4) are irreducible chains. The chain of example 1.6 is not irreducible.

In example 1.6 the states 2, 4 and 5 can only be visited in this order: if we are currently in state 2 (i.e. $X_t = 2$), then we can only visit this state again at time $t + 3, t + 6, \ldots$. Such a behaviour is referred to as periodicity.

Definition 1.13 (Period). *(a) A state $i \in S$ is said to have* period

$$d(i) = \gcd\{m \geq 1 : \ K_{ii}^{(m)} > 0\},$$

where $\gcd$ *denotes the greatest common denominator.*

(b) If $d(i) = 1$ the state i is called aperiodic.

(c) If $d(i) > 1$ the state i is called periodic.

For a periodic state i, the number of steps required to possibly get back to this state must be a multiple of the period $d(i)$.

To analyse the periodicity of a state i we must check the existence of paths of positive probability and of length m going from the state i back to i. If no path of length m exists, then $K_{ii}^{(m)} = 0$. If there exists a single path of positive probability of length m, then $K_{ii}^{(m)} > 0$.

Example 1.7 (Example 1.6 continued). In example 1.6 the state 2 has period $d(2) = 3$, as all paths from 2 back to 2 have a length which is a multiple of 3, thus

$$K_{22}^{(3)} > 0, \qquad K_{22}^{(6)} > 0, \qquad K_{22}^{(9)} > 0, \qquad \ldots$$

All other $K_{22}^{(m)} = 0$ ($\frac{m}{3} \notin \mathbb{N}_0$), thus the period is $d(2) = 3$ (3 being the greatest common denominator of $3, 6, 9, \ldots$). Similarly $d(4) = 3$ and $d(5) = 3$.

The states 3 and 6 are aperiodic, as there is a positive probability of remaining in these states, thus $K_{33}^{(m)} > 0$ and $K_{66}^{(m)} > 0$ for all m, thus $d(3) = d(6) = 1$. ◁

In example 1.6 all states within one communicating class had the same period. This holds in general, as the following proposition shows:

Proposition 1.14. *(a) All states within a communicating class have the same period.*

(b) In an irreducible chain all states have the same period.

Proof. (a) Suppose $i \sim j$. Thus there are paths of positive probability between these two states. Suppose we can get from i to j in m_{ij} steps and from j to i in m_{ji} steps. Suppose also that we can get from j back to j in m_{jj} steps. Then we can get from i back to i in $m_{ij} + m_{ji}$ steps as well as in $m_{ij} + m_{jj} + m_{ji}$ steps. Thus $m_{ij} + m_{ji}$ and $m_{ij} + m_{jj} + m_{ji}$ must be divisible by the period $d(i)$ of state i. Thus m_{jj} is also divisible by $d(i)$ (being the difference of two numbers divisible by $d(i)$).

The above argument holds for any path between j and j, thus the length of any path from j back to j is divisible by $d(i)$. Thus $d(i) \leq d(j)$ ($d(j)$ being the greatest common denominator).

Repeating the same argument with the rôles of i and j swapped gives us $d(j) \leq d(i)$, thus $d(i) = d(j)$.

(b) An irreducible chain consists of a single communicating class, thus (b) is implied by (a). □

1.2.3 Recurrence and transience

If we follow the Markov chain of example 1.6 long enough, we will eventually end up switching between state 3 and 6 without ever coming back to the other states Whilst the states 3 and 6 will be visited infinitely often, the other states will eventually be left forever.

In order to formalise these notions we will introduce the *number of visits* in state i:

$$V_i = \sum_{t=0}^{+\infty} 1_{\{X_t=i\}}$$

The expected number of visits in state i given that we start the chain in i is

$$\mathbb{E}(V_i|X_0=i) = \mathbb{E}\left(\sum_{t=0}^{+\infty} 1_{\{X_t=i\}} \Big| X_0=i\right) = \sum_{t=0}^{+\infty} \mathbb{E}(1_{\{X_t=i\}}|X_0=i) = \sum_{t=0}^{+\infty} \mathbb{P}(X_t=i|X_o=i) = \sum_{t=0}^{+\infty} k_{ii}^{(t)}$$

Based on whether the expected number of visits in a state is infinite or not, we will classify states as recurrent or transient:

Definition 1.15 (Recurrence and transience). *(a) A state i is called* recurrent *if* $\mathbb{E}(V_i|X_0=i) = +\infty$.
(b) A state i is called transient *if* $\mathbb{E}(V_i|X_0=i) < +\infty$.

One can show that a recurrent state will (almost surely) be visited infinitely often, whereas a transient state will (almost surely) be visited only a finite number of times.

In proposition 1.14 we have seen that within a communicating class either all states are aperiodic, or all states are periodic. A similar dichotomy holds for recurrence and transience.

Proposition 1.16. *Within a communicating class, either all states are transient or all states are recurrent.*

Proof. Suppose $i \sim j$. Then there exists a path of length m_{ij} leading from i to j and a path of length m_{ji} from j back to i, i.e. $k_{ij}^{(m_{ij})} > 0$ and $k_{ji}^{(m_{ji})} > 0$.

Suppose furthermore that the state i is transient, i.e. $\mathbb{E}(V_i|X_0=i) = \sum_{t=0}^{+\infty} k_{ii}^{(t)} < +\infty$.

This implies

$$\mathbb{E}(V_j|X_0=j) = \sum_{t=0}^{+\infty} k_{jj}^{(t)} = \frac{1}{k_{ij}^{(m_{ij})} k_{ji}^{(m_{ji})}} \sum_{t=0}^{+\infty} \underbrace{k_{ij}^{(m_{ij})} k_{jj}^{(t)} k_{ji}^{(m_{ji})}}_{\leq k_{ii}^{(m+t+n)}} \leq \frac{1}{k_{ij}^{(m_{ij})} k_{ji}^{(m_{ji})}} \sum_{t=0}^{+\infty} k^{(m_{ij}+t+m_{ji})}$$

$$\leq \frac{1}{k_{ij}^{(m_{ij})} k_{ji}^{(m_{ji})}} \sum_{s=0}^{+\infty} k_{ii}^{(s)} < +\infty,$$

thus state j is be transient as well. □

Finally we state without proof two simple criteria for determining recurrence and transience.

Proposition 1.17. *(a) Every class which is not closed is transient.*
(b) Every finite closed class is recurrent.

Proof. For a proof see (Norris, 1997, sect. 1.5). □

Example 1.8 (Examples 1.6 and 1.7 continued). The chain of example 1.6 had three classes: $\{1\}$, $\{2,4,5\}$, and $\{3,6\}$. The classes $\{1\}$ and $\{2,4,5\}$ are not closed, so they are transient. The class $\{3,6\}$ is closed and finite, thus recurrent. ◁

Note that an infinite closed class is not necessarily recurrent. The random walk on $\mathbb{Z}$ studied in examples 1.2 and 1.4 is only recurrent if it is symmetric, i.e. $\alpha = \beta$, otherwise it drifts off to $-\infty$ or $+\infty$. An interesting result is that a symmetric random walk on $\mathbb{Z}^p$ is only recurrent if $p \leq 2$ (see e.g. Norris, 1997, sect. 1.6).

1.2.4 Invariant distribution and equilibrium

In this section we will study the long-term behaviour of Markov chains. A key concept for this is the invariant distribution.

Definition 1.18 (Invariant distribution). *Let $\boldsymbol{\mu} = (\mu_i)_{i \in S}$ be a probability distribution on the state space S, and let X be a Markov chain with transition kernel $\mathbf{K}$. Then $\boldsymbol{\mu}$ is called the* invariant distribution *(or* stationary distribution*) of the Markov chain X if*[3]

$$\boldsymbol{\mu}'\mathbf{K} = \boldsymbol{\mu}'.$$

If $\boldsymbol{\mu}$ is the stationary distribution of a chain with transition kernel $\mathbf{K}$, then

$$\boldsymbol{\mu}' = \underbrace{\boldsymbol{\mu}'}_{=\boldsymbol{\mu}'\mathbf{K}}\mathbf{K} = \boldsymbol{\mu}'\mathbf{K}^2 = \ldots = \boldsymbol{\mu}'\mathbf{K}^m = \boldsymbol{\mu}'\mathbf{K}^{(m)}$$

for all $m \in \mathbb{N}$. Thus if X_0 in drawn from $\boldsymbol{\mu}$, then all X_m have distribution $\boldsymbol{\mu}$: according to proposition 1.10

$$\mathbb{P}(X_m = j) = (\boldsymbol{\mu}'\mathbf{K}^{(m)})_j = (\boldsymbol{\mu})_j$$

for all m. Thus, if the chain has $\boldsymbol{\mu}$ as initial distribution, the distribution of X will not change over time.

Example 1.9 (Phone line (continued)). In example 1.1,1.3, and 1.5 we studied a Markov chain with the two states 0 ("free") and 1 ("in use") and which modeled whether a phone is free or not. Its transition kernel was

$$\mathbf{K} = \begin{pmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{pmatrix}.$$

To find the invariant distribution, we need to solve $\boldsymbol{\mu}'\mathbf{K} = \boldsymbol{\mu}'$ for $\boldsymbol{\mu}$, which is equivalent to solving the following system of linear equations:

$$(\mathbf{K}' - \mathbf{I})\boldsymbol{\mu} = \mathbf{0}, \qquad \text{i.e.} \qquad \begin{pmatrix} -0.1 & 0.3 \\ 0.1 & -0.3 \end{pmatrix} \cdot \begin{pmatrix} \mu_0 \\ \mu_1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

It is easy to see that the corresponding system is under-determined and that $-\mu_0 + 3\mu_1 = 0$, i.e. $\boldsymbol{\mu} = (\mu_0, \mu_1)' \propto (3,1)$, i.e. $\boldsymbol{\mu} = \left(\frac{3}{4}, \frac{1}{4}\right)'$ (as $\boldsymbol{\mu}$ has to be a probability distribution, thus $\mu_0 + \mu_1 = 1$). ◁

Not every Markov chain has an invariant distribution. The random walk on $\mathbb{Z}$ (studied in examples 1.2 and 1.4) for example does not have an invariant distribution, as the following example shows:

Example 1.10 (Random walk on $\mathbb{Z}$ (continued)). The random walk on $\mathbb{Z}$ had the transition kernel (see example 1.4)

$$\mathbf{K} = \begin{pmatrix} \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ \ddots & \alpha & 1-\alpha-\beta & \beta & 0 & 0 & 0 & \ddots \\ \ddots & 0 & \alpha & 1-\alpha-\beta & \beta & 0 & 0 & \ddots \\ \ddots & 0 & 0 & \alpha & 1-\alpha-\beta & \beta & 0 & \ddots \\ \ddots & 0 & 0 & 0 & \alpha & 1-\alpha-\beta & \beta & \ddots \\ \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \end{pmatrix}$$

[3] i.e. $\boldsymbol{\mu}$ is the left eigenvector of $\mathbf{K}$ corresponding to the eigenvalue 1.

As $\alpha+(1-\alpha-\beta)+\beta=1$ we have for $\boldsymbol{\mu}=(\dots,1,1,1,\dots)'$ that $\boldsymbol{\mu}'\mathbf{K}=\boldsymbol{\mu}'$, however $\boldsymbol{\mu}$ cannot be renormalised to become a probability distribution. ◁

We will now show that if a Markov chain is irreducible and aperiodic, its distribution will in the long run tend to the invariant distribution.

Theorem 1.19 (Convergence to equilibrium). *Let X be an irreducible and aperiodic Markov chain with invariant distribution $\boldsymbol{\mu}$. Then*

$$\mathbb{P}(X_t=i)\xrightarrow{t\to+\infty}\mu_i$$

for all states i.

Outline of the proof. We will explain the outline of the proof using the idea of coupling.

Suppose that X has initial distribution $\boldsymbol{\lambda}$ and transition kernel $\mathbf{K}$. Define a new Markov chain Y with initial distribution $\boldsymbol{\mu}$ and same transition kernel $\mathbf{K}$. Let T be the first time the two chains “meet” in the state i, i.e.

$$T=\min\{t\geq 0:\ X_t=Y_t=i\}$$

Then one can show that $\mathbb{P}(T<\infty)=1$ and define a new process Z by

$$Z_t=\begin{cases}X_t & \text{if } t\leq T\\ Y_t & \text{if } t>T\end{cases}$$

Figure 1.6 illustrates this new chain Z. One can show that Z is a Markov chain with initial distribution $\boldsymbol{\lambda}$ (as $X_0=Z_0$) and transition kernel $\mathbf{K}$ (as both X and Y have the transition kernel $\mathbf{K}$). Thus X and Z have the same distribution and for all $t\in\mathbb{N}_0$ we have that $\mathbb{P}(X_t=j)=\mathbb{P}(Z_t=j)$ for all states $j\in S$.

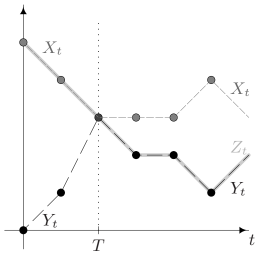


Figure 1.6. Illustration of the chains X (– – –), Y (— —) and Z (thick line) used in the proof of theorem 1.19.

The chain Y has its invariant distribution as initial distribution, thus $\mathbb{P}(Y_t=j)=\mu_j$ for all $t\in\mathbb{N}_0$ and $j\in S$.
As $t\to+\infty$ the probability of $\{Y_t=Z_t\}$ tends to 1, thus

$$\mathbb{P}(X_t=j)=\mathbb{P}(Z_t=j)\to\mathbb{P}(Y_t=j)=\mu_j.$$

A more detailed proof of this theorem can be found in (Norris, 1997, sec. 1.8).

Example 1.11 (Phone line (continued)). We have shown in example 1.9 that the invariant distribution of the Markov chain modeling the phone line is $\boldsymbol{\mu}=\left(\frac{3}{4},\frac{1}{4}\right)$, thus according to theorem 1.19 $\mathbb{P}(X_t=0)\to\frac{3}{4}$ and $\mathbb{P}(X_t=1)\to\frac{1}{4}$. Thus, in the long run, the phone will be free 75% of the time. ◁

Example 1.12. This example illustrates that the aperiodicity condition in theorem 1.19 is necessary.
Consider a Markov chain X with two states $S=\{1,2\}$ and transition kernel

$$\mathbf{K}=\begin{pmatrix}0&1\\1&0\end{pmatrix}.$$

This Markov chains switches deterministically, thus goes either $1,\ 0,\ 1,\ 0,\ \dots$ or $0,\ 1,\ 0,\ 1,\ \dots$. Thus it is periodic with period 2.
Its invariant distribution is $\boldsymbol{\mu}'=\left(\frac{1}{2},\frac{1}{2}\right)$, as

$$\boldsymbol{\mu}'\mathbf{K}=\left(\frac{1}{2},\frac{1}{2}\right)\begin{pmatrix}0&1\\1&0\end{pmatrix}=\left(\frac{1}{2},\frac{1}{2}\right)=\boldsymbol{\mu}'.$$

However if the chain is started in $X_0=1$, i.e. $\boldsymbol{\lambda}=(1,0)$, then

$$\mathbb{P}(X_t=0)=\begin{cases}1 & \text{if } t \text{ is odd}\\ 0 & \text{if } t \text{ is even}\end{cases},\qquad \mathbb{P}(X_t=1)=\begin{cases}0 & \text{if } t \text{ is odd}\\ 1 & \text{if } t \text{ is even}\end{cases},$$

which is different from the invariant distribution, under which all these probabilities would be $\frac{1}{2}$. ◁

1.2.5 Reversibility and detailed balance

In our study of Markov chains we have so far focused on conditioning on the past. For example, we have defined the transition kernel to consist of $k_{ij}=\mathbb{P}(X_{t+1}=j|X_t=i)$. What happens if we analyse the distribution of X_t conditional on the future, i.e we turn the universal clock backwards?

$$\mathbb{P}(X_t=j|X_{t+1}=i)=\frac{\mathbb{P}(X_t=j,X_{t+1}=i)}{\mathbb{P}(X_{t+1}=i)}=\mathbb{P}(X_{t+1}=i|X_t=j)\cdot\frac{\mathbb{P}(X_t=j)}{\mathbb{P}(X_{t+1}=i)}$$

This suggests defining a new Markov chain which goes back in time. As the defining property of a Markov chain was that the past and future are conditionally independent given the present, the same should hold for the “backward chain”, just with the rôles of past and future swapped.

Definition 1.20 (Time-reversed chain). *For $\tau\in\mathbb{N}$ let $\{X_t:\ t=0,\dots,\tau\}$ be a Markov chain. Then $\{Y_t:\ t=0,\dots,\tau\}$ defined by $Y_t=X_{\tau-t}$ is called the* time-reversed chain *corresponding to X.*

We have that

$$\begin{aligned}\mathbb{P}(Y_t=j|Y_{t-1}=i)&=\mathbb{P}(X_{\tau-t}=j|X_{\tau-t+1}=i)=\mathbb{P}(X_s=j|X_{s+1}=i)=\frac{\mathbb{P}(X_s=j,X_{s+1}=i)}{\mathbb{P}(X_{s+1}=i)}\\&=\mathbb{P}(X_{s+1}=i|X_s=j)\cdot\frac{\mathbb{P}(X_s=j)}{\mathbb{P}(X_{s+1}=i)}=k_{ji}\cdot\frac{\mathbb{P}(X_s=j)}{\mathbb{P}(X_{s+1}=i)},\end{aligned}$$

thus the time-reversed chain is in general not homogeneous, even if the forward chain X is homogeneous.

This changes however if the forward chain X is initialised according to its invariant distribution $\boldsymbol{\mu}$. In this case $\mathbb{P}(X_{s+1}=i)=\mu_i$ and $\mathbb{P}(X_s=j)=\mu_j$ for all s, and thus Y is a homogeneous Markov chain with transition probabilities

$$\mathbb{P}(Y_t=j|Y_{t-1}=i)=k_{ji}\cdot\frac{\mu_j}{\mu_i}. \tag{1.2}$$

In general, the transition probabilities for the time-reversed chain will thus be different from the forward chain.

Example 1.13 (Phone line (continued)). In the example of the phone line (examples 1.1, 1.3, 1.5, 1.9, and 1.11) the transition matrix was

$$\mathbf{K}=\begin{pmatrix}0.9&0.1\\0.3&0.7\end{pmatrix}.$$

The invariant distribution was $\boldsymbol{\mu}=\left(\frac{3}{4},\frac{1}{4}\right)'$.
If we use the invariant distribution $\boldsymbol{\mu}$ as initial distribution for X_0, then using (1.2)

$$
\begin{aligned}
\mathbb{P}(Y_t = 0|Y_{t-1} = 0) &= k_{00} \cdot \frac{\mu_0}{\mu_0} = k_{00} = \mathbb{P}(X_t = 0|X_{t-1} = 1)\\
\mathbb{P}(Y_t = 0|Y_{t-1} = 1) &= k_{01} \cdot \frac{\mu_0}{\mu_1} = 0.1 \cdot \frac{\frac{3}{4}}{\frac{1}{4}} = 0.3 = k_{10} = \mathbb{P}(X_t = 0|X_{t-1} = 1)\\
\mathbb{P}(Y_t = 1|Y_{t-1} = 0) &= k_{10} \cdot \frac{\mu_1}{\mu_0} = 0.3 \cdot \frac{\frac{1}{4}}{\frac{3}{4}} = 0.1 = k_{01} = \mathbb{P}(X_t = 1|X_{t-1} = 0)\\
\mathbb{P}(Y_t = 1|Y_{t-1} = 1) &= k_{11} \cdot \frac{\mu_1}{\mu_1} = k_{11} = \mathbb{P}(X_t = 1|X_{t-1} = 1)
\end{aligned}
$$

Thus in this case both the forward chain X and the time-reversed chain Y have the same transition probabilities. We will call such chains *time-reversible*, as their dynamics do not change when time is reversed. ◁

We will now introduce a criterion for checking whether a chain is time-reversible.

Definition 1.21 (Detailed balance). *A transition kernel* $\mathbf{K}$ *is said to be in* detailed balance *with a distribution* $\boldsymbol{\mu}$ *if for all* $i, j \in S$

$$\mu_i k_{ij} = \mu_j k_{ji}.$$

It is easy to see that Markov chain studied in the phone line example (see example 1.13) satisfies the detailed-balance condition.

The detailed-balance condition is a very important concept that we will require when studying Markov Chain Monte Carlo (MCMC) algorithms later. The reason for its relevance is the following theorem, which says that if a Markov chain is in detailed balance with a distribution $\boldsymbol{\mu}$, then the chain is time-reversible, and, more importantly, $\boldsymbol{\mu}$ is the invariant distribution. The advantage of the detailed-balance condition over the condition of definition 1.18 is that the detailed-balance condition is often simpler to check, as it does not involve a sum (or a vector-matrix product).

Theorem 1.22. *Let* X *be a Markov chain with transition kernel* $\mathbf{K}$ *which is in detailed balance with some distribution* $\boldsymbol{\mu}$ *on the states of the chain. Then*

i. $\boldsymbol{\mu}$ *is the invariant distribution of* X.

ii. *If initialised according to* $\boldsymbol{\mu}$, X *is time-reversible, i.e. both* X *and its time reversal have the same transition kernel.*

Proof. i. We have that

$$(\boldsymbol{\mu}'\mathbf{K})_i = \sum_j \underbrace{\mu_j k_{ji}}_{=\mu_i k_{ij}} = \mu_i \underbrace{\sum_j k_{ij}}_{=1} = \mu_i,$$

thus $\boldsymbol{\mu}'\mathbf{K} = \boldsymbol{\mu}'$, i.e. $\boldsymbol{\mu}$ is the invariant distribution.

ii. Let Y be the time-reversal of X, then using (1.2)

$$\mathbb{P}(Y_t = j|Y_{t-1} = i) = \frac{\overbrace{\mu_j k_{ji}}^{\mu_i k_{ij}}}{\mu_i} = k_{ij} = \mathbb{P}(X_t = j|X_{t-1} = i),$$

thus X and Y have the same transition probabilities. □

Note that not every chain which has an invariant distribution is time-reversible, as the following example shows:

Example 1.14. Consider the following Markov chain on $S = \{1, 2, 3\}$ with transition matrix

$$\mathbf{K} = \begin{pmatrix} 0 & 0.8 & 0.2 \\ 0.2 & 0 & 0.8 \\ 0.8 & 0.2 & 0 \end{pmatrix}$$

The corresponding Markov graph is shown in figure 1.7: The stationary distribution of the chain is $\boldsymbol{\mu} = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$.

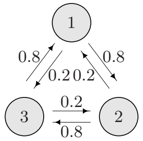


Figure 1.7. Markov graph for the Markov chain of example 1.14.

However the distribution is not time-reversible. Using equation (1.2) we can find the transition matrix of the time-reversed chain Y, which is

$$\begin{pmatrix} 0 & 0.2 & 0.8 \\ 0.8 & 0 & 0.2 \\ 0.2 & 0.8 & 0 \end{pmatrix},$$

which is equal to $\mathbf{K}'$, rather than $\mathbf{K}$. Thus the chains X and its time reversal Y have different transition kernels. When going forward in time, the chain is much more likely to go clockwise in figure 1.7; when going backwards in time however, the chain is much more likely to go counter-clockwise. ◁

1.3 General state space Markov chains

So far, we have restricted our attention to Markov chains with a discrete (i.e. at most countable) state space S. The main reason for this was that this kind of Markov chain is much easier to analyse than Markov chains having a more general state space.

However, most applications of Markov Chain Monte Carlo algorithms are concerned with continuous random variables, i.e. the corresponding Markov chain has a continuous state space S, thus the theory studied in the preceding section does not directly apply. Largely, we defined most concepts for discrete state spaces by looking at events of the type $\{X_t = j\}$, which is only meaningful if the state space is discrete.

In this section we will give a brief overview of the theory underlying Markov chains with general state spaces. Although the basic principles are not entirely different from the ones we have derived in the discrete case, the study of general state space Markov chains involves many more technicalities and subtleties, so that we will not present any proofs here. The interested reader can find a more rigorous treatment in (Meyn and Tweedie, 1993), (Nummelin, 1984), or (Robert and Casella, 2004, chapter 6).

Though this section is concerned with general state spaces we will notationally assume that the state space is $S = \mathbb{R}^d$.

First of all, we need to generalise our definition of a Markov chain (definition 1.4). We defined a Markov chain to be a stochastic process in which, conditionally on the present, the past and the future are independent. In the discrete case we formalised this idea using the conditional probability of $\{X_t = j\}$ given different collections of past events.

In a general state space it can be that all events of the type $\{X_t = j\}$ have probability 0, as it is the case for a process with a continuous state space. A process with a continuous state space spreads the probability so thinly that the probability of exactly hitting one given state is 0 for all states. Thus we have to work with conditional probabilities of sets of states, rather than individual states.

Definition 1.23 (Markov chain). *Let* X *be a stochastic process in discrete time with general state space* S. X *is called a* Markov chain *if* X *satisfies the* Markov property

$$\mathbb{P}(X_{t+1} \in A|X_0 = x_0, \ldots, X_t = x_t) = \mathbb{P}(X_{t+1} \in A|X_t = x_t)$$

for all measurable sets $A \subset S$.

If S is at most countable, this definition is equivalent to definition 1.4.

In the following we will assume that the Markov chain is *homogeneous*, i.e. the probabilities $\mathbb{P}(X_{t+1} \in A|X_t = x_t)$ are independent of t. For the remainder of this section we shall also assume that we can express the probability from definition 1.23 using a *transition kernel* $K : S \times S \to \mathbb{R}_0^+$:

$$\mathbb{P}(X_{t+1} \in A|X_t = x_t) = \int_A K(x_t, x_{t+1})\, dx_{t+1} \tag{1.3}$$

where the integration is with respect to a suitable dominating measure, i.e. for example with respect to the Lebesgue measure if $S = \mathbb{R}^d$.[4] The transition kernel $K(x, y)$ is thus just the conditional probability density of X_{t+1} given $X_t = x_t$.

We obtain the special case of definition 1.8 by setting $K(i, j) = k_{ij}$, where k_{ij} is the (i, j)-th element of the transition matrix $\mathbf{K}$. For a discrete state space the dominating measure is the counting measure, so integration just corresponds to summation, i.e. equation (1.3) is equivalent to

$$\mathbb{P}(X_{t+1} \in A|X_t = x_t) = \sum_{x_{t+1} \in A} k_{x_t x_{t+1}}.$$

We have for measurable set $A \subset S$ that

$$\mathbb{P}(X_{t+m} \in A|X_t = x_t) = \int_A \int_S \cdots \int_S K(x_t, x_{t+1})K(x_{t+1}, x_{t+2}) \cdots K(x_{t+m-1}, x_{t+m})\, dx_{t+1} \cdots dx_{t+m-1} dx_{t+m},$$

thus the m-step transition kernel is

$$K^{(m)}(x_0, x_m) = \int_S \cdots \int_S K(x_0, x_1) \cdots K(x_{m-1}, x_m)\, dx_{m-1} \cdots dx_1$$

The m-step transition kernel allows for expressing the m-step transition probabilities more conveniently:

$$\mathbb{P}(X_{t+m} \in A|X_t = x_t) = \int_A K^{(m)}(x_t, x_{t+m})\, dx_{t+m}$$

Example 1.15 (Gaussian random walk on $\mathbb{R}$). Consider the random walk on $\mathbb{R}$ defined by

$$X_{t+1} = X_t + E_t,$$

where $E_t \sim \mathsf{N}(0, 1)$, i.e. the probability density function of E_t is $\phi(z) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right)$. This is equivalent to assuming that

$$X_{t+1}|X_t = x_t \sim \mathsf{N}(x_t, 1).$$

We also assume that E_t is independent of $X_0, E_1, \ldots, E_{t-1}$. Suppose that $X_0 \sim \mathsf{N}(0, 1)$. In contrast to the random walk on $\mathbb{Z}$ (introduced in example 1.2) the state space of the Gaussian random walk is $\mathbb{R}$. In complete analogy with example 1.2 we have that

$$\begin{aligned}\mathbb{P}(X_{t+1} \in A|X_t = x_t, \ldots, X_0 = x_0) &= \mathbb{P}(E_t \in A - x_t|X_t = x_t, \ldots, X_0 = x_0)\\ &= \mathbb{P}(E_t \in A - x_t) = \mathbb{P}(X_{t+1} \in A|X_t = x_t),\end{aligned}$$

where $A - x_t = \{a - x_t : a \in A\}$. Thus X is indeed a Markov chain. Furthermore we have that

$$\mathbb{P}(X_{t+1} \in A|X_t = x_t) = \mathbb{P}(E_t \in A - x_t) = \int_A \phi(x_{t+1} - x_t)\, dx_{t+1}$$

Thus the transition kernel (which is nothing other than the conditional density of $X_{t+1}|X_t = x_t$) is thus

$$K(x_t, x_{t+1}) = \phi(x_{t+1} - x_t)$$

To find the m-step transition kernel we could use equation (1.3). However, the resulting integral is difficult to compute. Rather we exploit the fact that

[4] A more correct way of stating this would be $\mathbb{P}(X_{t+1} \in A|X_t = x_t) = \int_A K(x_t, dx_{t+1})$.

$$X_{t+m} = X_t + \underbrace{E_t + \ldots + E_{t+m-1}}_{\sim \mathsf{N}(0,m)},$$

thus $X_{t+m}|X_t = x_t \sim \mathsf{N}(x_t, m)$.

$$\mathbb{P}(X_{t+m} \in A|X_t = x_t) = \mathbb{P}(X_{t+m} - X_t \in A - x_t) = \int_A \frac{1}{\sqrt{m}} \phi\left(\frac{x_{t+m} - x_t}{\sqrt{m}}\right)\, dx_{t+m}$$

Comparing this with (1.3) we can identify

$$K^{(m)}(x_t, x_{t+m}) = \frac{1}{\sqrt{m}} \phi\left(\frac{x_{t+m} - x_t}{\sqrt{m}}\right)$$

as m-step transition kernel. ◁

In section 1.2.2 we defined a Markov chain to be irreducible if there is a positive probability of getting from any state $i \in S$ to any other state $j \in S$, possibly via intermediate steps.

Again, we cannot directly apply definition 1.12 to Markov chains with general state spaces: it might be — as it is the case for a continuous state space — that the probability of hitting a given state is 0 for all states. We will again resolve this by looking at sets of states rather than individual states.

Definition 1.24 (Irreducibility). *Given a distribution μ on the states S, a Markov chain is said to be μ-*irreducible *if for all sets A with $\mu(A) > 0$ and for all $x \in S$, there exists an $m \in \mathbb{N}_0$ such that*

$$\mathbb{P}(X_{t+m} \in A|X_t = x) = \int_A K^{(m)}(x, y)\, dy > 0.$$

If the number of steps $m = 1$ for all A, then the chain is said to be strongly μ-irreducible.

Example 1.16 (Gaussian random walk (continued)). In example 1.15 we had that $X_{t+1}|X_t = x_t \sim \mathsf{N}(x_t, 1)$. As the range of the Gaussian distribution is $\mathbb{R}$, we have that $\mathbb{P}(X_{t+1} \in A|X_t = x_t) > 0$ for all sets A of non-zero Lebesgue measure. Thus the chain is strongly irreducible with the respect to any continuous distribution. ◁

Extending the concepts of periodicity, recurrence, and transience studied in sections 1.2.2 and 1.2.3 from the discrete case to the general case requires additional technical concepts like *atoms* and *small sets*, which are beyond the scope of this course (for a more rigorous treatment of these concepts see e.g. Robert and Casella, 2004, sections 6.3 and 6.4). Thus we will only generalise the concept of recurrence.

In section 1.2.3 we defined a discrete Markov chain to be recurrent, if all states are (on average) visited infinitely often. For more general state spaces, we need to consider the number of visits to a set of states rather than single states. Let $V_A = \sum_{t=0}^{+\infty} \mathbb{1}_{\{X_t \in A\}}$ be the number of visits the chain makes to states in the set $A \subset S$. We then define the expected number of visits in $A \subset S$, when we start the chain in $x \in S$:

$$\mathbb{E}(V_A|X_0 = x) = \mathbb{E}\left(\sum_{t=0}^{+\infty} \mathbb{1}_{\{X_t \in A\}} \middle| X_0 = x\right) = \sum_{t=0}^{+\infty} \mathbb{E}(\mathbb{1}_{\{X_t \in A\}}|X_0 = x) = \sum_{t=0}^{+\infty} \int_A K^{(t)}(x, y)\, dy$$

This allows us to define recurrence for general state spaces. We start with defining recurrence of sets before extending the definition of recurrence of an entire Markov chain.

Definition 1.25 (Recurrence). *(a) A set $A \subset S$ is said to be* recurrent *for a Markov chain X if for all $x \in A$*

$$\mathbb{E}(V_A|X_0 = x) = +\infty,$$

(b) A Markov chain is said to be recurrent, *if*

i. The chain is μ-irreducible for some distribution μ.

ii. Every measurable set $A \subset S$ with $\mu(A) > 0$ is recurrent.

According to the definition a set is recurrent if on average it is visited infinitely often. This is already the case if there is a non-zero probability of visiting the set infinitely often. A stronger concept of recurrence can be obtained if we require that the set is visited infinitely often with probability 1. This type of recurrence is referred to as *Harris recurrence*.

Definition 1.26 (Harris Recurrence). *(a) A set $A \subset S$ is said to be* Harris-recurrent *for a Markov chain X if for all $x \in A$*

$$\mathbb{P}(V_A = +\infty | X_0 = x) = 1,$$

(b) A Markov chain is said to be Harris-recurrent*, if*

i. The chain is μ-irreducible for some distribution μ.

ii. Every measurable set $A \subset S$ with $\mu(A) > 0$ is Harris-recurrent.

It is easy to see that Harris recurrence implies recurrence. For discrete state spaces the two concepts are equivalent.

Checking recurrence or Harris recurrence can be very difficult. We will state (without) proof a proposition which establishes that if a Markov chain is irreducible and has a unique invariant distribution, then the chain is also recurrent.

However, before we can state this proposition, we need to define invariant distributions for general state spaces.

Definition 1.27 (Invariant Distribution). *A distribution μ with density function f_μ is said to be the* invariant distribution *of a Markov chain X with transition kernel K if*

$$f_\mu(y) = \int_S f_\mu(x) K(x, y)\, dx$$

for almost all $y \in S$.

Proposition 1.28. *Suppose that X is a μ-irreducible Markov chain having μ as unique invariant distribution. Then X is also recurrent.*

Proof. see (Tierney, 1994, theorem 1) or (Athreya et al., 1992) □

Checking the invariance condition of definition 1.27 requires computing an integral, which can be quite cumbersome. A simpler (sufficient, but not necessary) condition is, just like in the case discrete case, detailed balance.

Definition 1.29 (Detailed balance). *A transition kernel K is said to be in* detailed balance *with a distribution μ with density f_μ if for almost all $x, y \in S$*

$$f_\mu(x) K(x, y) = f_\mu(y) K(y, x).$$

In complete analogy with theorem 1.22 one can also show in the general case that if the transition kernel of a Markov chain is in detailed balance with a distribution μ, then the chain is time-reversible and has μ as its invariant distribution. Thus theorem 1.22 also holds in the general case.

1.4 Ergodic theorems

In this section we will study the question whether we can use observations from a Markov chain to make inferences about its invariant distribution. We will see that under some regularity conditions it is even enough to follow a single sample path of the Markov chain.

For independent identically distributed data the Law of Large Numbers is used to justify estimating the expected value of a functional using empirical averages. A similar result can be obtained for Markov chains. This result is the reason why Markov Chain Monte Carlo methods work: it allows us to set up simulation algorithms to generate a Markov chain, whose sample path we can then use for estimating various quantities of interest.

Theorem 1.30 (Ergodic Theorem). *Let X be a μ-irreducible, recurrent $\mathbb{R}^d$-valued Markov chain with invariant distribution μ. Then we have for any integrable function $g : \mathbb{R}^d \to \mathbb{R}$ that with probability 1*

$$\lim_{t\to\infty} \frac{1}{t} \sum_{i=1}^{t} g(X_i) \to \mathbb{E}_\mu(g(X)) = \int_S g(x) f_\mu(x)\, dx$$

for almost every starting value $X_0 = x$. If X is Harris-recurrent this holds for every starting value x.

Proof. For a proof see (Roberts and Rosenthal, 2004, fact 5), (Robert and Casella, 2004, theorem 6.63), or (Meyn and Tweedie, 1993, theorem 17.3.2). □

Under additional regularity conditions one can also derive a Central Limit Theorem which can be used to justify Gaussian approximations for ergodic averages of Markov chains. This would however be beyond the scope of this course.

We conclude by giving an example that illustrates that the conditions of irreducibility and recurrence are necessary in theorem 1.30. These conditions ensure that the chain is permanently exploring the entire state space, which is a necessary condition for the convergence of ergodic averages.

Example 1.17. Consider a discrete chain with two states $S = \{1, 2\}$ and transition matrix

$$\mathbf{K} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

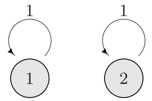


Figure 1.8. Markov graph of the chain of example 1.17

The corresponding Markov graph is shown in figure 1.8. This chain will remain in its intial state forever. Any distribution $\boldsymbol{\mu}$ on $\{1, 2\}$ is an invariant distribution, as

$$\boldsymbol{\mu}'\mathbf{K} = \boldsymbol{\mu}'\mathbf{I} = \boldsymbol{\mu}'$$

for all $\boldsymbol{\mu}$. However, the chain is not irreducible (or recurrent): we cannot get from state 1 to state 2 and vice versa. If the initial distribution is $\boldsymbol{\mu} = (\alpha, 1-\alpha)'$ with $\alpha \in [0, 1]$ then for every $t \in \mathbb{N}_0$ we have that

$$\mathbb{P}(X_t = 1) = \alpha \qquad \mathbb{P}(X_t = 2) = 1 - \alpha.$$

By observing one sample path (which is either $1, 1, 1, \ldots$ or $2, 2, 2, \ldots$) we can make no inference about the distribution of X_t or the parameter α. The reason for this is that the chain fails to explore the space (i.e. switch between the states 1 and 2). In order to estimate the parameter α we would need to look at more than one sample path. ◁

Note that theorem 1.30 does not require the chain to the aperiodic. In example 1.12 we studied a periodic chain. Due to the periodicity we could not apply theorem 1.19. We can however apply theorem 1.30 to this chain. The reason for this is that whilst theorem 1.19 was about the distribution of states at a given time t, theorem 1.30 is about averages, and the periodic behaviour does not affect averages.

Chapter 2

An Introduction to Monte Carlo Methods

2.1 What are Monte Carlo Methods?

This lecture course is concerned with Monte Carlo methods, which are sometimes referred to as *stochastic simulation* (Ripley (1987) for example only uses this term).

Examples of Monte Carlo methods include stochastic integration, where we use a simulation-based method to evaluate an integral, Monte Carlo tests, where we resort to simulation in order to compute the p-value, and Markov-Chain Monte Carlo (MCMC), where we construct a Markov chain which (hopefully) converges to the distribution of interest.

A formal definition of Monte Carlo methods was given (amongst others) by Halton (1970). He defined a Monte Carlo method as "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."

2.2 Introductory examples

Example 2.1 (A raindrop experiment for computing π). Assume we want to compute an Monte Carlo estimate of π using a simple experiment. Assume that we could produce "uniform rain" on the square $[-1, 1] \times [-1, 1]$, such that the probability of a raindrop falling into a region $\mathcal{R} \subset [-1, 1]^2$ is proportional to the area of $\mathcal{R}$, but independent of the position of $\mathcal{R}$. It is easy to see that this is the case iff the two coordinates X, Y are i.i.d. realisations of uniform distributions on the interval $[-1, 1]$ (in short $X, Y \overset{\text{i.i.d.}}{\sim} \mathsf{U}[-1, 1]$).

Now consider the probability that a raindrop falls into the unit circle (see figure 2.1). It is

$$\mathbb{P}(\text{drop within circle}) = \frac{\text{area of the unit circle}}{\text{area of the square}} = \frac{\displaystyle\iint_{\{x^2+y^2\leq 1\}} 1 \, dxdy}{\displaystyle\iint_{\{-1\leq x,y\leq 1\}} 1 \, dxdy} = \frac{\pi}{2\cdot 2} = \frac{\pi}{4}$$

In other words,

$$\pi = 4 \cdot \mathbb{P}(\text{drop within circle}),$$

i.e. we found a way of expressing the desired quantity π as a function of a probability.

Of course we cannot compute $\mathbb{P}(\text{drop within circle})$ without knowing π, however we can estimate the probability using our raindrop experiment. If we observe n raindrops, then the number of raindrops Z that fall inside the circle is a binomial random variable:

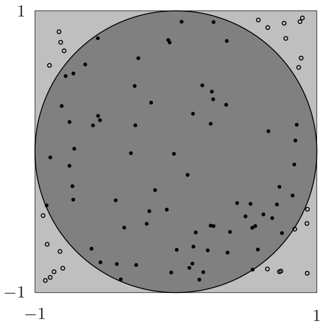


Figure 2.1. Illustration of the raindrop experiment for estimating π

$$Z \sim \mathsf{B}(n, p), \qquad \text{with } p = \mathbb{P}(\text{drop within circle}).$$

Thus we can estimate p by its maximum-likelihood estimate

$$\hat{p} = \frac{Z}{n},$$

and we can estimate π by

$$\hat{\pi} = 4\hat{p} = 4 \cdot \frac{Z}{n}.$$

Assume we have observed, as in figure 2.1, that 77 of the 100 raindrops were inside the circle. In this case, our estimate of π is

$$\hat{\pi} = \frac{4 \cdot 77}{100} = 3.08,$$

which is relatively poor.

However the *law of large numbers* guarantees that our estimate $\hat{\pi}$ converges almost surely to π. Figure 2.2 shows the estimate obtained after n iterations as a function of n for $n = 1, \ldots, 2000$. You can see that the estimate improves as n increases.

We can assess the quality of our estimate by computing a confidence interval for π. As we have $Z \sim \mathsf{B}(100, p)$ and $\hat{p} = \frac{Z}{n}$, we use the approximation that $Z \sim \mathsf{N}(100p, 100p(1-p))$. Hence, $\hat{p} \sim \mathsf{N}(p, p(1-p)/100)$, and we can obtain a 95% confidence interval for p using this Normal approximation:

$$\left[0.77 - 1.96 \cdot \sqrt{\frac{0.77 \cdot (1-0.77)}{100}},\ 0.77 + 1.96 \cdot \sqrt{\frac{0.77 \cdot (1-0.77)}{100}}\right] = [0.6875,\ 0.8525],$$

As our estimate of π is four times the estimate of p, we now also have a confidence interval for π:

$$[2.750,\ 3.410]$$

In more general, let $\hat{\pi}_n = 4\hat{p}_n$ denote the estimate after having observed n raindrops. A $(1-2\alpha)$ confidence interval for p is then

$$\left[\hat{p}_n - z_{1-\alpha}\sqrt{\frac{\hat{p}_n(1-\hat{p}_n)}{n}}, \hat{p}_n + z_{1-\alpha}\sqrt{\frac{\hat{p}_n(1-\hat{p}_n)}{n}}\right],$$

thus a $(1-2\alpha)$ confidence interval for π is

$$\left[\hat{\pi}_n - z_{1-\alpha}\sqrt{\frac{\hat{\pi}_n(4-\hat{\pi}_n)}{n}}, \hat{\pi}_n + z_{1-\alpha}\sqrt{\frac{\hat{\pi}_n(4-\hat{\pi}_n)}{n}}\right] \qquad \triangleleft$$

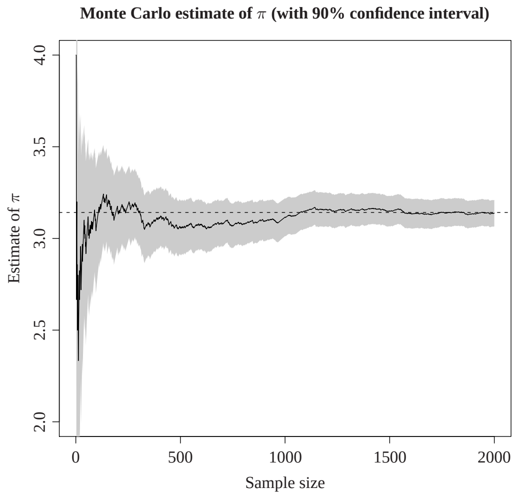


Figure 2.2. Estimate of π resulting from the raindrop experiment

Let us recall again the different steps we have used in the example:

- We have written the quantity of interest (in our case π) as an expectation.[1]
- Second, we have replaced this algebraic representation of the quantity of interest by a sample approximation to it. The law of large numbers guaranteed that the sample approximation converges to the algebraic representation, and thus to the quantity of interest. Furthermore we used the central limit theorem to assess the speed of convergence.

It is of course of interest whether the Monte Carlo methods offer more favourable rates of convergence than other numerical methods. We will investigate this in the case of Monte Carlo integration using the following simple example.

Example 2.2 (Monte Carlo Integration). Assume we want to evaluate the integral

$$\int_0^1 f(x)\,dx \qquad \text{with} \qquad f(x) = \frac{1}{27}\cdot\left(-65536x^8 + 262144x^7 - 409600x^6 + 311296x^5 - 114688x^4 + 16384x^3\right)$$

using a Monte Carlo approach.[2] Figure 2.3 shows the function for $x \in [0,1]$. Its graph is fully contained in the unit square $[0,1]^2$.

Once more, we can resort to a raindrop experiment. Assume we can produce uniform rain on the unit square. The probability that a raindrop falls below the curve is equal to the area below the curve, which of course equals the integral we want to evaluate (the area of the unit square is 1, so we don't need to rescale the result).

A more formal justification for this is, using the fact that $f(x) = \int_0^{f(x)} 1\ dt$,

$$\int_0^1 f(x)\ dx = \int_0^1 \int_0^{f(x)} 1\ dt\ dx = \iint\limits_{\{(x,t):t\le f(x)\}} 1dt\ dx = \frac{\iint\limits_{\{(x,t):t\le f(x)\}} 1\ dt\ dx}{\iint\limits_{\{0\le x,t\le 1\}} 1dt\ dx}$$

The numerator is nothing other than the dark grey area under the curve, and the denominator is the area of the unit square (shaded in light grey in figure 2.3). Thus the expression on the right hand side is the probability that a raindrop falls below the curve.

We have thus re-expressed our quantity of interest as a probability in a statistical model. Figure 2.3 shows the result obtained when observing 100 raindrops. 52 of them are below the curve, yielding a Monte-Carlo estimate of the integral of 0.52.

If after n raindrops a proportion $\hat{p}_n$ is found to lie below the curve, a $(1-2\alpha)$ confidence interval for the value of the integral is

$$\left[\hat{p}_n - z_{1-\alpha}\sqrt{\frac{\hat{p}_n(1-p_n)}{n}}, \hat{p}_n + z_{1-\alpha}\sqrt{\frac{\hat{p}_n(1-p_n)}{n}}\right]$$

Thus the speed of convergence of our (rather crude) Monte Carlo method is $O_{\mathbb{P}}(n^{-1/2})$. ◁

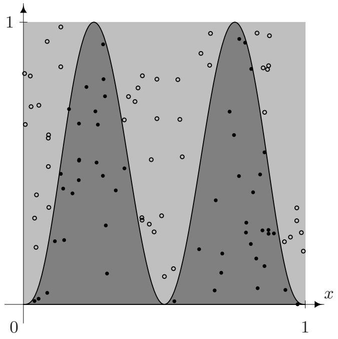


Figure 2.3. Illustration of the raindrop experiment to compute $\int_0^1 f(x)dx$

When using Riemann sums (as in figure 2.4) to approximate the integral from example 2.2 the error is of order $O(n^{-1})$.[3,4]

Recall that our Monte Carlo method was "only" of order $O_{\mathbb{P}}(n^{-1/2})$. However, it is easy to see that its speed of convergence is of the same order, regardless of the dimension of the support of f. This is not the case for other (deterministic) numerical integration methods. For a two-dimensional function f the error made by the Riemann approximation using n function evaluations is $O(n^{-1/2})$. [5]

This makes the Monte Carlo methods especially suited for high-dimensional problems. Furthermore the Monte Carlo method offers the advantage of being relatively simple and thus easy to implement on a computer.

2.3 A Brief History of Monte Carlo Methods

Experimental Mathematics is an old discipline: the Old Testament (1 Kings vii. 23 and 2 Chronicles iv. 2) contains a rough estimate of π (using the columns of King Solomon's temple). Monte Carlo methods are a somewhat more recent discipline. One of the first documented Monte Carlo experiments is *Buffon's needle* experiment (see example 2.3 below). Laplace (1812) suggested that this experiment can be used to approximate π.

[1] A probability is a special case of an expectation as $\mathbb{P}(A) = \mathbb{E}(\mathbb{I}_A)$.

[2] As f is a polynomial we can obtain the result analytically, it is $\frac{4096}{8505} = \frac{2^{12}}{3^5\cdot 5\cdot 7} \approx 0.4816$.

[3] The error made for each "bar" can be upper bounded by $\frac{\Delta^2}{2}\max|f'(x)|$. Let n denote the number evaluations of f (and thus the number of "bars"). As Δ is proportional to $\frac{1}{n}$, the error made for each bar is $O(n^{-2})$. As there are n "bars", the total error is $O(n^{-1})$.

[4] The order of convergence can be improved when using the trapezoid rule and (even more) by using Simpson's rule.

[5] Assume we partition both axes into m segments, i.e. we have to evaluate the function $n = m^2$ times. The error made for each "bar" is $O(m^{-3})$ (each of the two sides of the base area of the "bar" is proportional to m^{-1}, so is the upper bound on $|f(x) - f(\xi_{\text{mid}})|$, yielding $O(m^{-3})$). There are in total m^2 bars, so the total error is only $O(m^{-1})$, or equivalently $O(n^{-1/2})$.

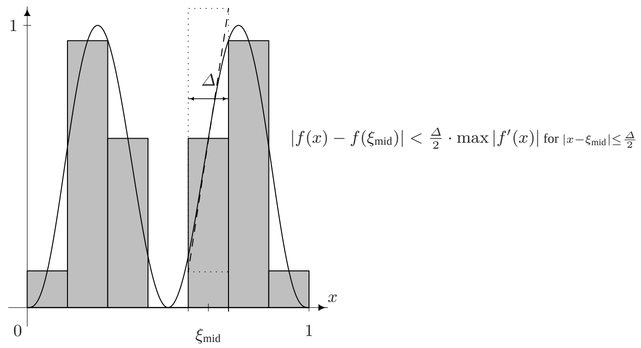


Figure 2.4. Illustration of numerical integration by Riemann sums

Example 2.3 (Buffon's needle). In 1733, the Comte de Buffon, George Louis Leclerc, asked the following question (Buffon, 1733): Consider a floor with equally spaced lines, a distance δ apart. What is the probability that a needle of length $l < \delta$ dropped on the floor will intersect one of the lines?
Buffon answered the question himself in 1777 (Buffon, 1777).
Assume the needle landed such that its angle is θ (see figure 2.5). Then the question whether the needle intersects a line is equivalent to the question whether a box of width $l \sin\theta$ intersects a line. The probability of this happening is

$$\mathbb{P}(\text{intersect}|\theta) = \frac{l \sin\theta}{\delta}.$$

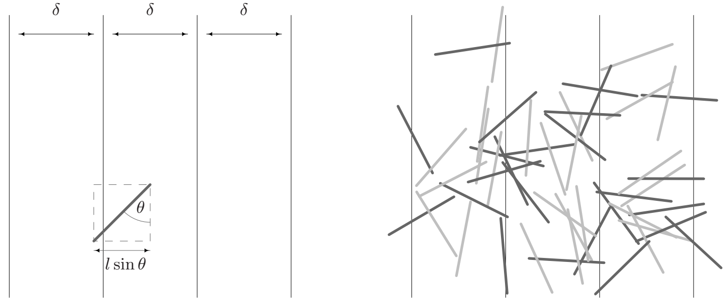
Assuming that the angle θ is uniform on $[0, \pi)$ we obtain

$$\mathbb{P}(\text{intersect}) = \int_0^\pi \mathbb{P}(\text{intersect}|\theta) \cdot \frac{1}{\pi}\, d\theta = \int_0^\pi \frac{l \sin\theta}{\delta} \cdot \frac{1}{\pi}\, d\theta = \frac{l}{\pi\delta} \cdot \underbrace{\int_0^\pi \sin\theta\, d\theta}_{=2} = \frac{2l}{\pi\delta}.$$

When dropping n needles the expected number of needles crossing a line is thus

$$\frac{2nl}{\pi\delta}.$$

Thus we can estimate π by



(a) Illustration of the geometry behind *Buffon's needle*

(b) Results of the *Buffon's needle* experiment using 50 needles. Dark needles intersect the thin vertical lines, light needles do not.

Figure 2.5. Illustration of *Buffon's needle*

$$\pi \approx \frac{2nl}{X\delta},$$

where X is the number of needles crossing a line.
The Italian mathematician Mario Lazzarini performed Buffon's needle experiment in 1901 using a needle of length $l = 2.5cm$ and lines $d = 3cm$ apart (Lazzarini, 1901). Of 3408 needles 1808 needles crossed a line, so Lazzarini's estimate of π was

$$\pi \approx \frac{2 \cdot 3408 \cdot 2.5}{1808 \cdot 3} = \frac{17040}{5424} = \frac{355}{133},$$

which is nothing other than the best rational approximation to π with at most 4 digits each in the denominator and the numerator.[6] ◁

Historically, the main drawback of Monte Carlo methods was that they used to be expensive to carry out. Physical random experiments were difficult to perform and so was the numerical processing of their results.

This however changed fundamentally with the advent of the digital computer. Amongst the first to realise this potential were John von Neuman and Stanisław Ulam, who were then working for the Manhattan project in Los Alamos. They proposed in 1947 to use a computer simulation for solving the problem of neutron diffusion in fissionable material (Metropolis, 1987). Enrico Fermi previously considered using Monte Carlo techniques in the calculation of neutron diffusion, however he proposed to use a mechanical device, the so-called "Fermiac", for generating the randomness. The name "Monte Carlo" goes back to Stanisław Ulam, who claimed to be stimulated by playing poker (Ulam, 1983). In 1949 Metropolis and Ulam published their results in the *Journal of the American Statistical Association* (Metropolis and Ulam, 1949). Nonetheless, in the following 30 years Monte Carlo methods were used and analysed predominantly by physicists, and not by statisticians: it was only in the 1980s — following the paper by Geman and Geman (1984) proposing the Gibbs sampler — that the relevance of Monte Carlo methods in the context of (Bayesian) statistics was fully realised.

2.4 Pseudo-random numbers

For any Monte-Carlo simulation we need to be able to reproduce randomness by a computer algorithm, which, by definition, is deterministic in nature — a philosophical paradox. In the following chapters we will assume that independent (pseudo-)random realisations from a uniform $\mathsf{U}[0, 1]$ distribution[7] are readily available. This section tries to give very brief overview of how pseudo-random numbers can be generated. For a more detailed discussion of pseudo-random number generators see Ripley (1987) or Knuth (1997).

A pseudo-random number generator (RNG) is an algorithm for whose output the $\mathsf{U}[0, 1]$ distribution is a suitable model. In other words, the number generated by the pseudo-random number generator should have the same *relevant* statistical properties as independent realisations of a $\mathsf{U}[0, 1]$ random variable. Most importantly:

- The numbers generated by the algorithm should reproduce independence, i.e. the numbers $X_1, \ldots, X_n$ that we have already generated should not contain any discernible information on the next value X_{n+1}. This property is often referred to as the lack of predictability.
- The numbers generated should be spread out evenly across the interval $[0, 1]$.

In the following we will briefly discuss the linear congruential generator. It is not a particularly powerful generator (so we discourage you from using it in practise), however it is easy enough to allow some insight into how pseudo-random number generators work.

[6] That Lazzarini's experiment was that precise, however, casts some doubt over the results of his experiments (see Badger, 1994, for a more detailed discussion).

[7] We will only use the $\mathsf{U}(0, 1)$ distribution as a source of randomness. Samples from other distributions can be derived from realisations of $\mathsf{U}(0, 1)$ random variables using deterministic algorithms.

Algorithm 2.1 (Congruential pseudo-random number generator).

```
1. Choose $a, M \in \mathbb{N}$, $c \in \mathbb{N}_0$, and the initial value ("seed") $Z_0 \in \{1, \dots M-1\}$.
2. For $i = 1, 2, \dots$
     Set $Z_i = (aZ_{i-1} + c) \mod M$, and $X_i = Z_i/M$.
```

The integers Z_i generated by the algorithm are from the set $\{0, 1, \dots, M-1\}$ and thus the X_i are in the interval $[0, 1)$.

It is easy to see that the sequence of pseudo-random numbers only depends on the seed Z_0. Running the pseudo-random number generator twice with the same seed thus generates exactly the same sequence of pseudo-random numbers. This can be a very useful feature when debugging your own code.

Example 2.4. Cosider the choice of $a = 81$, $c = 35$, $M = 256$, and seed $Z_0 = 4$.

$$\begin{aligned} Z_1 &= (81 \cdot 4 + 35) \mod 256 = 359 \mod 256 = 103 \\ Z_2 &= (81 \cdot 103 + 35) \mod 256 = 8378 \mod 256 = 186 \\ Z_3 &= (81 \cdot 186 + 35) \mod 256 = 15101 \mod 256 = 253 \\ &\dots \end{aligned}$$

The corresponding X_i are $X_1 = 103/256 = 0.4023438$, $X_2 = 186/256 = 0.72656250$, $X_1 = 253/256 = 0.98828120$. ◁

The main flaw of the congruential generator its "crystalline" nature (Marsaglia, 1968). If the sequence of generated values $X_1, X_2, \dots$ is viewed as points in an n-dimension cube[8], they lie on a finite, and often very small number of parallel hyperplanes. Or as Marsaglia (1968) put it: "the points [generated by a congruential generator] are about as randomly spaced in the unit n-cube as the atoms in a perfect crystal at absolute zero." The number of hyperplanes depends on the choice of a, c, and M.

An example for a notoriously poor design of a congruential pseudo-random number generator is RANDU, which was (unfortunately) very popular in the 1970s and used for example in IBM's System/360 and System/370, and Digital's PDP-11. It used $a = 2^{16} + 3$, $c = 0$, and $M = 2^{31}$. The numbers generated by RANDU lie on only 15 hyperplanes in the 3-dimensional unit cube (see figure 2.6).

Figure 2.7 shows another cautionary example (taken from Ripley, 1987). The left-hand panel shows a plot of 1,000 realisations of a congruential generator with $a = 1229$, $c = 1$, and $M = 2^{11}$. The random numbers lie on only 5 hyperplanes in the unit square. The right hand panel shows the outcome of the Box-Muller method for transforming two uniform pseudo-random numbers into a pair of Gaussians (see example 3.2).

Due to this flaw of the congruential pseudo-random number generator, it should not be used in Monte Carlo experiments. For more powerful pseudo-random number generators see e.g. Marsaglia and Zaman (1991) or Matsumoto and Nishimura (1998). GNU R (and other environments) provide you with a large choice of powerful random number generators, see the corresponding help page (`?RNGkind`) for details.

[8] The $(k+1)$-th point has the coordinates $(X_{nk+1}, \dots, X_{nk+n-1})$.

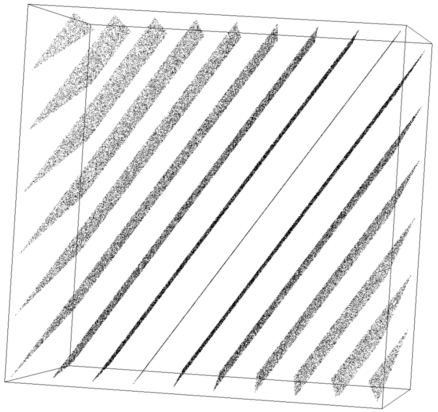
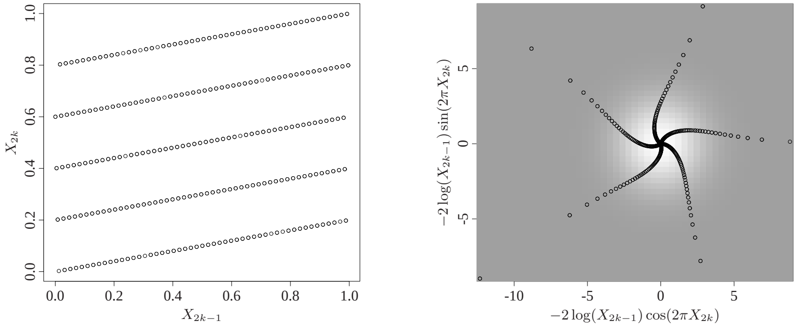


Figure 2.6. 300,000 realisations of the RANDU pseudo-random number generator plotted in 3D. A point corresponds to a triplet $(x_{3k-2}, x_{3k-1}, x_{3k})$ for $k = 1, \dots, 100000$. The data points lie on 15 hyperplanes.



(a) 1,000 realisations of this congruential generator plotted in 2D.

(b) Supposedly bivariate Gaussian pseudo-random numbers obtained using the pseudo-random numbers shown in panel (a).

Figure 2.7. Results obtained using a congruential generator with $a = 1229$, $c = 1$, and $M = 2^{11}$

Chapter 3

Fundamental Concepts: Transformation, Rejection, and Reweighting

3.1 Transformation methods

In section 2.4 we have seen how to create (pseudo-)random numbers from the uniform distribution $\mathsf{U}[0,1]$. One of the simplest methods of generating random samples from a distribution with cumulative distribution function (c.d.f.) $F(x) = \mathbb{P}(X \leq x)$ is based on the inverse of the c.d.f..

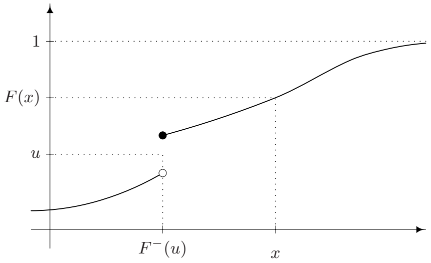


Figure 3.1. Illustration of the definition of the generalised inverse F^{-} of a c.d.f. F

The c.d.f. is an increasing function, however it is not necessarily continuous. Thus we define the *generalised inverse* $F^{-}(u) = \inf\{x : \ F(x) \geq u\}$. Figure 3.1 illustrates its definition. If F is continuous, then $F^{-}(u) = F^{-1}(u)$.

Theorem 3.1 (Inversion Method). *Let $U \sim \mathsf{U}[0,1]$ and F be a c.d.f.. Then $F^{-}(U)$ has the c.d.f. F.*

Proof. It is easy to see (e.g. in figure 3.1) 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 c.d.f. of $X = F^{-}(U)$. □

Example 3.1 (Exponential Distribution). The exponential distribution with rate $\lambda > 0$ has the c.d.f. $F_\lambda(x) = 1 - \exp(-\lambda x)$ for $x \geq 0$. Thus $F_\lambda^{-}(u) = F_\lambda^{-1}(u) = -\log(1-u)/\lambda$. Thus we can generate random samples from $\mathsf{Expo}(\lambda)$ 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 use $-\log(U)/\lambda$ as well. ◁

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

Note however that the generalised inverse of the c.d.f. is just one possible transformation and that there might be other transformations that yield the desired distribution. An example of such a method is the Box-Muller method for generating Gaussian random variables.

Example 3.2 (Box-Muller Method for Sampling from Gaussians). When sampling from the normal distribution, one faces the problem that neither the c.d.f. $\Phi(\cdot)$, nor its inverse has a closed-form expression. Thus we cannot use the inversion method.

It turns out however, that if we consider a pair $X_1, X_2 \overset{\text{i.i.d.}}{\sim} \mathsf{N}(0,1)$, as a point (X_1, X_2) in the plane, then its polar coordinates (R, θ) are again independent and have distributions we can easily sample from: $\theta \sim \mathsf{U}[0, 2\pi]$, and $R^2 \sim \mathsf{Expo}(1/2)$.

This can be shown as follows. Assume that $\theta \sim \mathsf{U}[0,2\pi]$ and $R^2 \sim \mathsf{Expo}(1/2)$. Then the joint density of (θ, r^2) is

$$f_{(\theta,r^2)}(\theta, r^2) = \frac{1}{2\pi}\mathbb{1}_{[0,2\pi]}(\theta) \cdot \frac{1}{2}\exp\left(-\frac{1}{2}r^2\right) = \frac{1}{4\pi}\exp\left(-\frac{1}{2}r^2\right) \cdot \mathbb{1}_{[0,2\pi]}(\theta)$$

To obtain the probability density function of

$$X_1 = \sqrt{R^2} \cdot \cos(\theta), \qquad X_2 = \sqrt{R^2} \cdot \sin(\theta)$$

we need to use the transformation of densities formula.

$$\begin{aligned} f_{(X_1,X_2)}(x_1,x_2) &= f_{(\theta,r^2)}(\theta(x_1,x_2), r^2(x_1,x_2)) \cdot \left| \begin{array}{cc} \frac{\partial x_1}{\partial \theta} & \frac{\partial x_1}{\partial r^2} \\ \frac{\partial x_2}{\partial \theta} & \frac{\partial x_2}{\partial r^2} \end{array} \right|^{-1} = \frac{1}{4\pi}\exp\left(-\frac{1}{2}(x_1^2 + x_2^2)^2\right) \cdot 2 \\ &= \left(\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}x_1^2\right)\right) \cdot \left(\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}x_2^2\right)\right) \end{aligned}$$

as

$$\left| \begin{array}{cc} \frac{\partial x_1}{\partial \theta} & \frac{\partial x_1}{\partial r^2} \\ \frac{\partial x_2}{\partial \theta} & \frac{\partial x_2}{\partial r^2} \end{array} \right| = \left| \begin{array}{cc} -r\sin(\theta) & \frac{\cos(\theta)}{2r} \\ r\cos(\theta) & \frac{\sin(\theta)}{2r} \end{array} \right| = \left| -\frac{r\sin(\theta)^2}{2r} - \frac{r\cos(\theta)^2}{2r} \right| = \frac{1}{2}$$

Thus $X_1, X_2 \sim \mathsf{N}(0,1)$. As their joint density factorises, X_1 and X_2 are independent, as required.

Thus we only need to generate $\theta \sim \mathsf{U}[0, 2\pi]$, and $R^2 \sim \mathsf{Expo}(1/2)$. Using $U_1, U_2 \overset{\text{i.i.d.}}{\sim} \mathsf{U}[0,1]$ and example 3.1 we can generate $R = \sqrt{R^2}$ 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. In many situations, we cannot find such a transformation in closed form. In these cases we have to find other ways of correcting for the fact that we sample from the "wrong" distribution. The next two sections present two such ideas: rejection sampling and importance sampling.

3.2 Rejection sampling

The basic idea of rejection sampling is to sample from an *instrumental distribution*[1] and reject samples that are "unlikely" under the target distribution.

Assume that we want to sample from a target distribution whose density f is known to us. The simple idea underlying rejection sampling (and other Monte Carlo algorithms) is the rather trivial identity

$$f(x) = \int_0^{f(x)} 1 \; du = \int_0^1 \underbrace{1_{0<u<f(x)}}_{=f(x,u)} du$$

Thus $f(x)$ can be interpreted as the marginal density of a uniform distribution on the area under the density $f(x)$

$$\{(x,u) : \; 0 \le u \le f(x)\}.$$

Figure 3.2 illustrates this idea. This suggests that we can generate a sample from f by sampling from the area under the curve.

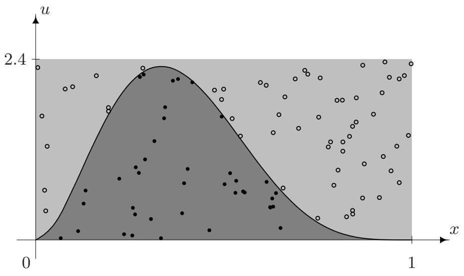


Figure 3.2. Illustration of example 3.3. Sampling from the area under the curve (dark grey) corresponds to sampling from the $\mathsf{Beta}(3,5)$ density. In example 3.3 we use a uniform distribution of the light grey rectangle as proposal distribution. Empty circles denote rejected values, filled circles denote accepted values.

Example 3.3 (Sampling from a Beta distribution). The $\mathsf{Beta}(a,b)$ distribution ($a, b \ge 0$) has the density

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

where $\Gamma(a) = \int_0^{+\infty} t^{a-1} \exp(-t) \; dt$ is the Gamma function. For $a, b > 1$ the $\mathsf{Beta}(a,b)$ density is unimodal with mode $(a-1)/(a+b-2)$. Figure 3.2 shows the density of a $\mathsf{Beta}(3,5)$ distribution. It attains its maximum of $1680/729 \approx 2.305$ at $x = 1/3$.

Using the above identity we can draw from $\mathsf{Beta}(3,5)$ 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 3.2).

In order to sample from the area under the density, we will use a similar trick as in examples 2.1 and 2.2. We will sample from the light grey rectangle and only keep the samples that fall in the area under the curve. Figure 3.2 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

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

[1] The instrumental distribution is sometimes referred to as *proposal distribution*.

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 3.3 is based on bounding the density of the Beta distribution by a box. Whilst this is a powerful idea, it cannot be directly applied to other distributions, as the density might be unbounded or have infinite support. However we might be able to bound the density of $f(x)$ by $M \cdot g(x)$, where $g(x)$ is a density that we can easily sample from.

Algorithm 3.1 (Rejection sampling). Given two densities f, g with $f(x) < M \cdot g(x)$ 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. We have

$$\mathbb{P}(X \in \mathcal{X} \text{ and is accepted}) = \int_{\mathcal{X}} g(x) \underbrace{\frac{f(x)}{M \cdot g(x)}}_{=\mathbb{P}(X \text{ is accepted}|X=x)} dx = \frac{\int_{\mathcal{X}} f(x) \; dx}{M}, \tag{3.1}$$

and thus[2]

$$\mathbb{P}(X \text{ is accepted}) = \mathbb{P}(X \in S \text{ and is accepted}) = \frac{1}{M}, \tag{3.2}$$

yielding

$$\mathbb{P}(x \in \mathcal{X}|X \text{ is accepted}) = \frac{\mathbb{P}(X \in \mathcal{X} \text{ and is accepted})}{\mathbb{P}(X \text{ is accepted})} = \frac{\int_{\mathcal{X}} f(x) \; dx/M}{1/M} = \int_{\mathcal{X}} f(x) \; dx. \tag{3.3}$$

Thus the density of the values accepted by the algorithm is $f(\cdot)$. □

Remark 3.2. If we know f only up to a multiplicative constant, i.e. if we only know $\pi(x)$, where $f(x) = C \cdot \pi(x)$, we can carry out rejection sampling using

$$\frac{\pi(X)}{M \cdot g(X)}$$

as probability of rejecting X, provided $\pi(x) < M \cdot g(x)$ for all x. Then by analogy with (3.1) - (3.3) we have

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

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

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

Example 3.4 (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

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

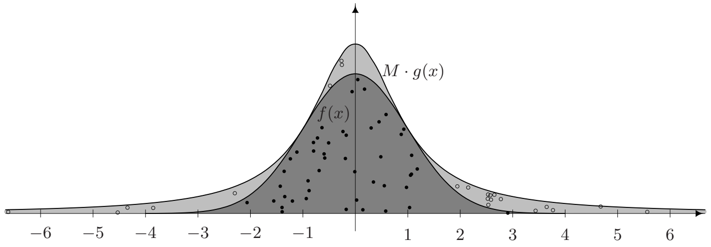


Figure 3.3. Illustration of example 3.3. Sampling from the area under the density $f(x)$ (dark grey) corresponds to sampling from the $\mathsf{N}(0,1)$ density. The proposal $g(x)$ is a $\mathsf{Cauchy}(0,1)$.

as instrumental distribution.[3] The smallest M we can choose such that $f(x) \leq Mg(x)$ is $M = \sqrt{2\pi} \cdot \exp(-1/2)$. Figure 3.3 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 from a Cauchy distribution using a $\mathsf{N}(0,1)$ distribution as instrumental distribution: there is no $M \in \mathbb{R}$ such that

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

the Cauchy distribution has heavier tails than the Gaussian distribution. ◁

3.3 Importance sampling

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

Importance sampling is based on the identity

$$\mathbb{P}(X \in A) = \int_A f(x)\, dx = \int_A g(x) \underbrace{\frac{f(x)}{g(x)}}_{=:w(x)} dx = \int_A g(x) w(x)\, dx \tag{3.4}$$

for all $g(\cdot)$, 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:

$$\mathbb{E}_f(h(X)) = \int_S f(x)h(x)\, dx = \int_S g(x) \underbrace{\frac{f(x)}{g(x)}}_{=:w(x)} h(x)\, dx = \int_S g(x)w(x)h(x)\, dx = \mathbb{E}_g(w(X) \cdot h(X)), \tag{3.5}$$

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 $\mathbb{E}_g|w(X) \cdot h(X)|$ exists,

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

(by the Law of Large Numbers) and thus by (3.5)

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

[2] We denote by S the set of all possible values X can take, i.e., $\int_S f(x)dx = 1$.

[3] There is not much point is using this method is practise. The Box-Muller method is more efficient.

In other words, we can estimate $\mu = \mathbb{E}_f(h(X))$ by

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

Note that whilst $\mathbb{E}_g(w(X)) = \int_S \frac{f(x)}{g(x)} g(x)\, dx = \int_S f(x) = 1$, 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 3.2 (Importance Sampling). Choose g such that supp$(g) \supset$ 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
      $\tilde{\mu} = \frac{\sum_{i=1}^{n} w(X_i)h(X_i)}{n}$
```

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

Theorem 3.3 (Bias and Variance of Importance Sampling). *(a)* $\mathbb{E}_g(\tilde{\mu}) = \mu$

(b) $\mathrm{Var}_g(\tilde{\mu}) = \dfrac{\mathrm{Var}_g(w(X) \cdot h(X))}{n}$

(c) $\mathbb{E}_g(\hat{\mu}) = \mu + \dfrac{\mu \mathrm{Var}_g(w(X)) - \mathrm{Cov}_g(w(X), w(X) \cdot h(X))}{n} + O(n^{-2})$

(d) $\mathrm{Var}_g(\hat{\mu}) = \dfrac{\mathrm{Var}_g(w(X) \cdot h(X)) - 2\mu \mathrm{Cov}_g(w(X), w(X) \cdot h(X)) + \mu^2 \mathrm{Var}_g(w(X))}{n} + O(n^{-2})$

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

(b) $\mathrm{Var}_g\left(\frac{1}{n}\sum_{i=1}^{n} w(X_i)h(X_i)\right) = \frac{1}{n^2}\sum_{i=1}^{n} \mathrm{Var}_g(w(X_i)h(X_i)) = \frac{\mathrm{Var}_g(w(X)h(X))}{n}$

(c) and (d) see (Liu, 2001, p. 35) □

Note that the theorem implies that in contrast 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 it is often the case in hierarchical Bayesian modelling. Assume $f(x) = C \cdot \pi(x)$, then

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

i.e. the self-normalised estimator $\hat{\mu}$ does not depend on the normalisation constant C.[4] On the other hand, as we have seen in the proof of theorem 3.3 it is a lot harder to analyse the theoretical properties of the self-normalised estimator $\hat{\mu}$.

Although the above equations (3.4) and (3.5) hold for every g with supp$(g) \supset$ 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) for a finite variance of $\tilde{\mu}$:

[4] By complete analogy one can show that is enough to know g up to a multiplicative constant.

- $f(x) < M \cdot g(x)$ and $\mathrm{Var}_f(h(X)) < +\infty$.
- S is compact, f is bounded above on S, and g is bounded below on S.

Note that under the first condition rejection sampling can also be used to sample from f.

So far we have only studied whether a distribution g leads to a finite-variance estimator. This leads to the question which instrumental distribution is *optimal*, i.e. for which choice $\mathrm{Var}(\tilde{\mu})$ is minimal. The following theorem answers this question:

Theorem 3.4 (Optimal proposal). *The proposal distribution g that minimises the variance of $\tilde{\mu}$ is*

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

Proof. We have from theroem 3.3 (b) that

$$n{\cdot}\mathrm{Var}_g(\tilde{\mu}) = \mathrm{Var}_g\left(w(X)\cdot h(X)\right) = \mathrm{Var}_g\left(\frac{h(X)\cdot f(X)}{g(X)}\right) = \mathbb{E}_g\left(\left(\frac{h(X)\cdot f(X)}{g(X)}\right)^2\right) - \Bigg(\underbrace{\mathbb{E}_g\left(\frac{h(X)\cdot f(X)}{g(X)}\right)}_{=\mathbb{E}_g(\tilde{\mu})=\mu}\Bigg)^2.$$

Thus we only have to minimise $\mathbb{E}_g\left(\left(\frac{h(X)\cdot f(X)}{g(X)}\right)^2\right)$. When plugging in $g^\star$ we obtain:

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

On the other hand, we can apply the Jensen inequality [5] to $\mathbb{E}_g\left(\left(\frac{h(X)\cdot f(X)}{g(X)}\right)^2\right)$ yielding

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

□

An important corollary of theorem 3.4 is that importance sampling can be *super-efficient*, i.e. when using the optimal $g^\star$ from theorem 3.4 the variance of $\tilde{\mu}$ is less than the variance obtained when sampling directly from f:

$$\begin{aligned} n\cdot\mathrm{Var}_f\left(\frac{h(X_1)+\ldots+h(X_n)}{n}\right) &= \mathbb{E}_f(h(X)^2)-\mu^2\\ \geq\quad (\mathbb{E}_f|h(X)|)^2-\mu^2 &= \left(\int_S |h(x)|f(x)\,dx\right)^2-\mu^2 = n\cdot\mathrm{Var}_{g^\star}(\tilde{\mu})\end{aligned}$$

by Jensen's inequality. Unless $h(X)$ 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 of high probability where $|h|$ is large, which contribute most to the integral $\mathbb{E}_f(h(X))$.

Theorem 3.4 is, however, a rather formal optimality result. When using $\tilde{\mu}$ we need to know the normalisation constant of $g^\star$, which is exactly the integral we are looking for. Further we need to be able to draw samples from $g^\star$ efficiently. The practically important corollary of theorem 3.4 is that we should choose an instrumental distribution g whose shape is close to the one of $f\cdot|h|$.

Example 3.5 (Computing $\mathbb{E}_f|X|$ 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. Three different schemes are considered

[5] If X is real-valued random variable, and ψ a convex function, then $\psi(\mathbb{E}(X)) \leq \mathbb{E}(\psi(X))$.

- 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 instrumental 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 3.4 shows.
- Third, we will consider importance sampling using a $\mathsf{N}(0,1)$ distribution as instrumental distribution.

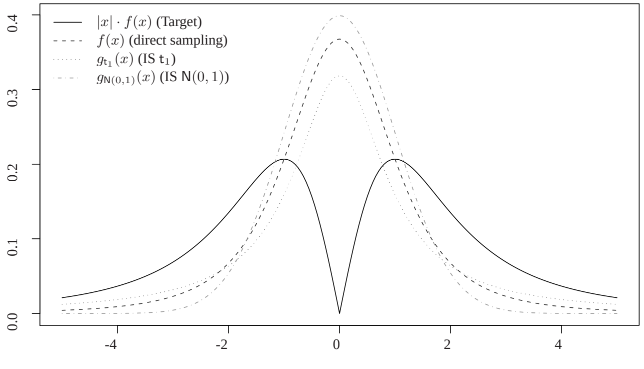


Figure 3.4. Illustration of the different instrumental distributions in example 3.5.

Note that the third choice yields weights of infinite variance, as the instrumental distribution ($\mathsf{N}(0,1)$) has lighter tails than the distribution we want to sample from (t_3). The right-hand panel of figure 3.5 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. This however 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 instrumental distribution, which is 0.0182. So using a t_1 distribution as instrumental distribution is super-efficient (see figure 3.5).

Figure 3.6 somewhat explains why the t_1 distribution is a far better choice than the $\mathsf{N}(0,1)$ distributon. As the $\mathsf{N}(0,1)$ distribution does not have heavy enough tails, the weight tends to infinity as $|x| \to +\infty$. Thus large $|x|$ get large weights, causing the jumps of the estimate $\tilde{\mu}$ shown in figure 3.5. The t_1 distribution has heavy enough tails, so 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 instrumental distribution. ◁

Example 3.6 (Partially labelled data). Suppose that we are given count data from observations in two groups, such that

$$\begin{aligned} &Y_i \sim \mathsf{Poi}(\lambda_1) &&\text{if the } i\text{-th observation is from group 1}\\ &Y_i \sim \mathsf{Poi}(\lambda_2) &&\text{if the } i\text{-th observation is from group 2}\end{aligned}$$

The data is given in the table 3.1. Note that only the first ten observations are labelled, the group label is missing for the remaining ten observations.

We will use a $\mathsf{Gamma}(\alpha,\beta)$ distribution as (conjugate) prior distribution for λ_j, i.e. the prior density of λ_j is

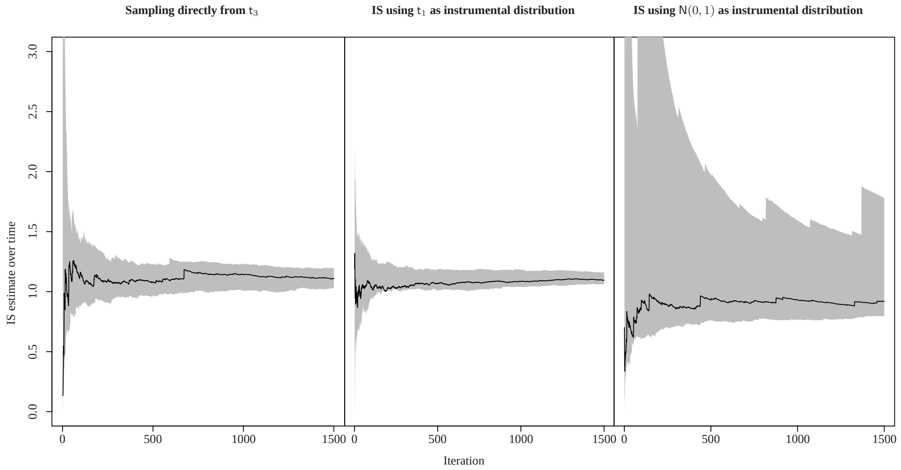


Figure 3.5. Estimates of $\mathbb{E}|X|$ for $X \sim t_3$ obtained after 1 to 1500 iterations. The three panels correspond to the three different sampling schemes used. The areas shaded in grey correspond to the range of 100 replications.

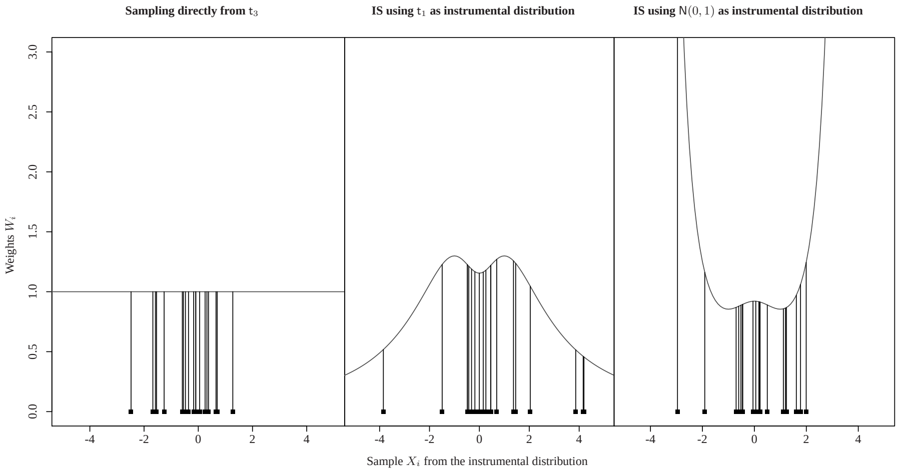


Figure 3.6. Weights W_i obtained for 20 realisations X_i from the different instrumental distributions.

Group	Count Y_i	Group	Count Y_i	Group	Count Y_i	Group	Count Y_i
1	3	2	14	*	15	*	21
1	6	2	12	*	4	*	11
1	3	2	11	*	1	*	3
1	5	2	19	*	6	*	7
1	9	2	18	*	11	*	18

Table 3.1. Data of example 3.6.

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

Furthermore, we believe that a priori each observation is equally likely to stem from group 1 or group 2.

We start with analysing the labelled data only, ignoring the 10 unlabelled observations. In this case, we can analyse the two groups separately. In group 1 we have that the joint distribution of $Y_1, \ldots, Y_5, \lambda_1$ is given by

$$\begin{aligned} f(y_1, \ldots, y_5, \lambda_1) = f(y_1, \ldots, y_5|\lambda_1) f(\lambda_1) &= \left(\prod_{i=1}^{5} \frac{\exp(-\lambda_1)\lambda_1^{y_i}}{y_i!} \right) \cdot \frac{1}{\Gamma(\alpha)} \lambda_1^{\alpha-1} \beta^{\alpha} \exp(-\beta\lambda) \\ &= \frac{1}{\prod_{i=1}^{5} y_i!} \cdot \frac{1}{\Gamma(\alpha)} \lambda_1^{\alpha+\sum_{i=1}^{5} y_i} \beta^{\alpha} \exp(-(\beta+5)\lambda_1) \propto \lambda_1^{\alpha+\sum_{i=1}^{5} y_i} \exp(-(\beta+5)\lambda_1) \end{aligned}$$

The posterior distribution of λ_1 given the data from group 1 is

$$\begin{aligned} f(\lambda_1|y_1, \ldots, y_5) = \frac{f(y_1, \ldots, y_5, \lambda_1)}{\int_\lambda f(y_1, \ldots, y_5, \lambda)\, d\lambda} &\propto f(y_1, \ldots, y_5, \lambda_1) \\ &\propto \lambda_1^{\alpha+\sum_{i=1}^{5} y_i} \exp(-(\beta+5)\lambda_1) \end{aligned}$$

Comparing this to the density of the Gamma distribution we obtain that

$$\lambda_1 | Y_1, \ldots, Y_5 \sim \mathsf{Gamma}\left(\alpha + \sum_{i=1}^{5} y_i, \beta + 5 \right),$$

and similarly

$$\lambda_2 | Y_6, \ldots, Y_{10} \sim \mathsf{Gamma}\left(\alpha + \sum_{i=6}^{10} y_i, \beta + 5 \right).$$

Thus, when only using the labelled data, we do not have to resort to Monte Carlo methods for finding the posterior distribution.

This however is not the case any more once we also want to include the unlabelled data. The conditional density of $Y_i|\lambda_1, \lambda_2$ for an unlabelled observation ($i > 10$) is

$$f(y_i|\lambda_1, \lambda_2) = \frac{1}{2} \frac{\exp(-\lambda_1)\lambda_1^{y_i}}{y_i!} + \frac{1}{2} \frac{\exp(-\lambda_2)\lambda_2^{y_i}}{y_i!}$$

The posterior density for the entire sample (using both labelled and unlabelled data) is

$$\begin{aligned} f(\lambda_1, \lambda_2|y_1, \ldots, y_{20}) &\propto \underbrace{f(\lambda_1) f(y_1, \ldots, y_5|\lambda_1)}_{\propto f(\lambda_1|y_1, \ldots, y_5)} \underbrace{f(\lambda_2) f(y_6, \ldots, y_{10}|\lambda_2)}_{\propto f(\lambda_2|y_6, \ldots, y_{10})} \cdot \underbrace{f(y_{11}, \ldots, y_{20}|\lambda_1, \lambda_2)}_{=\prod_{i=11}^{20} f(y_i|\lambda_1, \lambda_2)} \\ &\propto f(\lambda_1|y_1, \ldots, y_5) f(\lambda_2|y_6, \ldots, y_{10}) \prod_{i=11}^{20} f(y_i|\lambda_1, \lambda_2) \end{aligned}$$

This suggests using importance sampling with the product of the distributions of $\lambda_1|Y_1, \ldots, Y_5$ and $\lambda_2|Y_6, \ldots, Y_{10}$ as instrumental distributions, i. e. use

$$g(\lambda_1, \lambda_2) = f(\lambda_1|y_1, \ldots, y_5) f(\lambda_2|y_6, \ldots, y_{10}).$$

The target distribution is $f(\lambda_1, \lambda_2|y_1, \ldots, y_{20})$, thus the weights are

$$\begin{aligned} w(\lambda_1, \lambda_2) &= \frac{f(\lambda_1, \lambda_2 | y_1, \ldots, y_{20})}{g(\lambda_1, \lambda_2)} \qquad (3.6) \\ &\propto \frac{f(\lambda_1 | y_1, \ldots, y_5) f(\lambda_2 | y_6, \ldots, y_{10}) \prod_{i=11}^{20} f(y_i | \lambda_1, \lambda_2)}{f(\lambda_1 | y_1, \ldots, y_5) f(\lambda_2 | y_6, \ldots, y_{10})} \\ &= \prod_{i=11}^{20} f(y_i | \lambda_1, \lambda_2) = \prod_{i=11}^{20} \left(\frac{1}{2} \frac{\exp(-\lambda_1) \lambda_1^{y_i}}{y_i!} + \frac{1}{2} \frac{\exp(-\lambda_2) \lambda_2^{y_i}}{y_i!} \right) \end{aligned}$$

Thus we can draw a weighted sample of size n from the distribution of $f(\lambda_1, \lambda_2 | y_1, \ldots, y_{20})$ by repeating the three steps below n times:

1. Draw $\lambda_1 \sim \mathsf{Gamma}\left(\alpha + \sum_{i=1}^{5} y_i, \beta + 5\right)$
2. Draw $\lambda_2 \sim \mathsf{Gamma}\left(\alpha + \sum_{i=6}^{10} y_i, \beta + 5\right)$
3. Compute the weight $w(\lambda_1, \lambda_2)$ using equation (3.6).

From a simulation with $n = 50,000$ I obtained 4.4604 as posterior mean of λ_1 and 14.5294 as posterior mean of λ_2. The posterior densities are shown in figure 3.7. ◁

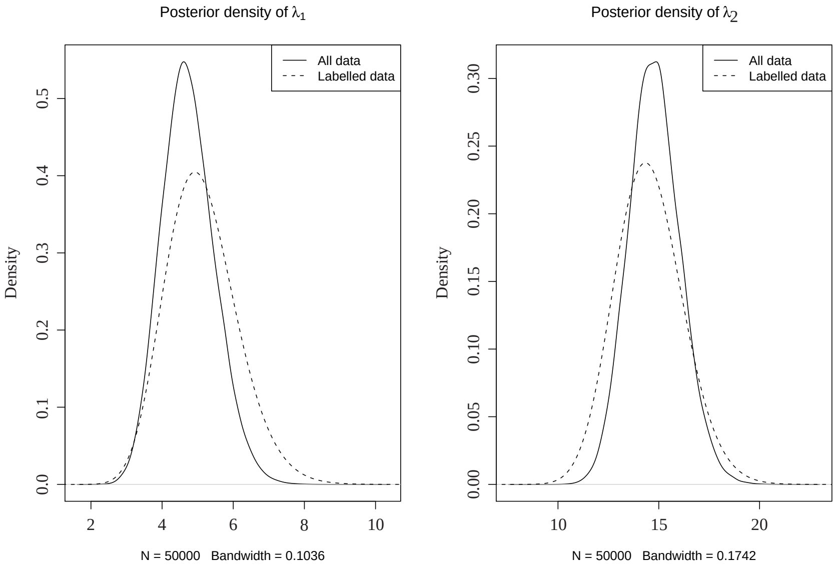


Figure 3.7. Posterior distributions of λ_1 and λ_2 in example 3.6. The dashed line is the posterior density obtained only from the labelled data.

Chapter 4

The Gibbs Sampler

4.1 Introduction

In section 3.3 we have seen that, using importance sampling, we can approximate an expectation $\mathbb{E}_f(h(X))$ without having to sample directly from f. However, finding an instrumental distribution which allows us to *efficiently* estimate $\mathbb{E}_f(h(X))$ can be difficult, especially in large dimensions.

In this chapter and the following chapters we will use a somewhat different approach. We will discuss methods that allow obtaining an *approximate* sample from f without having to sample from f directly. More mathematically speaking, we will discuss methods which generate a Markov chain whose stationary distribution is the distribution of interest f. Such methods are often referred to as Markov Chain Monte Carlo (MCMC) methods.

Example 4.1 (Poisson change point model). Assume the following Poisson model of two regimes for n random variables $Y_1, \ldots, Y_n$.[1]

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

A suitable (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

$$\begin{aligned} f(y_1, \ldots, y_n, \lambda_1, \lambda_2, M) &= \left(\prod_{i=1}^{M} \frac{\exp(-\lambda_1)\lambda_1^{y_i}}{y_i!} \right) \cdot \left(\prod_{i=M+1}^{n} \frac{\exp(-\lambda_2)\lambda_2^{y_i}}{y_i!} \right) \\ &\quad \cdot \frac{1}{\Gamma(\alpha_1)} \lambda_1^{\alpha_1 - 1} \beta_1^{\alpha_1} \exp(-\beta_1 \lambda_1) \cdot \frac{1}{\Gamma(\alpha_2)} \lambda_2^{\alpha_2 - 1} \beta_2^{\alpha_2} \exp(-\beta_2 \lambda_2). \end{aligned}$$

If M is known, the posterior distribution of λ_1 has the density

$$f(\lambda_1 | Y_1, \ldots, Y_n, M) \propto \lambda_1^{\alpha_1 - 1 + \sum_{i=1}^{M} y_i} \exp(-(\beta_1 + M)\lambda_1),$$

so

$$\lambda_1 | Y_1, \ldots Y_n, M \sim \mathsf{Gamma}\left(\alpha_1 + \sum_{i=1}^{M} y_i, \beta_1 + M \right) \tag{4.1}$$

$$\lambda_2 | Y_1, \ldots Y_n, M \sim \mathsf{Gamma}\left(\alpha_2 + \sum_{i=M+1}^{n} y_i, \beta_2 + n - M \right). \tag{4.2}$$

[1] The probability distribution function of the $\mathsf{Poi}(\lambda)$ distribution is $p(y) = \frac{\exp(-\lambda)\lambda^y}{y!}$.

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

$$p(M | Y_1, \ldots, Y_n, \lambda_1, \lambda_2) \propto \lambda_1^{\sum_{i=1}^{M} y_i} \cdot \lambda_2^{\sum_{i=M+1}^{n} y_i} \cdot \exp((\lambda_2 - \lambda_1) \cdot M) \tag{4.3}$$

The conditional distributions in (4.1) to (4.3) 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 ((4.1) to (4.3) in the example). This tentative strategy however raises some questions.

- Is the joint distribution uniquely specified by the conditional distributions?
- 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?

As we will see in sections 4.3 and 4.4, the answer to both questions is — under certain conditions — yes. The next section will however first of all state the Gibbs sampling algorithm.

4.2 Algorithm

The Gibbs sampler was first proposed by Geman and Geman (1984) and further developed by Gelfand and Smith (1990). Denote with $x_{-i} := (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_p)$.

Algorithm 4.1 ((Systematic sweep) Gibbs sampler). Starting with $(X_1^{(0)}, \ldots, X_p^{(0)})$ iterate for $t = 1, 2, \ldots$

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

Figure 4.1 illustrates the Gibbs sampler. The conditional distributions as used in the Gibbs sampler are often referred to as *full conditionals*. Note that the Gibbs sampler is *not* reversible. Liu et al. (1995) proposed the following algorithm that yields a reversible chain.

Algorithm 4.2 (Random sweep Gibbs sampler). Starting with $(X_1^{(0)}, \ldots, X_p^{(0)})$ iterate for $t = 1, 2, \ldots$

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

4.3 The Hammersley-Clifford Theorem

An interesting property of the full conditionals, which the Gibbs sampler is based on, is that they fully specify the joint distribution, as Hammersley and Clifford proved in 1970[2]. Note that the set of marginal distributions does *not* have this property.

[2] Hammersley and Clifford actually never published this result, as they could not extend the theorem to the case of non-positivity.

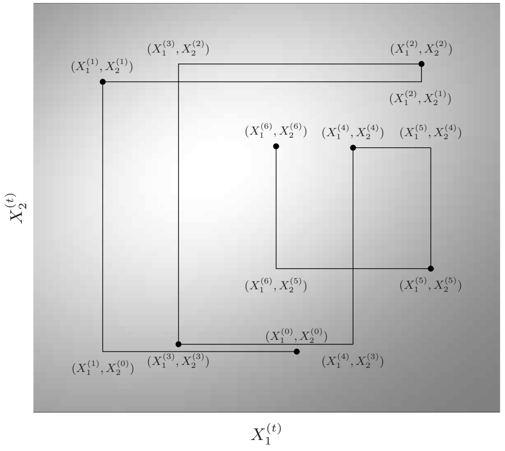


Figure 4.1. Illustration of the Gibbs sampler for a two-dimensional distribution

Definition 4.1 (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 $f_{X_i}(x_i) > 0$ for all $x_1, \ldots, x_p$ implies that $f(x_1, \ldots, x_p) > 0$.*

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 4.2 (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 $(\xi_1, \ldots, \xi_p) \in supp(f)$*

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

Proof. We have

$$f(x_1, \ldots, x_{p-1}, x_p) = f_{X_p|X_{-p}}(x_p|x_1, \ldots, x_{p-1}) f(x_1, \ldots, x_{p-1}) \tag{4.4}$$

and by complete analogy

$$f(x_1, \ldots, x_{p-1}, \xi_p) = f_{X_p|X_{-p}}(\xi_p|x_1, \ldots, x_{p-1}) f(x_1, \ldots, x_{p-1}), \tag{4.5}$$

thus

$$\begin{aligned} f(x_1, \ldots, x_p) &\overset{(4.4)}{=} \underbrace{f(x_1, \ldots, x_{p-1})}_{\overset{(4.5)}{=} f(x_1,\ldots,,x_{p-1},\xi_p)/f_{X_p|X_{-p}}(\xi_p|x_1,\ldots,x_{p-1})} f_{X_p|X_{-p}}(x_p|x_1, \ldots, x_{p-1}) \\ &= f(x_1, \ldots, x_{p-1}, \xi_p) \frac{f_{X_p|X_{-p}}(x_p|x_1, \ldots, x_{p-1})}{f_{X_p|X_{-p}}(\xi_p|x_1, \ldots, x_{p-1})} \\ &= \ldots \\ &= f(\xi_1, \ldots, \xi_p) \frac{f_{X_1|X_{-1}}(x_1|\xi_2, \ldots, \xi_p)}{f_{X_1|X_{-1}}(\xi_1|\xi_2, \ldots, \xi_p)} \cdots \frac{f_{X_p|X_{-p}}(x_p|x_1, \ldots, x_{p-1})}{f_{X_p|X_{-p}}(\xi_p|x_1, \ldots, x_{p-1})} \end{aligned}$$

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 modeling such problems mostly arise when using improper prior distributions.

Example 4.2. Consider the following "model"

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

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

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

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

4.4 Convergence of the Gibbs sampler

First of all we have to analyse whether the joint distribution $f(x_1, \ldots, x_p)$ is indeed the stationary distribution of the Markov chain generated by the Gibbs sampler[3]. For this we first have to determine the transition kernel corresponding to the Gibbs sampler.

Lemma 4.3. *The transition kernel of the Gibbs sampler is*

$$\begin{aligned} K(\mathbf{x}^{(t-1)}, \mathbf{x}^{(t)}) &= f_{X_1|X_{-1}}(x_1^{(t)}|x_2^{(t-1)}, \ldots, x_p^{(t-1)}) \cdot f_{X_2|X_{-2}}(x_2^{(t)}|x_1^{(t)}, x_3^{(t-1)}, \ldots, x_p^{(t-1)}) \cdot \ldots \\ &\quad \cdot f_{X_p|X_{-p}}(x_p^{(t)}|x_1^{(t)}, \ldots, x_{p-1}^{(t)}) \end{aligned}$$

Proof. We have

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

□

[3] All the results in this section will be derived for the systematic scan Gibbs sampler (algorithm 4.1). Very similar results hold for the random scan Gibbs sampler (algorithm 4.2).

Proposition 4.4. *The joint distribution $f(x_1, \ldots, x_p)$ is indeed the invariant distribution of the Markov chain $(\mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots)$ generated by the Gibbs sampler.*

Proof.

$$\begin{aligned}
& \int f(\mathbf{x}^{(t-1)}) K(\mathbf{x}^{(t-1)}, \mathbf{x}^{(t)})\, d\mathbf{x}^{(t-1)} \\
= & \int \cdots \underbrace{\underbrace{\int f(x_1^{(t-1)}, \ldots, x_p^{(t-1)})\, dx_1^{(t-1)}}_{=f(x_2^{(t-1)},\ldots,x_p^{(t-1)})} f_{X_1|X_{-1}}(x_1^{(t)}|x_2^{(t-1)}, \ldots, x_p^{(t-1)})}_{=f(x_1^{(t)},x_2^{(t-1)},\ldots,x_p^{(t-1)})} \cdots f_{X_p|X_{-p}}(x_p^{(t)}|x_1^{(t)}, \ldots, x_{p-1}^{(t)}) dx_2^{(t-1)} \ldots dx_p^{(t-1)} \\
= & \int \cdots \underbrace{\underbrace{\int f(x_1^{(t)}, x_2^{(t-1)}, \ldots, x_p^{(t-1)})\, dx_2^{(t-1)}}_{=f(x_1^{(t)},x_3^{(t-1)},\ldots,x_p^{(t-1)})} f_{X_2|X_{-2}}(x_2^{(t)}|x_1^{(t)}, x_3^{(t-1)}, \ldots, x_p^{(t-1)})}_{=f(x_1^{(t)},x_2^{(t)},x_3^{(t-1)},\ldots,x_p^{(t-1)})} \cdots f_{X_p|X_{-p}}(x_p^{(t)}|x_1^{(t)}, \ldots, x_{p-1}^{(t)}) dx_3^{(t-1)} \ldots dx_p^{(t-1)} \\
= & \ \ldots \\
= & \underbrace{\underbrace{\int f(x_1^{(t)}, \ldots, x_{p-1}^{(t)}, x_p^{(t-1)})\, dx_p^{(t-1)}}_{=f(x_1^{(t)},\ldots,x_{p-1}^{(t)})} f_{X_p|X_{-p}}(x_p^{(t)}|x_1^{(t)}, \ldots, x_{p-1}^{(t)})}_{=f(x_1^{(t)},\ldots,x_p^{(t)})} \\
= & \ f(x_1^{(t)}, \ldots, x_p^{(t)})
\end{aligned}$$

Thus according to definition 1.27 f is indeed the invariant distribution. □

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[4]. The following example shows that this does not need to be the case.

Example 4.3 (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 4.2 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_2 it will stay there and never reach C_1. The reason for this is that the conditional distribution $X_2|X_1$ $(X_1|X_2)$ is for $X_1 < 0$ $(X_2 < 0)$ entirely concentrated on C_2. ◁

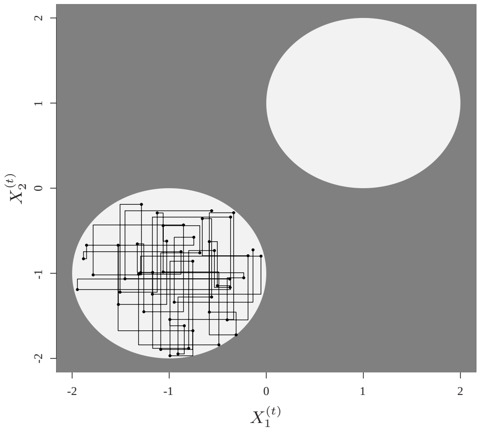


Figure 4.2. Illustration of a Gibbs sampler failing to sample from a distribution with unconnected support (uniform distribution on $\{(x_1, x_2) : \|(x_1, x_2) - (1, 1)\| \leq 1|\} \cup \{(x_1, x_2) : \|(x_1, x_2) - (-1, -1)\| \leq 1|\}$)

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

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

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

where the conditional densities are non-zero by the positivity condition. Thus the Markov Chain $(\mathbf{X}^{(t)})_t$ is strongly f-irreducible. As f is the unique invariant distribution of the Markov chain, it is as well recurrent (proposition 1.28). □

[4] Here and in the following we understand by “irreducibilty” irreducibility with respect to the target distribution f.

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 theorem 1.30.

Theorem 4.6. *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* $h : E \to \mathbb{R}$

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

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

Theorem 4.6 guarantees that we can approximate expectations $\mathbb{E}_f\left(h(\mathbf{X})\right)$ by their empirical counterparts using *a single* Markov chain.

Example 4.4. Assume that we want to use a Gibbs sampler to estimate the probability $\mathbb{P}(X_1 \geq 0, X_2 \geq 0)$ for a $\mathsf{N}_2\left(\begin{pmatrix}\mu_1\\ \mu_2\end{pmatrix}, \begin{pmatrix}\sigma_1^2 & \sigma_{12}\\ \sigma_{12} & \sigma_2^2\end{pmatrix}\right)$ distribution.[5] The marginal distributions are

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

In order to construct a Gibbs sampler, we need the conditional distributions $X_1|X_2 = x_2$ and $X_2|X_1 = x_1$. We have[6]

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

i.e.

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

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

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

[5] A Gibbs sampler is of course not the optimal way to sample from a $\mathsf{N}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ distribution. A more efficient way is: draw $Z_1, \ldots, Z_p \overset{\text{i.i.d.}}{\sim} N(0, 1)$ and set $(X_1, \ldots, X_p)' = \boldsymbol{\Sigma}^{1/2}(Z_1, \ldots, Z_p)' + \boldsymbol{\mu}$

[6] We make use of

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

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

Now consider the special case $\mu_1 = \mu_2 = 0$, $\sigma_1^2 = \sigma_2^2 = 1$ and $\sigma_{12} = 0.3$. Figure 4.4 shows the sample paths of this Gibbs sampler.
Using theorem 4.6 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 4.3 shows this estimate. ◁

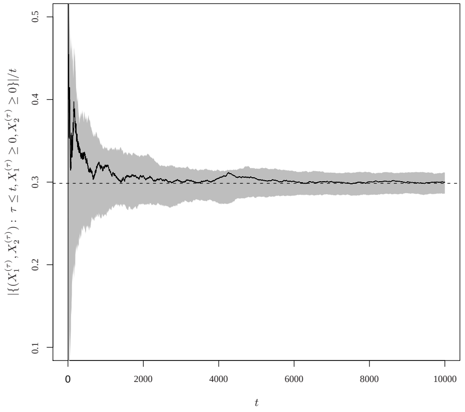
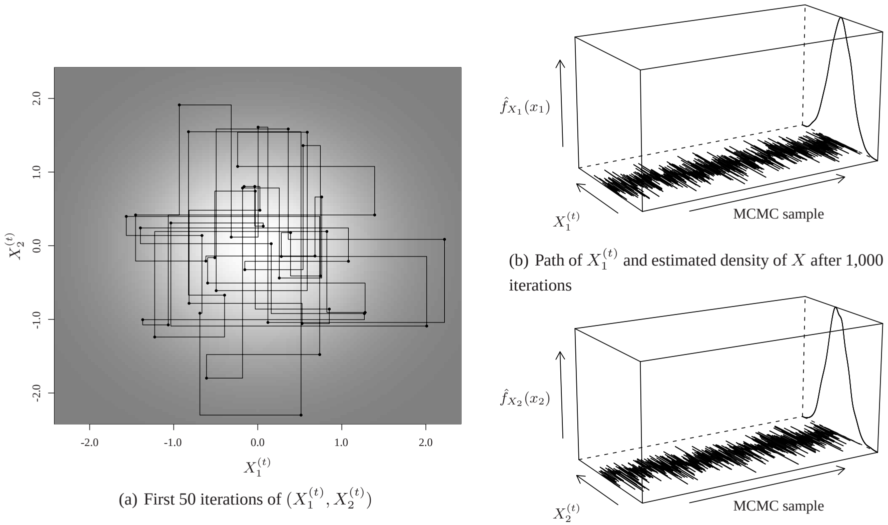


Figure 4.3. Estimate of the $\mathbb{P}(X_1 \geq 0, X_2 \geq 0)$ obtained using a Gibbs sampler. The area shaded in grey corresponds to the range of 100 replications.

Note that the realisations $(\mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots)$ form a Markov chain, and are thus *not* independent, but typically positively correlated. The correlation between the $\mathbf{X}^{(t)}$ 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 4.5 (Sampling from a highly correlated bivariate Gaussian). Figure 4.5 shows the results obtained when sampling from a bivariate Normal distribution as in example 4.4, 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 4.4 (and displayed in figure 4.4): 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 and thus yields to a slower convergence, as the plot of the estimated densities show (panels (b) and (c) of figures 4.4 and 4.5). ◁

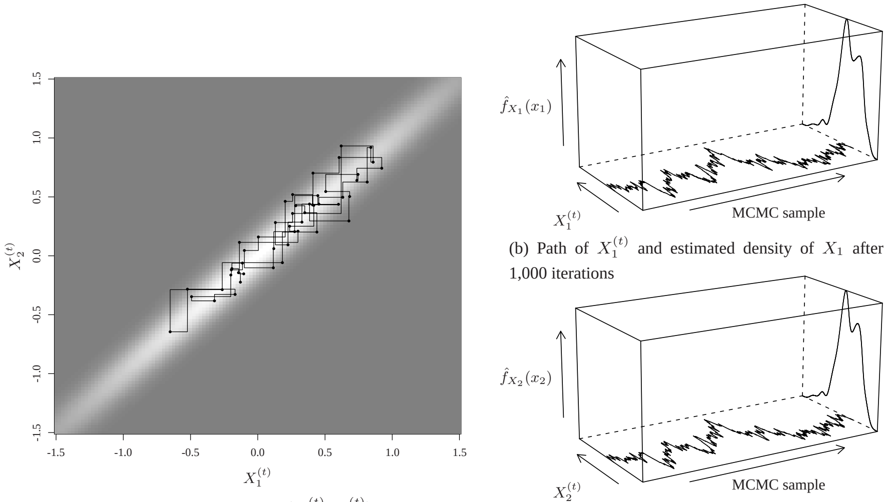


(a) First 50 iterations of $(X_1^{(t)}, X_2^{(t)})$

(b) Path of $X_1^{(t)}$ and estimated density of X after 1,000 iterations

(c) Path of $X_2^{(t)}$ and estimated density of X_2 after 1,000 iterations

Figure 4.4. Gibbs sampler for a bivariate standard normal distribution with correlation $\rho(X_1, X_2) = 0.3$.



(a) First 50 iterations of $(X_1^{(t)}, X_2^{(t)})$

(b) Path of $X_1^{(t)}$ and estimated density of X_1 after 1,000 iterations

(c) Path of $X_2^{(t)}$ and estimated density of X_2 after 1,000 iterations

Figure 4.5. Gibbs sampler for a bivariate normal distribution with correlation $\rho(X_1, X_2) = 0.99$.

Chapter 5

The Metropolis-Hastings Algorithm

5.1 Algorithm

In the previous chapter we have studied the Gibbs sampler, a special case of a Monte Carlo Markov Chain (MCMC) method: the target distribution is the invariant distribution of the Markov chain generated by the algorithm, to which it (hopefully) converges.

This chapter will introduce another MCMC method: the Metropolis-Hastings algorithm, which goes back to Metropolis et al. (1953) and Hastings (1970). Like the rejection sampling algorithm 3.1, the Metropolis-Hastings algorithm is based on proposing values sampled from an instrumental 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 3.1 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 5.1 (Metropolis-Hastings). Starting with $\mathbf{X}^{(0)} := (X_1^{(0)}, \ldots, X_p^{(0)})$ iterate for $t = 1, 2, \ldots$

1. Draw $\mathbf{X} \sim q(\cdot|\mathbf{X}^{(t-1)})$.
2. Compute
$$\alpha(\mathbf{X}|\mathbf{X}^{(t-1)}) = \min\left\{1, \frac{f(\mathbf{X}) \cdot q(\mathbf{X}^{(t-1)}|\mathbf{X})}{f(\mathbf{X}^{(t-1)}) \cdot q(\mathbf{X}|\mathbf{X}^{(t-1)})}\right\}. \tag{5.1}$$
3. With probability $\alpha(\mathbf{X}|\mathbf{X}^{(t-1)})$ set $\mathbf{X}^{(t)} = \mathbf{X}$, otherwise set $\mathbf{X}^{(t)} = \mathbf{X}^{(t-1)}$.

Figure 5.1 illustrates the Metropolis-Hasting algorithm. Note that if the algorithm rejects the newly proposed value (open disks joined by dotted lines in figure 5.1) it stays at its current value $\mathbf{X}^{(t-1)}$. The probability that the Metropolis-Hastings algorithm accepts the newly proposed state $\mathbf{X}$ given that it currently is in state $\mathbf{X}^{(t-1)}$ is
$$a(\mathbf{x}^{(t-1)}) = \int \alpha(\mathbf{x}|\mathbf{x}^{(t-1)}) q(\mathbf{x}|\mathbf{x}^{(t-1)})\, d\mathbf{x}. \tag{5.2}$$
Just like the Gibbs sampler, the Metropolis-Hastings algorithm generates a Markov chain, whose properties will be discussed in the next section.

Remark 5.1. The probability of acceptance (5.1) does not depend on the normalisation constant, i.e. if $f(\mathbf{x}) = C \cdot \pi(\mathbf{x})$, then
$$\frac{f(\mathbf{x}) \cdot q(\mathbf{x}^{(t-1)}|\mathbf{x})}{f(\mathbf{x}^{(t-1)}) \cdot q(\mathbf{x}|\mathbf{x}^{(t-1)})} = \frac{C\pi(\mathbf{x}) \cdot q(\mathbf{x}^{(t-1)}|\mathbf{x})}{C\pi(\mathbf{x}^{(t-1)}) \cdot q(\mathbf{x}|\mathbf{x}^{(t-1)})} = \frac{\pi(\mathbf{x}) \cdot q(\mathbf{x}^{(t-1)}|\mathbf{x})}{\pi(\mathbf{x}^{(t-1)}) \cdot q(\mathbf{x}|\mathbf{x}^{(t-1)})}$$

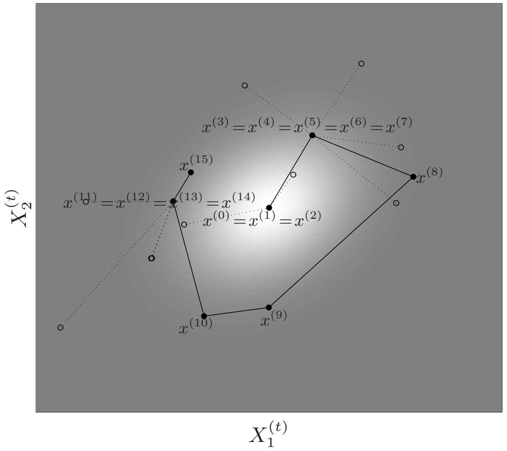


Figure 5.1. Illustration of the Metropolis-Hastings algorithm. Filled dots denote accepted states, open circles rejected values.

Thus f only needs to be known up to normalisation constant.[1]

5.2 Convergence results

Lemma 5.2. *The transition kernel of the Metropolis-Hastings algorithm is*
$$K(\mathbf{x}^{(t-1)}, \mathbf{x}^{(t)}) = \alpha(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}) q(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}) + (1 - a(\mathbf{x}^{(t-1)}))\delta_{\mathbf{x}^{(t-1)}}(\mathbf{x}^{(t)}), \tag{5.3}$$
where $\delta_{\mathbf{x}^{(t-1)}}(\cdot)$ *denotes Dirac-mass on* $\{\mathbf{x}^{(t-1)}\}$.

Note that the transition kernel (5.3) is *not* continuous with respect to the Lebesgue measure.

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

Proposition 5.3. *The Metropolis-Hastings kernel (5.3) satisfies the detailed balance condition*
$$K(\mathbf{x}^{(t-1)}, \mathbf{x}^{(t)}) f(\mathbf{x}^{(t-1)}) = K(\mathbf{x}^{(t)}, \mathbf{x}^{(t-1)}) f(\mathbf{x}^{(t)})$$
and thus $f(\mathbf{x})$ *is the invariant distribution of the Markov chain* $(\mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots)$ *generated by the Metropolis-Hastings sampler. Furthermore the Markov chain is reversible.*

[1] On a similar note, it is enough to know $q(\mathbf{x}^{(t-1)}|\mathbf{x})$ up to a multiplicative constant independent of $\mathbf{x}^{(t-1)}$ and $\mathbf{x}$.

Proof. We have that

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

and thus

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

The other conclusions follow by theorem 1.22, which also applies in the continuous case (see page 21). □

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

Example 5.1 (Reducible Metropolis-Hastings). Consider using a Metropolis-Hastings algorithm for sampling from a uniform distribution on $[0, 1] \cup [2, 3]$ and a $\mathsf{U}(x^{(t-1)} - \delta, x^{(t-1)} + \delta)$ distribution as proposal distribution $q(\cdot|x^{(t-1)})$. Figure 5.2 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 $[2, 3]$. ◁

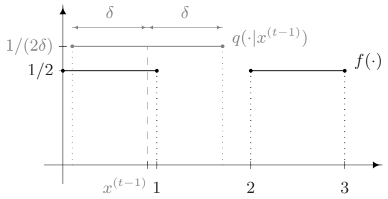


Figure 5.2. Illustration of example 5.1

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

- If $q(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)})$ is positive for all $\mathbf{x}^{(t-1)}, \mathbf{x}^{(t)} \in \operatorname{supp}(f)$, 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 $q(\cdot|\mathbf{x}^{(t-1)})$ like multivariate Gaussians or t-distributions fulfil this condition.
- Roberts and Tweedie (1996) give a more general condition for the irreducibility of the resulting Markov chain: they only require that
$$\forall\, \epsilon\ \exists\, \delta:\ q(\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}) > \epsilon \text{ if } \|\mathbf{x}^{(t-1)} - \mathbf{x}^{(t)}\| < \delta$$
together with the boundedness of f on any compact subset of its support.

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

$$\mathbb{P}\left(f(\mathbf{X}^{(t-1)})q(\mathbf{X}|\mathbf{X}^{(t-1)}) > f(\mathbf{X})q(\mathbf{X}^{(t-1)}|\mathbf{X})\right) > 0.$$

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.

Proposition 5.4. *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 unique invariant distribution (using proposition 1.28). For a proof of Harris recurrence see (Tierney, 1994). □

As we have now established (Harris-)recurrence, we are now ready to state an ergodic theorem (using theorem 1.30).

Theorem 5.5. *If the Markov chain generated by the Metropolis-Hastings algorithm is irreducible, then for any integrable function $h : E \to \mathbb{R}$*

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

for every starting value $\mathbf{X}^{(0)}$.

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

5.3 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 $\mathbf{X}$ not using the fairly general

$$\mathbf{X} \sim q(\cdot|\mathbf{X}^{(t-1)}), \tag{5.4}$$

from algorithm 5.1, but rather

$$\mathbf{X} = \mathbf{X}^{(t-1)} + \boldsymbol{\varepsilon}, \qquad \boldsymbol{\varepsilon} \sim g, \tag{5.5}$$

with g being a *symmetric* distribution. It is easy to see that (5.5) is a special case of (5.4) using $q(\mathbf{x}|\mathbf{x}^{(t-1)}) = g(\mathbf{x} - \mathbf{x}^{(t-1)})$. When using (5.5) the probability of acceptance simplifies to

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

as $q(\mathbf{X}|\mathbf{X}^{(t-1)}) = g(\mathbf{X} - \mathbf{X}^{(t-1)}) = g(\mathbf{X}^{(t-1)} - \mathbf{X}) = q(\mathbf{X}^{(t-1)}|\mathbf{X})$ using the symmetry of g. This yields the following algorithm which is a special case of algorithm 5.1, which is actually the original algorithm proposed by Metropolis et al. (1953).

Algorithm 5.2 (Random walk Metropolis). Starting with $\mathbf{X}^{(0)} := (X_1^{(0)}, \ldots, X_p^{(0)})$ and using a symmetric distributon g, iterate for $t = 1, 2, \ldots$

1. Draw $\boldsymbol{\varepsilon} \sim g$ and set $\mathbf{X} = \mathbf{X}^{(t-1)} + \boldsymbol{\varepsilon}$.
2. Compute
$$\alpha(\mathbf{X}|\mathbf{X}^{(t-1)}) = \min\left\{1, \frac{f(\mathbf{X})}{f(\mathbf{X}^{(t-1)})}\right\}. \tag{5.6}$$
3. With probability $\alpha(\mathbf{X}|\mathbf{X}^{(t-1)})$ set $\mathbf{X}^{(t)} = \mathbf{X}$, otherwise set $\mathbf{X}^{(t)} = \mathbf{X}^{(t-1)}$.

Example 5.2 (Bayesian probit model). In a medical study on infections resulting from birth by Cesarean section (taken from Fahrmeir and Tutz, 2001) three influence factors have been studied: an indicator whether the Cesarian 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 5.1.

Number of births		planned	risk factors	antibiotics
with infection	total			
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

Table 5.1. Data used in example 5.2

The data can be modeled by assuming that

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

where $\mathbf{z}_i = (1, z_{i1}, z_{i2}, z_{i3})$ 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

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

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 $\boldsymbol{\varepsilon} \sim N(\mathbf{0}, \boldsymbol{\Sigma})$ and set $\boldsymbol{\beta} = \boldsymbol{\beta}^{(t-1)} + \boldsymbol{\varepsilon}$.
2. Compute
$$\alpha(\boldsymbol{\beta}|\boldsymbol{\beta}^{(t-1)}) = \min\left\{1, \frac{f(\boldsymbol{\beta}|Y_1,\ldots,Y_n)}{f(\boldsymbol{\beta}^{(t-1)}|Y_1,\ldots,Y_n)}\right\}.$$
3. With probability $\alpha(\boldsymbol{\beta}|\boldsymbol{\beta}^{(t-1)})$ set $\boldsymbol{\beta}^{(t)} = \boldsymbol{\beta}$, otherwise set $\boldsymbol{\beta}^{(t)} = \boldsymbol{\beta}^{(t-1)}$.

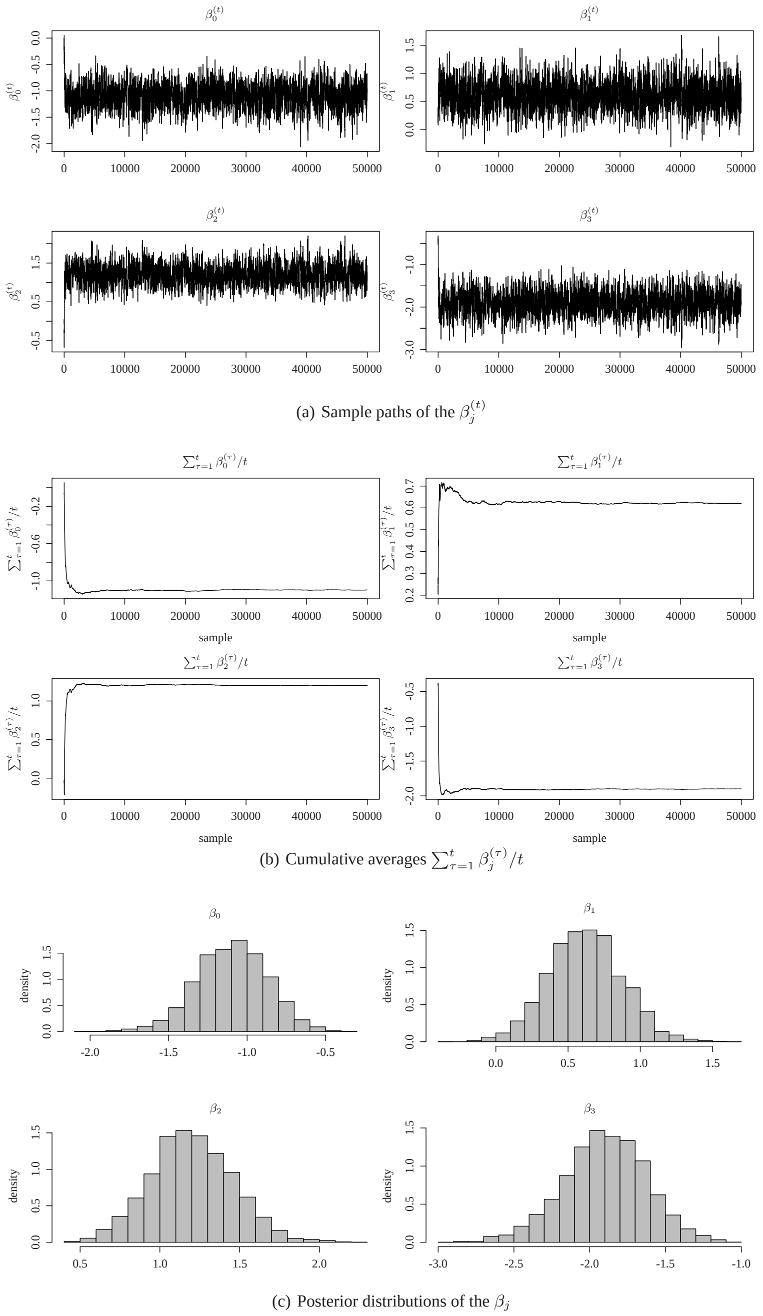
To keep things simple, we choose the covariance $\boldsymbol{\Sigma}$ of the proposal to be $0.08 \cdot \mathbb{I}$.

Figure 5.3 and table 5.2 show the results obtained using 50,000 samples[2]. 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 5.3 and table 5.2 the first 10,000 samples have been discarded ("burn-in"). ◁

		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

Table 5.2. Parameter estimates obtained for the Bayesian probit model from example 5.2

[2] You might want to consider a longer chain in practise.



(a) Sample paths of the $\beta_j^{(t)}$

(b) Cumulative averages $\sum_{\tau=1}^{t}\beta_j^{(\tau)}/t$

(c) Posterior distributions of the β_j

Figure 5.3. Results obtained for the Bayesian probit model from example 5.2

5.4 Choosing the proposal distribution

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

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

Thus we would ideally want a proposal distribution that both allows for fast changes in the $\mathbf{X}^{(t)}$ 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 $X^{(t)}$ change only very slowly. If, on the other hand, we choose a proposal distribution with a large variance, the $X^{(t)}$ can potentially move very fast, however the probability of acceptance will be rather low.

Example 5.3. 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 5.4 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 5.3 shows the average correlation $\rho(X^{(t-1)}, X^{(t)})$ as well as the average probability of acceptance $\alpha(X|X^{(t-1)})$ 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 et al. (1995). ◁

	Autocorrelation $\rho(X^{(t-1)}, X^{(t)})$		Probability of acceptance $\alpha(X, X^{(t-1)})$	
	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)

Table 5.3. Average correlation $\rho(X^{(t-1)}, X^{(t)})$ and average probability of acceptance $\alpha(X|X^{(t-1)})$ found in example 5.3 for different choices of the proposal variance σ^2.

Finding the ideal proposal distribution $q(\cdot|\mathbf{x}^{(t-1)})$ is an art.[3] This is the price we have to pay for the generality of the Metropolis-Hastings algorithm. Popular choices for random walk proposals are multivariate Gaussians or t-distributions. The latter have heavier tails, making them a safer choice. The covariance structure of the proposal distribution should ideally reflect the expected covariance of the $(X_1, \ldots, X_p)$. Gelman et al. (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. Note however that these are just rough guidelines.

Example 5.4 (Bayesian probit model (continued)). In the Bayesian probit model we studied in example 5.2 we drew

[3] 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!

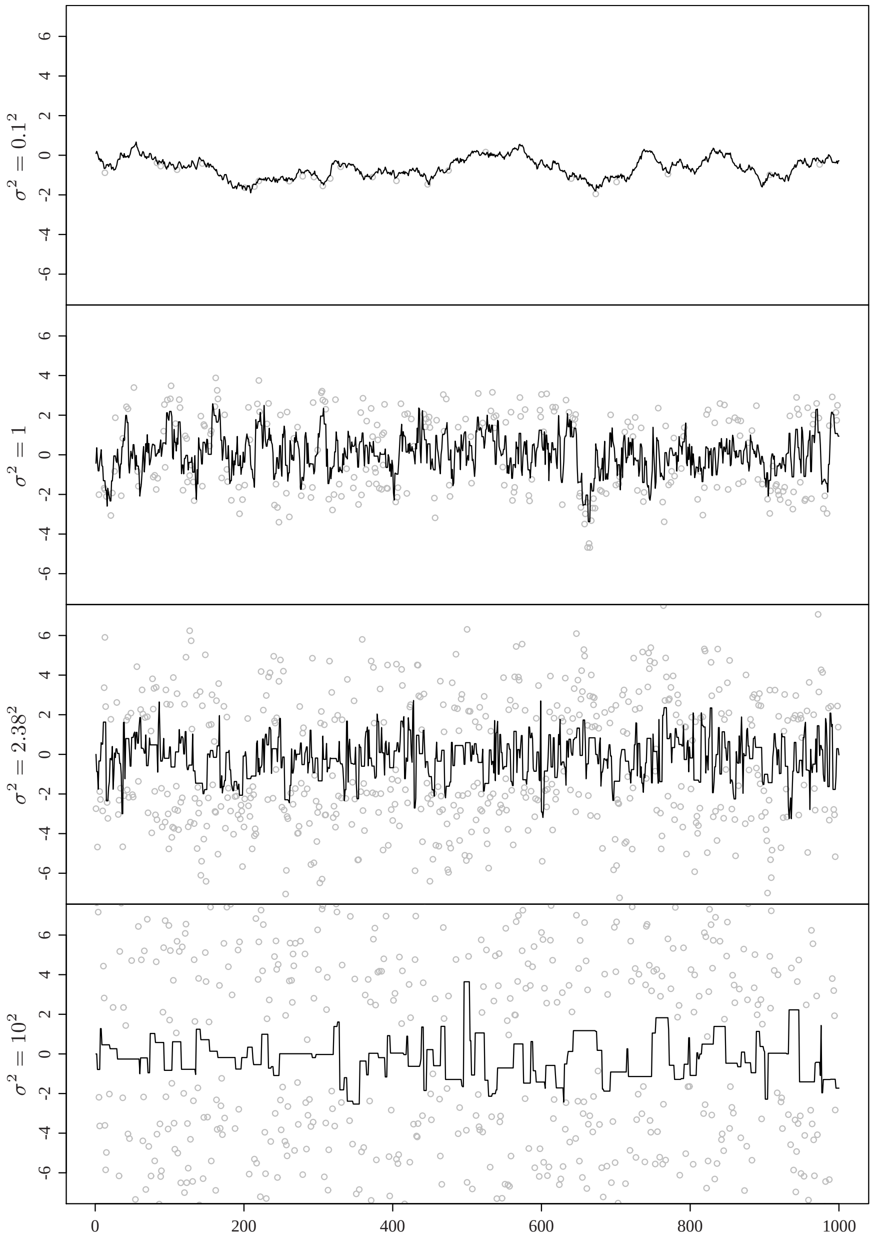


Figure 5.4. Sample paths for example 5.3 for different choices of the proposal variance σ^2. Open grey discs represent rejected values.

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

with $\boldsymbol{\Sigma} = 0.08 \cdot \mathbf{I}$, i.e. we modeled the components of ε to be independent. The proportion of accepted values we obtained in example 5.2 was 13.9%. Table 5.4 (a) 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 $(\mathbf{Z}'\mathbf{D}\mathbf{Z})^{-1}$, where $\mathbf{Z}$ is the matrix of the covariates, and $\mathbf{D}$ is a suitable diagonal matrix. When using $\boldsymbol{\Sigma} = 2 \cdot (\mathbf{Z}'\mathbf{D}\mathbf{Z})^{-1}$ in the algorithm presented in section 5.2 we can obtain better mixing performance: the autocorrelation is reduced (see table 5.4 (b)), 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 entirely due to a difference in the structure of the the covariance. ◁

(a) $\boldsymbol{\Sigma} = 0.08 \cdot \mathbf{I}$

	β_0	β_1	β_2	β_3
Autocorrelation $\rho(\beta_j^{(t-1)}, \beta_j^{(t)})$	0.9496	0.9503	0.9562	0.9532

(b) $\boldsymbol{\Sigma} = 2 \cdot (\mathbf{Z}'\mathbf{D}\mathbf{Z})^{-1}$

	β_0	β_1	β_2	β_3
Autocorrelation $\rho(\beta_j^{(t-1)}, \beta_j^{(t)})$	0.8726	0.8765	0.8741	0.8792

Table 5.4. Autocorrelation $\rho(\beta_j^{(t-1)}, \beta_j^{(t)})$ between subsequent samples for the two choices of the covariance $\boldsymbol{\Sigma}$.

Chapter 6

Diagnosing convergence

6.1 Practical considerations

The theory of Markov chains we have seen in chapter 1 guarantees that a Markov chain that is irreducible and has invariant distribution f converges to the invariant distribution. The ergodic theorems 4.6 and 5.5 allow for approximating expectations $\mathbb{E}_f(h(\mathbf{X}))$ by their the corresponding means

$$\frac{1}{T}\sum_{t=1}^{T} h(\mathbf{X}^{(t)}) \longrightarrow \mathbb{E}_f(h(\mathbf{X}))$$

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

Burn-in Depending on how $\mathbf{X}^{(0)}$ is chosen, the distribution of $(\mathbf{X}^{(t)})_t$ for small t might still be far from the stationary distribution f. Thus it might be beneficial to discard the first iterations $\mathbf{X}^{(t)}$, $t = 1, \ldots, T_0$. 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 $(\mathbf{X}^{(t)})_t$ is. Figure 6.1 illustrates the idea of a burn-in period.

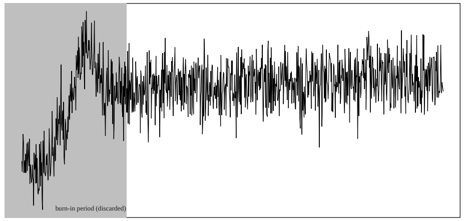


Figure 6.1. Illustration of the idea of a burn-in period.

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 $(\mathbf{Y}^{(t)})_t$ with $\mathbf{Y}^{(t)} = \mathbf{X}^{(m\cdot t)}$ instead of $(\mathbf{X}^{(t)})_t$. If the correlation $\rho(\mathbf{X}^{(t)}, \mathbf{X}^{(t+\tau)})$ decreases monotonically in τ, then

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

i.e. the thinned chain $(\mathbf{Y}^{(t)})_t$ exhibits less autocorrelation than the original chain $(\mathbf{X}^{(t)})_t$. Thus thinning can be seen as a technique for reducing the autocorrelation, however at the price of yielding a chain $(\mathbf{Y}^{(t)})_{t=1,\ldots\lfloor T/m\rfloor}$, whose length is reduced to $(1/m)$-th of the length of the original chain $(\mathbf{X}^{(t)})_{t=1,\ldots,T}$. Even though thinning is very popular, it cannot be justified when the objective is estimating $\mathbb{E}_f(h(\mathbf{X}))$, as the following lemma shows.

Lemma 6.1. *Let* $(\mathbf{X}^{(t)})_{t=1,\ldots,T}$ *be a sequence of random variables (e.g. from a Markov chain) with* $\mathbf{X}^{(t)} \sim f$ *and* $(\mathbf{Y}^{(t)})_{t=1,\ldots,\lfloor T/m\rfloor}$ *a second sequence defined by* $\mathbf{Y}^{(t)} := \mathbf{X}^{(m\cdot t)}$. *If* $\mathrm{Var}_f(h(\mathbf{X}^{(t)})) < +\infty$, *then*

$$\mathrm{Var}\left(\frac{1}{T}\sum_{t=1}^{T} h(\mathbf{X}^{(t)})\right) \leq \mathrm{Var}\left(\frac{1}{\lfloor T/m\rfloor}\sum_{t=1}^{\lfloor T/m\rfloor} h(\mathbf{Y}^{(t)})\right).$$

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

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

and

$$\mathrm{Var}\left(\sum_{t=1}^{T/m} h(\mathbf{X}^{(t\cdot m+\tau_1)})\right) = \mathrm{Var}\left(\sum_{t=1}^{T/m} h(\mathbf{X}^{(t\cdot m+\tau_2)})\right)$$

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

$$\begin{aligned}
\mathrm{Var}\left(\sum_{t=1}^{T} h(\mathbf{X}^{(t)})\right) &= \mathrm{Var}\left(\sum_{\tau=0}^{m-1}\sum_{t=1}^{T/m} h(\mathbf{X}^{(t\cdot m+\tau)})\right)\\
&= m\cdot\mathrm{Var}\left(\sum_{t=1}^{T/m} h(\mathbf{X}^{(t\cdot m)})\right) + \sum_{\eta\neq\tau=0}^{m-1} \underbrace{\mathrm{Cov}\left(\sum_{t=1}^{T/m} h(\mathbf{X}^{(t\cdot m+\eta)}), \sum_{t=1}^{T/m} h(\mathbf{X}^{(t\cdot m+\tau)})\right)}_{\leq \mathrm{Var}\left(\sum_{t=1}^{T/m} h(\mathbf{X}^{(t\cdot m)})\right)}\\
&\leq m^2\cdot\mathrm{Var}\left(\sum_{t=1}^{T/m} h(\mathbf{X}^{(t\cdot m)})\right) = m^2\cdot\mathrm{Var}\left(\sum_{t=1}^{T/m} h(\mathbf{Y}^{(t)})\right).
\end{aligned}$$

Thus

$$\mathrm{Var}\left(\frac{1}{T}\sum_{t=1}^{T} h(\mathbf{X}^{(t)})\right) = \frac{1}{T^2}\mathrm{Var}\left(\sum_{t=1}^{T} h(\mathbf{X}^{(t)})\right) \leq \frac{m^2}{T^2}\mathrm{Var}\left(\sum_{t=1}^{T/m} h(\mathbf{Y}^{(t)})\right) = \mathrm{Var}\left(\frac{1}{T/m}\sum_{t=1}^{T/m} h(\mathbf{Y}^{(t)})\right).$$

□

The concept of thinning can be useful for other reasons. If the computer's memory cannot hold the entire chain $(\mathbf{X}^{(t)})_t$, thinning is a good choice. Further, it can be easier to assess the convergence of the thinned chain $(\mathbf{Y}^{(t)})_t$ as opposed to entire chain $(\mathbf{X}^{(t)})_t$.

6.2 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 practise.

This section tries to give a brief overview over various approaches to diagnosing convergence. A more detailed review with many practical examples can be diagnofound in (Guihennec-Jouyaux et al., 1998) or (Robert and Casella, 2004, chapter 12). There is an R package (`CODA`) that provides a vast selection of tools for diagnosing convergence. Diagnosing convergence is an art. The techniques presented in the following are nothing other than exploratory tools that help you judging 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 tasks of diagnosing different aspects of convergence:

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

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

Convergence of the averages. Does $\sum_{t=1}^{T} h(\mathbf{X}^{(t)})/T$ provide a good approximation to the expectation $\mathbb{E}_f(h(\mathbf{X}))$ 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?

6.2.1 Basic plots

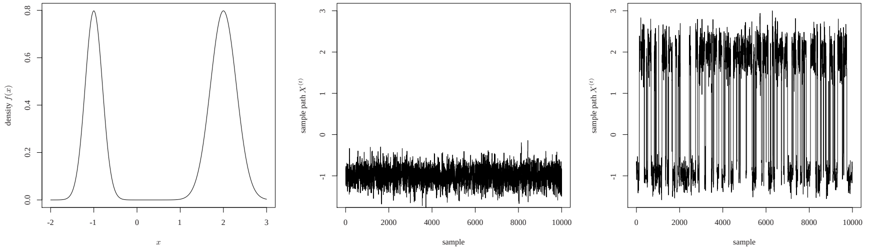
The most basic approach to diagnosing the output of a Markov Chain Monte Carlo algorithm is to plot the sample path $(\mathbf{X}^{(t)})_t$ as in figures 4.4 (b) (c), 4.5 (b) (c), 5.3 (a), and 5.4. Note that the convergence of $(\mathbf{X}^{(t)})_t$ is in distribution, i.e. the sample path is *not* supposed to converge to a single value. Ideally, the plot should be oscillating very fast and show very little structure or trend (like for example figure 4.4). The smoother the plot seems (like for example figure 4.5), the slower mixing the resulting chain is.

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.

Example 6.1 (A simple mixture of two Gaussians). In this example we sample 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 6.2 (a) for a plot of the density) using a random walk Metropolis algorithm with proposed value $X = X^{(t-1)} + \varepsilon$ with $\varepsilon \sim \mathsf{N}(0, \mathrm{Var}(\varepsilon))$. If we choose the proposal variance $\mathrm{Var}(\varepsilon)$ too small, we only sample from one population instead of both. Figure 6.2 shows the sample paths for two choices of $\mathrm{Var}(\varepsilon)$: $\mathrm{Var}(\varepsilon) = 0.4^2$ and $\mathrm{Var}(\varepsilon) = 1.2^2$. The first choice of $\mathrm{Var}(\varepsilon)$ 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 6.2 (b) alone that the chain has not explored the entire support of the target. ◁



(a) Density $f(x)$

(b) Sample path of a random walk Metropolis algorithm with proposal variance $\mathrm{Var}(\varepsilon) = 0.4^2$

(c) Sample path of a random walk Metropolis algorithm with proposal variance $\mathrm{Var}(\varepsilon) = 1.2^2$

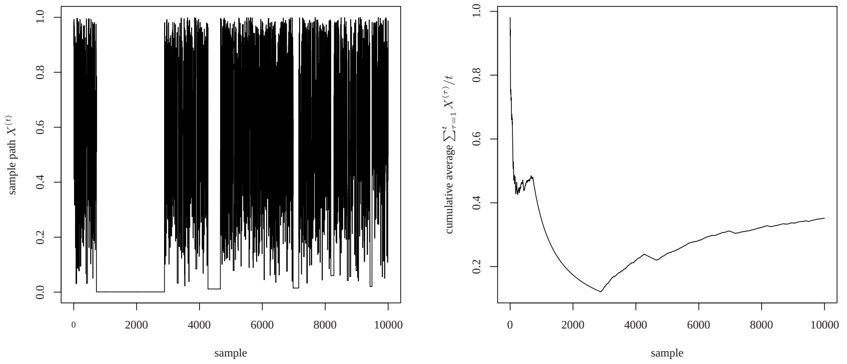
Figure 6.2. Density of the mixture distribution with two random walk Metropolis samples using two different variances $\mathrm{Var}(\varepsilon)$ of the proposal.

In order to diagnose the convergence of the 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 ($\mathbb{E}_f(h(\mathbf{X}))$. Figures 4.3, and 5.3 (b) show plots of the cumulative averages. An alternative to plotting the cumulative means is using the so-called CUSUMs $\left(\bar{h}(X_j) - \sum_{\tau=1}^{t} h(X_j^{(\tau)})/t\right)_t$ with $\bar{h}(X_j) = \sum_{\tau=1}^{T} h(X_j^{(\tau)})/T$, which is nothing other than the difference between the cumulative averages and the estimate of the limit $\mathbb{E}_f(h(\mathbf{X}))$.

Example 6.2 (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 $\mathsf{Beta}(\alpha, 1)$ 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 $X^{(t)} \sim \mathsf{Beta}(\alpha + 1, 1)$.

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 6.3 (b)). ◁



(a) Sample path $X^{(t)}$

(b) Cumulative means $\sum_{\tau=1}^{t} X^{(\tau)}/t$

Figure 6.3. Sample paths and cumulative means obtained for the pathological Beta generator.

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.

6.2.2 Non-parametric tests of stationarity

This section presents the Kolmogorov-Smirnov test, which is an example of how nonparametric tests can be used as a tool for diagnosing whether a Markov chain has already converged.

In its simplest version, it is based on splitting the chain into three parts: $(\mathbf{X}^{(t)})_{t=1,\ldots,\lfloor T/3 \rfloor}$, $(\mathbf{X}^{(t)})_{t=\lfloor T/3 \rfloor+1,\ldots,2\lfloor T/3 \rfloor}$, and $(\mathbf{X}^{(t)})_{t=2\lfloor T/3 \rfloor+1,\ldots,T}$. 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 we should be able to tell whether the chain has converged by comparing the distribution of $(\mathbf{X}^{(t)})_{t=\lfloor T/3 \rfloor+1,\ldots,2\lfloor T/3 \rfloor}$ to the one of $(\mathbf{X}^{(t)})_{t=2\lfloor T/3 \rfloor+1,\ldots,T}$ using suitable nonparametric two-sample tests. One such test is the Kolmogorov-Smirnov test.

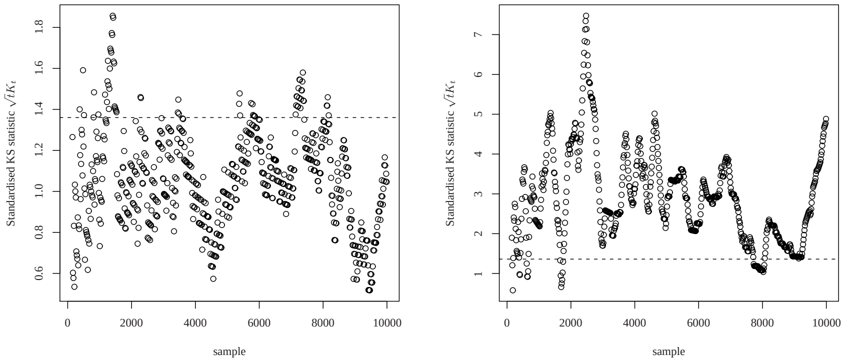
As the Kolmogorov-Smirnov test is designed for i.i.d. samples, we do not apply it to the $(\mathbf{X}^{(t)})_t$ directly, but to a thinned chain $(\mathbf{Y}^{(t)})_t$ with $\mathbf{Y}^{(t)} = \mathbf{X}^{(m \cdot t)}$: the thinned chain is less correlated and thus closer to being an i.i.d. sample. We can now compare the distribution of $(\mathbf{Y}^{(t)})_{t=\lfloor T/(3m) \rfloor+1,\ldots,2\lfloor T/(3m) \rfloor}$ to the one of

$(\mathbf{Y}^{(t)})_{t=2\lfloor T/(3m)\rfloor+1,\ldots,\lfloor T/m\rfloor}$ using the Kolmogorov-Smirnov statistic [1]

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

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.[2] As long as a significant proportion of the values of the 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 6.3 (Gibbs sampling from a bivariate Gaussian (continued)). In this example we consider sampling from a bivariate Gaussian distribution, once with $\rho(X_1, X_2) = 0.3$ (as in example 4.4) and once with $\rho(X_1, X_2) = 0.99$ (as in example 4.5). The former leads a fast mixing chain, the latter a very slowly mixing chain. Figure 6.4 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. ◁



(a) $\rho(X_1, X_2) = 0.3$

(b) $\rho(X_1, X_2) = 0.99$

Figure 6.4. Standardised Kolmogorov-Smirnov statistic for $X_1^{(5\cdot t)}$ from the Gibbs sampler from the bivariate Gaussian for two different correlations.

Note that the Kolmogorov-Smirnov test suffers from the "you've only seen where you've been" problem, as it is based on comparing $(\mathbf{Y}^{(t)})_{t=\lfloor T/(3m)\rfloor+1,\ldots,2\lfloor T/(3m)\rfloor}$ and $(\mathbf{Y}^{(t)})_{t=2\lfloor T/(3m)\rfloor+1,\ldots,\lfloor T/m\rfloor}$. A plot of the Kolmogorov-Smirnov statistic for the chain with $\text{Var}(\varepsilon) = 0.4$ from example 6.1 would not reveal anything unusual.

[1] 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

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

[2] K_t is hereby the Kolmogorov-Smirnov statistic obtained from the sample consisting of the first t observations only.

6.2.3 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 by the Riemann sum

$$\sum_{t=2}^{T} (X^{[t]} - X^{[t-1]}) f(X^{[t]}), \tag{6.1}$$

where $X^{[1]} \leq \ldots \leq X^{[T]}$ is the ordered sample from the Markov chain. If the Markov chain has explored all the support of f, then (6.1) should be around 1. 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.

Example 6.4 (A simple mixture of two Gaussians (continued)). In example 6.1 we considered two random-walk Metropolis algorithms: one ($\text{Var}(\varepsilon) = 0.4^2$) failed to explore the entire support of the target distribution, whereas the other one ($\text{Var}(\varepsilon) = 1.2^2$) 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 comparing several ways of estimating the same quantity. As long as 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 (6.1).

6.2.4 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 $(\mathbf{X}^{(1,t)})_t, \ldots, (\mathbf{X}^{(L,t)})_t$ — with overdispersed[3] starting values $\mathbf{X}^{(1,0)}, \ldots, \mathbf{X}^{(L,0)}$, covering at least the support of the target distribution.

All L chains should converge to the same distribution, so comparing the plots from section 6.2.1 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 chain and averaging over these results, i.e. our estimate is $\sum_{l=1}^{L} \delta_\alpha^{(l)}/L$, where $\delta_\alpha^{(l)}$ is the distance between the α and $(1-\alpha)$ quantile of the l-th chain$(X^{(l,t)})_t$. Alternatively, we can pool the data first, and then compute the distance between the α and $(1-\alpha)$ 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

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

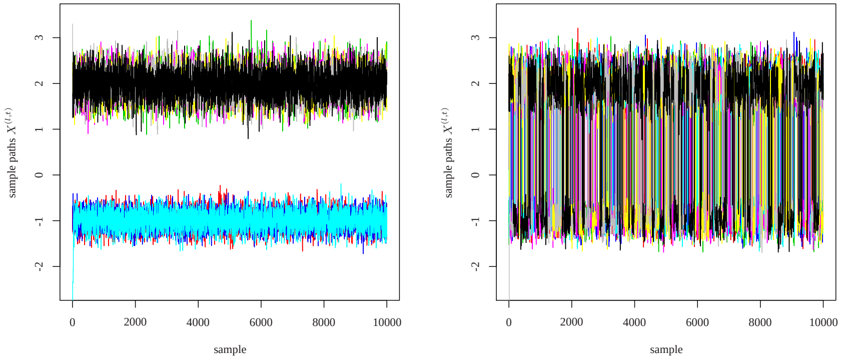
can be used as a tool to diagnose whether all chains sampled from the same distribution, in which case the ratio should be around 1.

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).

[3] i.e. the variance of the starting values should be larger than the variance of the target distribution.

Example 6.5 (A simple mixture of two Gaussians (continued)). In the example of the mixture of two Gaussians we will consider $L = 8$ chains initialised from a $\mathsf{N}(0, 10^2)$ distribution. Figure 6.5 shows the sample paths of the 8 chains for both choices of $\text{Var}(\varepsilon)$. The corresponding values of $\hat{S}^{\text{interval}}_{0.05}$ are:

$$\text{Var}(\varepsilon) = 0.4^2: \quad \hat{S}^{\text{interval}}_{0.05} = \frac{0.9789992}{3.630008} = 0.2696962$$
$$\text{Var}(\varepsilon) = 1.2^2: \quad \hat{S}^{\text{interval}}_{0.05} = \frac{3.634382}{3.646463} = 0.996687. \qquad \triangleleft$$



(a) $\text{Var}(\varepsilon) = 0.4^2$

(b) $\text{Var}(\varepsilon) = 1.2^2$

Figure 6.5. Comparison of the sample paths for $L = 8$ chains for the mixture of two Gaussians.

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

Example 6.6 (Witch's hat distribution). Consider a distribution with the following density:

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

which is a mixture of a Gaussian and a uniform distribution, both truncated to $[0,1] \times [0,1]$. Figure 6.6 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 $X_1|X_2$ is

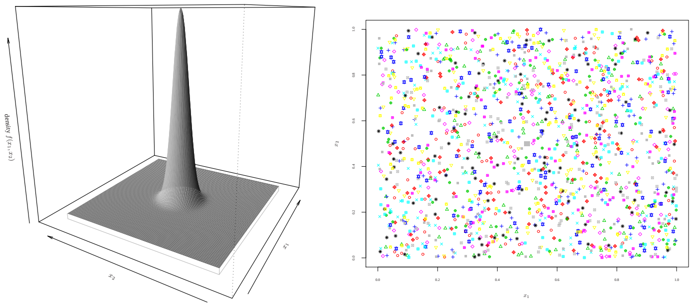
$$f(x_1|x_2) = \begin{cases} (1-\delta_{x_2})\phi_{(\boldsymbol{\mu}, \sigma^2 \cdot \mathbb{I})}(x_1, x_2) + \delta_{x_2} & \text{for } x_1 \in (0,1) \\ 0 & \text{else.} \end{cases}$$

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

Assume we want to estimate $\mathbb{P}(0.49 < X_1, X_2 \leq 0.51)$ 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. $\mathbb{P}(0.49 < X_1, X_2 \leq 0.51) = 0.999$.

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 6.6 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 6.6 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. $\triangleleft$



(a) Density for $\delta = 0.2$, $\boldsymbol{\mu} = (0.5, 0.5)'$, and $\sigma = 0.05$

(b) First 100 values from 15 samples using different starting values. ($\delta = 10^{-3}$, $\boldsymbol{\mu} = (0.5, 0.5)'$, and $\sigma = 10^{-5}$)

Figure 6.6. Density and sample from the witch's hat distribution.

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

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

$$\text{Var}\left(\frac{1}{T}\sum_{t=1}^{T} h(\mathbf{X}^{(t)})\right) \tag{6.2}$$

is larger than the variance we would obtain from an i.i.d. sample, which is $\text{Var}(h(\mathbf{X}^{(t)}))/T$.

The effective sample size (ESS) allows to quantify this loss of information caused by the 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 (6.2) as the estimate from the Markov chain $(\mathbf{X}^{(t)})_{t=1,\ldots,T}$.

In order to compute the variance (6.2) we make the simplifying assumption that $(h(\mathbf{X}^{(t)}))_{t=1,\ldots,T}$ is from a second-order stationary time series, i.e. $\text{Var}(h(\mathbf{X}^{(t)})) = \sigma^2$, and $\rho(h(\mathbf{X}^{(t)}), h(\mathbf{X}^{(t+\tau)})) = \rho(\tau)$. Then

$$\begin{aligned}
\text{Var}\left(\frac{1}{T}\sum_{t=1}^{T} h(\mathbf{X}^{(t)})\right) &= \frac{1}{T^2}\left(\sum_{t=1}^{T} \underbrace{\text{Var}(h(\mathbf{X}^{(t)}))}_{=\sigma^2} + 2 \sum_{1 \leq s < t \leq T} \underbrace{\text{Cov}(h(\mathbf{X}^{(s)}), h(\mathbf{X}^{(t)}))}_{=\sigma^2 \cdot \rho(t-s)}\right) \\
&= \frac{\sigma^2}{T^2}\left(T + 2\sum_{\tau=1}^{T-1}(T-\tau)\rho(\tau)\right) = \frac{\sigma^2}{T}\left(1 + 2\sum_{\tau=1}^{T-1}\left(1 - \frac{\tau}{T}\right)\rho(\tau)\right).
\end{aligned}$$

If $\sum_{\tau=1}^{+\infty} |\rho(\tau)| < +\infty$, then we can obtain from the dominated convergence theorem[4] that

[4] see e.g. Brockwell and Davis (1991, theorem 7.1.1) for details.

$$T \cdot \mathrm{Var}\left(\frac{1}{T}\sum_{t=1}^{T} h(\mathbf{X}^{(t)})\right) \longrightarrow \sigma^2 \left(1 + 2\sum_{\tau=1}^{+\infty} \rho(\tau)\right)$$

as $T \to \infty$. Note that the variance 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 $(h(\mathbf{X}^{(t)}))_{t=1,\ldots,T}$ is a first-order autoregressive time series (AR(1)), i.e. $\rho(\tau) = \rho(h(\mathbf{X}^{(t)}), h(\mathbf{X}^{(t+\tau)})) = \rho^{|\tau|}$, 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 6.7 (Gibbs sampling from a bivariate Gaussian (continued)). In examples 4.4 and 4.5 we obtained for the low-correlation setting that $\rho(X_1^{(t-1)}, X_1^{(t)}) = 0.078$, thus the effective sample size is

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

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:

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

◁

Chapter 7

State-space models and the Kalman filter algorithm

7.1 Motivation

In many real-world applications, observations arrive sequentially in time, and interest lies in performing on-line inference about unknown quantities from the given observations. If prior knowledge about these quantities is available, then it is possible to formulate a Bayesian model that incorporates this knowledge in the form of a prior distribution on the unknown quantities, and relates these to the observations via a likelihood function. Inference on the unknown quantities is then based on the posterior distribution obtained from Bayes' theorem. Examples include tracking an aircraft using radar measurements, speech recognition using noisy measurements of voice signals, or estimating the volatility of financial instruments using stock market data. In these examples, the unknown quantities of interest might be the location and velocity of the aircraft, the words in the speech signal, and the variance-covariance structure, respectively. In all three examples, the data is modelled dynamically in the sense that the underlying distribution evolves in time; these models are known as *dynamic models*. Sequential Monte Carlo (SMC) methods are a non-iterative, alternative class of algorithms to MCMC, designed specifically for inference in dynamic models. A comprehensive introduction to these methods is the book by Doucet et al. (2001). We point out that SMC methods are applicable in settings beyond dynamic models, such as non-sequential Bayesian inference, rare events simulation, and global optimization, provided that it is possible to define an evolving sequence of artificial distributions from which the distribution of interest is obtained via marginalisation.

Let $p_t(\mathbf{x}_t)$ denote the distribution at time $t \geq 1$, where $\mathbf{x}_t = (x_1, \ldots, x_t)$ typically increases in dimension with t, but it is possible that the dimension of $\mathbf{x}_t$ be constant $\forall t \geq 1$, or that $\mathbf{x}_t$ have one dimension less than $\mathbf{x}_{t-1}$. The particular feature of dynamic models is the evolving nature of the underlying distribution, where $p_t(\mathbf{x}_t)$ changes in time t as new observations are generated. Note that $\mathbf{x}_t$ are the quantities of interest, not the observations; the observations up to time t determine the form of the distribution, and this is implied by the subscript t in $p_t(\cdot)$. This is in contrast to non-dynamic models where the distribution is constant as new observations are generated, denoted by $p(\mathbf{x})$. In the latter case, MCMC methods have proven highly effective in generating approximate samples from low-dimensional distributions $p(\mathbf{x})$, when exact simulation is not possible. In the dynamic case, at each time step t a different MCMC sampler with stationary distribution $p_t(\mathbf{x}_t)$ is required, so the overall computational cost would increase with t. Moreover, for large t, designing the sampler and assessing its convergence would be increasingly difficult.

SMC methods are a non-iterative alternative to MCMC algorithms, based on the key idea that if $p_{t-1}(\mathbf{x}_{t-1})$ does not differ much from $p_t(\mathbf{x}_t)$, then it is possible to reuse the samples from $p_{t-1}(\mathbf{x}_{t-1})$ to obtain samples from

$p_t(\mathbf{x}_t)$. In most applications of interest, it is not possible to obtain exact samples from these evolving distributions, so the goal is to reuse an approximate sample, representative of $p_{t-1}(\mathbf{x}_{t-1})$, to obtain a good representation of $p_t(\mathbf{x}_t)$. Moreover, since inference is to be performed in real time as new observations arrive, it is necessary that the computational cost be fixed in t. We will see in the sequel that SMC methods are highly flexible, and widely applicable; we restrict our attention to a particular class of dynamic models called the *state-space model* (SSM).

7.2 State-space models

SSMs are a class of dynamic models that consist of an underlying Markov process, usually called the *state process*, X_t, that is hidden, i.e., unobserved, and an observed process, usually called the *observation process*, Y_t. Consider the following notation for a state-space model:

$$\begin{aligned} \text{observation:} \quad & y_t = a(x_t, u_t) \sim g(\cdot|x_t, \phi) \\ \text{hidden state:} \quad & x_t = b(x_{t-1}, v_t) \sim f(\cdot|x_{t-1}, \theta), \end{aligned}$$

where y_t and x_t are generated by functions $a(\cdot)$ and $b(\cdot)$ of the state and noise disturbances, denoted by u_t and v_t, respectively. Assume ϕ and θ to be known. Let $p(x_1)$ denote the distribution of the initial state x_1. The state process is a Markov chain, i.e., $p(x_t|x_1, \ldots, x_{t-1}) = p(x_t|x_{t-1}) = f(x_t|x_{t-1}, \theta)$, and the distribution of the observation y_t, conditional on x_t, is independent of previous values of the state and observation processes, i.e., $p(y_t|x_{1:t}, y_{1:t-1}) = p(y_t|x_t) = g(y_t|x_t, \phi)$. See Figure 7.1 for illustration.

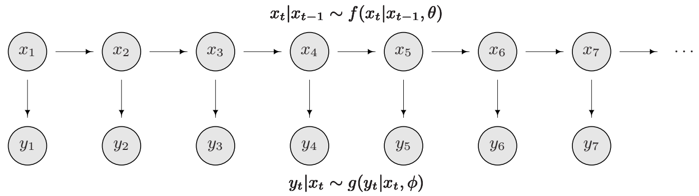

$\boldsymbol{x_t|x_{t-1} \sim f(x_t|x_{t-1}, \theta)}$
x_1
x_2
x_3
x_4
x_5
x_6
x_7
$\cdots$
y_1
y_2
y_3
y_4
y_5
y_6
y_7
$\boldsymbol{y_t|x_t \sim g(y_t|x_t, \phi)}$

Figure 7.1. The conditional independence structure of the first few states and observations in a hidden Markov Model.

Note that we use the notation $x_{1:t}$ to denote $x_1, \ldots, x_t$, and similarly for $y_{1:t}$. For simplicity, we drop the explicit dependence of the state transition and observation densities on θ and ϕ, and write $f(\cdot|x_{t-1})$, and $g(\cdot|x_t)$.

The literature sometimes distinguishes between state-space models where the state process is given by a discrete Markov chain, called *hidden Markov models* (HMM), as opposed to a continuous Markov chain. An extensive monograph on inference for state-space models is the book by Cappé et al. (2005), and a more recent overview is Cappé et al. (2007). In the present chapter and the following, we introduce several algorithms for inference in state-space models, and point out that the algorithms in Chapter 8 apply more generally to dynamic models.

7.2.1 Inference problems in SSMs

Under the notation introduced above, we have the joint density

$$p(x_{1:t}, y_{1:t}) = p(x_1)g(y_1|x_1)\prod_{i=2}^{t} p(x_i, y_i|x_{1:i-1}, y_{1:i-1}) = p(x_1)g(y_1|x_1)\prod_{i=2}^{t} f(x_i|x_{i-1})g(y_i|x_i),$$

and, by Bayes' theorem, the density of the distribution of interest

$$p(x_{1:t}|y_{1:t}) \propto p(x_{1:t}|y_{1:t-1})g(y_t|x_t) = p(x_{1:t-1}|y_{1:t-1})f(x_t|x_{t-1})g(y_t|x_t). \tag{7.1}$$

To connect this with the notation introduced for dynamic models, we can write $p_t(x_{1:t}) = p(x_{1:t}|y_{1:t})$, but we believe that stating the dependence on the observations explicitly leads to less confusion.

There exist several inference problems in state-space models that involve computing the posterior distribution of a collection of state variables conditional on a batch of observations:

- *filtering:* $p(x_t|y_{1:t})$
- *fixed lag smoothing:* $p(x_{t-l}|y_{1:t})$, for $0 \leq l \leq t-1$
- *fixed interval smoothing:* $p(x_{l:k}|y_{1:t})$, for $1 \leq l < k \leq t$
- *prediction:* $p(x_{l:k}|y_{1:t})$, for $k > t$ and $1 \leq l \leq k$.

The first three inference problems reduce to marginalisation of the full smoothing distribution $p(x_{1:t}|y_{1:t})$, i.e., integrating over the state variables that are not of interest, whereas the fourth reduces to marginalisation of

$$p(x_{1:k}|y_{1:t}) = p(x_{1:t}|y_{1:t}) \prod_{i=t+1}^{k} f(x_i|x_{i-1}).$$

So far we assumed that the state transition and observation densities are completely characterised, i.e., that the parameters θ and ϕ are known. If they are unknown, then Bayesian inference is concerned with the joint posterior distribution of the hidden states and the parameters:

$$p(x_{1:t}, \theta, \phi|y_{1:t}) \propto p(y_{1:t}|x_{1:t}, \theta, \phi)p(x_{1:t}|\theta, \phi)p(\theta, \phi) = p(\theta, \phi)p(x_1)g(y_1|x_1, \phi)\prod_{i=2}^{t} f(x_i|x_{i-1}, \theta)g(y_i|x_i, \phi).$$

If interest lies in the posterior distribution of the parameters, then the inference problem is called:

- *static parameter estimation:* $p(\theta, \phi|y_{1:t})$,

which reduces to integrating over the state variables in the joint posterior distribution $p(x_{1:t}, \theta, \phi|y_{1:t})$.

It is evident, then, that these inference problems depend on the tractability of the posterior distribution $p(x_{1:t}|y_{1:t})$, if the parameters are known, or $p(x_{1:t}, \theta, \phi|y_{1:t})$, otherwise. Notice that equation (7.1) gives the posterior distribution up to a normalising constant $\int p(x_{1:t}|y_{1:t-1})g(y_t|x_t)dx_{1:t}$, and it is oftentimes the case that the posterior distribution is known only up to a constant. In fact, these posterior distributions can be computed in closed form only in a few specific cases, such as the hidden Markov model, i.e., when the state process is a discrete Markov chain, and the linear Gaussian model, i.e., when the functions $a()$ and $b()$ are linear, and the noise disturbances u_t and v_t are Gaussian.

For HMMs with discrete state transition and observation distributions, the tutorial of Rabiner (1989) presents recursive algorithms for the smoothing and static parameter estimation problems. The Viterbi algorithm returns the optimal sequence of hidden states, i.e., the sequence that maximises the smoothing distribution, and the Expectation-Maximization (EM) algorithm returns parameter values for which the likelihood function of the observations attains a local maximum. If the observation distribution is continuous, then it can be approximated by a finite mixture of Gaussian distributions to insure that the EM algorithm applies to the problem of parameter estimation. These recursive algorithms involve summations over all states in the model, so they are impractical when the state space is large.

For the linear Gaussian model, the normalising constant in (7.1) can be computed analytically, and thus the posterior distribution of interest is known in closed form; in fact, it is the Gaussian distribution. The *Kalman filter* algorithm (Kalman, 1960) gives recursive expressions for the mean and variance of the filtering distribution $p(x_t|y_{1:t})$, under the assumption that all parameters in the model are known. Kalman (1960) obtains recursive expressions for the optimal values of the mean and variance parameters via a least-squares approach. The algorithm alternates between two steps: a prediction step (i.e., predict the state at time t conditional on $y_{1:t-1}$), and an update step (i.e., observe y_t, and update the prediction in light of the new observation). Section 7.3 presents a Bayesian formulation of the Kalman filter algorithm following Meinhold and Singpurwalla (1983).

When an analytic solution is intractable, exact inference is replaced by inference based on an approximation to the posterior distribution of interest. Grid-based methods using discrete numerical approximations to these posterior distributions are severely limited by parameter dimension. Alternatively, sequential Monte Carlo methods are a simulation-based approach that offer greater flexibility and scale better with increasing dimensionality. The key idea of SMC methods is to represent the posterior distribution by a weighted set of samples, called *particles*, that are *filtered* in time as new observations arrive, through a combination of sampling and resampling steps. Hence SMC sampling algorithms are oftentimes called *particle filters* (Carpenter et al., 1999). Chapter 8 presents SMC methods for the problems of filtering and smoothing.

7.3 The Kalman filter algorithm

From (7.1), the posterior distribution of the state x_t conditional on the observations $y_{1:t}$ is proportional to $p(x_t|y_{1:t-1})g(y_t|x_t)$. The first term is the distribution of x_t conditional on the first $t-1$ observations; computing this distribution is known as the *prediction* step. The second term is the distribution of the new observation y_t conditional on the hidden state at time t. Updating $p(x_t|y_{1:t-1})$ in light of the new observation involves taking the product of these two terms, and normalising; this is known as the *update* step. The result is the distribution of interest:

$$p(x_t|y_{1:t}) = \int p(x_{1:t}|y_{1:t})dx_1 \dots dx_{t-1} = \frac{p(x_t|y_{1:t-1})g(y_t|x_t)}{\int p(x_t|y_{1:t-1})g(y_t|x_t)dx_t}.$$

We now show how the prediction and update stages can be performed exactly for the linear Gaussian state-space model, which is represented as follows:

$$\text{observation:} \quad y_t = A_t x_t + u_t \sim \mathsf{N}(A_t x_t, \Phi^2) \tag{7.2}$$

$$\text{hidden state:} \quad x_t = B_t x_{t-1} + v_t \sim \mathsf{N}(B_t x_{t-1}, \Theta^2), \tag{7.3}$$

where $u_t \sim \mathsf{N}(0, \Phi^2)$ and $v_t \sim \mathsf{N}(0, \Theta^2)$ are independent noise sequences, and the parameters A_t, B_t, Φ^2, and Θ^2 are known. It is also possible to let the noise variances Φ^2 and Θ^2 vary with time; the derivation of the mean and variance of the posterior distribution follows as detailed below. We assume that both states and observations are vectors, in which case the parameters are matrices of appropriate sizes.

The Kalman filter algorithm proceeds as follows. Start with an initial Gaussian distribution on x_1: $x_1 \sim \mathsf{N}(\mu_1, \Sigma_1)$. At time $t-1$, let μ_{t-1} and Σ_{t-1} be the mean and variance of the Gaussian distribution of x_{t-1} conditional on $y_{1:t-1}$. Looking forward to time t, we begin by predicting the distribution of x_t conditional on $y_{1:t-1}$.

Prediction step: From equation (7.3), $x_t = B_t x_{t-1} + v_t$, where $x_{t-1}|y_{1:t-1} \sim \mathsf{N}(\mu_{t-1}, \Sigma_{t-1})$, and $v_t \sim \mathsf{N}(0, \Theta^2)$ independently. By results in multivariate statistical analysis (Anderson, 2003), we have that

$$x_t|y_{1:t-1} \sim \mathsf{N}(B_t\mu_{t-1}, B_t\Sigma_{t-1}B_t^T + \Theta^2), \tag{7.4}$$

where the superscript ^{T} indicates matrix transpose. This can be thought of as the prior distribution on x_t.

Update step: Upon observing y_t, we are interested in

$$p(x_t|y_{1:t}) \propto p(y_t|x_t, y_{1:t-1})p(x_t|y_{1:t-1}).$$

Following equation (7.2) and the result in (7.4), consider predicting y_t by $\hat{y}_t = A_t B_t \mu_{t-1}$, where $B_t\mu_{t-1}$ is the prior mean on x_t. The prediction error is $e_t = y_t - \hat{y}_t = y_t - A_t B_t \mu_{t-1}$, which is equivalent to observing y_t. So it follows that $p(x_t|y_{1:t}) \propto p(e_t|x_t, y_{1:t-1})p(x_t|y_{1:t-1})$. Finally, from (7.2), $e_t = A_t(x_t - B_t\mu_{t-1}) + u_t$, where $u_t \sim \mathsf{N}(0, \Phi^2)$, so $e_t|x_t, y_{1:t-1} \sim \mathsf{N}(A_t(x_t - B_t\mu_{t-1}), \Phi^2)$.

We now use the following results from Anderson (2003). Let X_1 an X_2 have a bivariate normal distribution:

$$\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \sim \mathsf{N}\left(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix} \right). \tag{7.5}$$

If equation (7.5) holds, then the conditional distribution of X_1 given $X_2 = x_2$ is given by

$$X_1|X_2 = x_2 \sim \mathsf{N}\left(\mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2), \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} \right). \tag{7.6}$$

Conversely, if (7.6) holds, and $X_2 \sim \mathsf{N}(\mu_2, \Sigma_{22})$, then (7.5) is true.

Since $e_t|x_t, y_{1:t-1} \sim \mathsf{N}(A_t(x_t - B_t\mu_{t-1}), \Phi^2)$ and $x_t|y_{1:t-1} \sim \mathsf{N}(B_t\mu_{t-1}, B_t\Sigma_{t-1}B_t^T + \Theta^2)$, it follows that

$$\begin{pmatrix} x_t \\ e_t \end{pmatrix} \Bigg| y_{1:t-1} \sim \mathsf{N}\left(\begin{pmatrix} B_t\mu_{t-1} \\ 0 \end{pmatrix}, \begin{pmatrix} B_t\Sigma_{t-1}B_t^T + \Theta^2 & (B_t\Sigma_{t-1}B_t^T + \Theta^2)A_t^T \\ A_t(B_t\Sigma_{t-1}B_t^T + \Theta^2) & A_t(B_t\Sigma_{t-1}B_t^T + \Theta^2)A_t^T + \Phi^2 \end{pmatrix} \right)$$

Using the result above, the filtering distribution is $p(x_t|y_{1:t}) = p(x_t|e_t, y_{1:t-1}) = \mathsf{N}(\mu_t, \Sigma_t)$, since observing e_t is equivalent to observing y_t, where

$$\begin{aligned} \mu_t &= B_t\mu_{t-1} + (B_t\Sigma_{t-1}B_t^T + \Theta^2)A_t^T(A_t(B_t\Sigma_{t-1}B_t^T + \Theta^2)A_t^T + \Phi^2)^{-1}e_t \\ \Sigma_t &= B_t\Sigma_{t-1}B_t^T + \Theta^2 - (B_t\Sigma_{t-1}B_t^T + \Theta^2)A_t^T(A_t(B_t\Sigma_{t-1}B_t^T + \Theta^2)A_t^T + \Phi^2)^{-1}A_t(B_t\Sigma_{t-1}B_t^T + \Theta^2). \end{aligned}$$

Algorithm 1 The Kalman filter algorithm.

```
1: Input: $\mu_1$ and $\Sigma_1$.
2: Set $t = 2$.
3: Compute mean and variance of prediction: $\hat{\mu}_t = B_t\mu_{t-1}$, $\hat{\Sigma}_t = B_t\Sigma_{t-1}B_t^T + \Theta^2$.
4: Observe $y_t$ and compute error in prediction: $e_t = y_t - A_t\hat{\mu}_t$.
5: Compute variance of prediction error: $R_t = A_t\hat{\Sigma}_tA_t^T + \Phi^2$.
6: Update the mean and variance of the posterior distribution:
       $\mu_t = \hat{\mu}_t + \hat{\Sigma}_tA_t^TR_t^{-1}e_t$
       $\Sigma_t = \hat{\Sigma}_t - \hat{\Sigma}_tA_t^TR_t^{-1}A_t\hat{\Sigma}_t$.
7: Set $t = t + 1$. Go to step 3.
```

Example 7.1 (First-order, linear autoregressive (AR(1)) model observed with noise). Consider the following AR(1) model:

$$\begin{aligned} x_t &= \phi x_{t-1} + \sigma_U u_t \sim \mathsf{N}(\phi x_{t-1}, \sigma_u^2) \\ y_t &= x_t + \sigma_V v_t \sim \mathsf{N}(x_t, \sigma_V^2), \end{aligned}$$

where $u_t \sim \mathsf{N}(0,1)$ and $v_t \sim \mathsf{N}(0,1)$ are independent, Gaussian white noise processes. The Markov chain $\{X_t\}_{t\geq 1}$ is a Gaussian random walk with transition kernel $\mathbf{K}(x_{t-1}, x_t)$ corresponding to the $\mathsf{N}(\phi x_{t-1}, \sigma_U^2)$ distribution.

A normal distribution $\mathsf{N}(\mu, \sigma^2)$ is stationary for $\{X_t\}_{t\geq 1}$ if $X_{t-1} \sim \mathsf{N}(\mu, \sigma^2)$ and $X_t|X_{t-1} = x_{t-1} \sim \mathsf{N}(\phi x_{t-1}, \sigma_u^2)$ imply that $X_t \sim \mathsf{N}(\mu, \sigma^2)$. We require that $\mathbb{E}(X_t) = \phi\mu = \mu$ and $\mathrm{Var}(X_t) = \phi^2\sigma^2 + \sigma_U^2 = \sigma^2$, which are satisfied by $\mu = 0$ and $\sigma^2 = \sigma_U^2/(1 - \phi^2)$, provided $|\phi| < 1$. In fact, the $\mathsf{N}\left(0, \sigma_U^2/(1 - \phi^2)\right)$ distribution is the unique stationary distribution of the chain.

Start the Kalman filter algorithm with $\mu_1 = 0$ and $\Sigma_1 = \sigma_U^2/(1 - \phi^2)$. At time $t - 1$, $t \geq 2$, let μ_{t-1} and Σ_{t-1} denote the posterior mean and variance, respectively. Then the mean and variance of the prediction at time t are: $\hat{\mu}_t = \phi\mu_{t-1}$ and $\hat{\Sigma}_t = \phi^2\Sigma_{t-1} + \sigma_U^2$. The prediction error is $e_t = y_t - \hat{\mu}_t$ with variance $\hat{\Sigma}_t + \sigma_V^2$. Finally, update the mean and variance of the posterior distribution:

$$\begin{aligned} \mu_t &= \hat{\mu}_t + \hat{\Sigma}_t\frac{1}{\hat{\Sigma}_t + \sigma_V^2}(y_t - \hat{\mu}_t) = \left(1 - \frac{\hat{\Sigma}_t}{\hat{\Sigma}_t + \sigma_V^2}\right)\hat{\mu}_t + \frac{\hat{\Sigma}_t}{\hat{\Sigma}_t + \sigma_V^2}y_t \\ \Sigma_t &= \hat{\Sigma}_t - \hat{\Sigma}_t\left(\frac{\hat{\Sigma}_t}{\hat{\Sigma}_t + \sigma_V^2}\right)\hat{\Sigma}_t = \hat{\Sigma}_t\left(1 - \hat{\Sigma}_t\left(\frac{\hat{\Sigma}_t}{\hat{\Sigma}_t + \sigma_V^2}\right)\right). \end{aligned}$$

◁

The Kalman filter algorithm (see Figure 1 for pseudo-code) is not robust to outlying observations y_t, i.e., when the prediction error e_t is large, because the mean μ_t is an unbounded function of e_t, and the variance Σ_t does not depend on the observed data y_t. Meinhold and Singpurwalla (1989) let the distributions of the error terms u_t and v_t be Student-t, and show that the posterior distribution of x_t given $y_{1:t}$ converges to the prior distribution of $p(x_t|y_{1:t-1})$ when e_t is large. In this case, the posterior distribution is no longer known exactly, but is approximated.

The underlying assumptions of the Kalman filter algorithm are that the state transition and observation equations are linear, and that the error terms are normally distributed. If the linearity assumption is violated, but the state transition and observation equations are differentiable functions, then the *extended Kalman filter* algorithm propagates the mean and covariance via the Kalman filter equations by linearizing the underlying non-linear model. If this model is highly non-linear, then this approach will result in very poor estimates of the mean and covariance. An alternative is the *unscented Kalman filter* which takes a deterministic sampling approach, representing the state transition distribution by a set of sample points that are propagated through the non-linear model. This approach improves the accuracy of the posterior mean and covariance; for details, see Wan and van der Merwe (2000).

Chapter 8

Sequential Monte Carlo

In this chapter we introduce sequential Monte Carlo (SMC) methods for sampling from dynamic models; these methods are based on importance sampling and resampling techniques. In particular, we present SMC methods the filtering and smoothing problems in state-space models.

8.1 Importance Sampling revisited

In Section 3.3, importance sampling is introduced as a technique for approximating a given integral $\mu = \int h(x)f(x)dx$ under a distribution f, by sampling from an instrumental distribution g with support satisfying supp$(g) \supset$ supp$(f \cdot h)$. This is based on the observation that

$$\mu = \mathbb{E}_f(h(X)) = \int h(x)f(x)dx = \int h(x)\frac{f(x)}{g(x)}g(x)dx = \int h(x)w(x)g(x)dx = \mathbb{E}_g(h(X) \cdot w(X)), \qquad (8.1)$$

where the right-most expectation in (8.1) is approximated by the empirical average of $h \cdot w$ evaluated at n i.i.d. samples from g.

In practice, we want to select g as close as possible to f such that the estimator of μ has finite variance. One sufficient condition is that $f(x) < M \cdot g(x)$ and $\text{Var}_f(h(X)) < \infty$. Under this condition, it is possible to use rejection sampling to sample from f and approximate μ. We argue in the following subsection that importance sampling is more efficient than rejection sampling, in terms of producing weights with smaller variance.

8.1.1 Importance Sampling versus Rejection Sampling

Let E be the support of f. Define the artificial target distribution $\bar{f}(x,y)$ on $E \times [0,1]$ as

$$\bar{f}(x,y) = \begin{cases} Mg(x) & \text{for } \left\{(x,y) : x \in E, y \in \left[0, \frac{f(x)}{Mg(x)}\right]\right\} \\ 0 & \text{otherwise} \end{cases}$$

where

$$f(x) = \int \bar{f}(x,y)dy = \int_0^{\frac{f(x)}{Mg(x)}} Mg(x)dy.$$

Consider the instrumental distribution $\bar{g}(x,y) = g(x)\mathsf{U}_{[0,1]}(y)$, for $(x,y) \in E \times [0,1]$, where $\mathsf{U}_{[0,1]}(\cdot)$ is the uniform distribution on $[0,1]$. Then, performing importance sampling on $E \times [0,1]$ with weights

$$w(x,y) = \frac{\bar{f}(x,y)}{\bar{g}(x,y)} = \begin{cases} M & \text{for } \left\{(x,y) : x \in E, y \in \left[0, \frac{f(x)}{Mg(x)}\right]\right\} \\ 0 & \text{otherwise} \end{cases}$$

is equivalent to rejection sampling to sample from f using instrumental distribution g. In contrast, importance sampling from f using instrumental distribution g has weights $w(x) = f(x)/g(x)$.

We now show that the weights for rejection sampling, $w(x,y)$, have higher variance than those for importance sampling, $w(x)$. For this purpose, we introduce the following technical lemma which relates the variance of a random variable to its conditional variance and expectation. In the following, any expectation or variance with a subscript corresponding to a random variable should be interpreted as the expectation or variance with respect to that random variable.

Lemma 8.1 (Law of Total Variance). *Given two random variables, A and B, on the same probability space, such that* $\text{Var}(A) < \infty$*, then the following decomposition exists:*

$$\text{Var}(A) = \mathbb{E}_B\left[\text{Var}_A(A|B)\right] + \text{Var}_B\left(\mathbb{E}_A[A|B]\right).$$

Proof. By definition, and the law of total probability, we have:

$$\text{Var}(A) = \mathbb{E}\left[A^2\right] - \mathbb{E}[A]^2 = \mathbb{E}_B\left[\mathbb{E}_A\left[A^2|B\right]\right] - \mathbb{E}_B\left[\mathbb{E}_A[A|B]\right]^2.$$

Considering the definition of conditional variance, and then variance, it is clear that:

$$\begin{aligned} \text{Var}(A) &= \mathbb{E}_B\left[\text{Var}_A(A|B) + \mathbb{E}_A[A|B]^2\right] - \mathbb{E}_B\left[\mathbb{E}_A[A|B]\right]^2 \\ &= \mathbb{E}_B\left[\text{Var}_A(A|B)\right] + \mathbb{E}_B\left[\mathbb{E}_A[A|B]^2\right] - \mathbb{E}_B\left[\mathbb{E}_A[A|B]\right]^2 \\ &= \mathbb{E}_B\left[\text{Var}_A(A|B)\right] + \text{Var}_B\left(\mathbb{E}_A[A|B]\right). \end{aligned}$$

□

Returning to importance sampling versus rejection sampling, we have by Lemma 8.1 that

$$\begin{aligned} \text{Var}(w(X,Y)) = \text{Var}(\mathbb{E}(w(X,Y)|X)) + \mathbb{E}(\text{Var}(w(X,Y)|X)) &= \text{Var}(w(X)) + \mathbb{E}(\text{Var}(w(X,Y)|X)) \\ &\geq \text{Var}(w(X)), \end{aligned}$$

since

$$\mathbb{E}(w(X,Y)|X) = \int_0^1 w(x,y)\frac{\bar{g}(x,y)}{g(x)}dy = \int_0^{\frac{f(X)}{Mg(X)}} M dy = \frac{f(X)}{g(X)} = w(X),$$

and the fact that $\text{Var}(w(X,Y)|X)$ is a non-negative function.

8.1.2 Empirical distributions

Consider a collection of i.i.d. points $\{x_i\}_{i=1}^n$ in E drawn from f. We can approximate f by the following *empirical measure*, associated with these points,

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^n \mathbb{I}(x = x_i) = \frac{1}{n}\sum_{i=1}^n \delta_{x_i}(x),$$

where, for any $x \in E$, $\delta_{x_i}(x)$ is the Dirac measure which places all of its mass at point x_i, i.e., $\delta_{x_i}(x) = 1$ if $x = x_i$ and 0 otherwise. Similarly, we can define the *empirical distribution* function

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

where $\mathbb{I}(x_i \leq x)$ is the indicator of event $x_i \leq x$.

If the collection of points has associated positive, real-valued weights $\{x_i, w_i\}_{i=1}^n$, then the empirical measure is defined as follows

$$\hat{f}(x) = \frac{\sum_{i=1}^n w_i \delta_{x_i}(x)}{\sum_{i=1}^n w_i}.$$

For fixed $x \in E$, $\hat{f}(x)$ and $\hat{F}(x)$, as functions of a random sample, are a random measure and distribution, respectively. The strong law of large numbers justifies approximating the true density and distribution functions by $\hat{f}(x)$ and $\hat{F}(x)$ as the number of samples n increases to infinity.

From these approximations, we can then estimate integrals with respect to f by integrals with respect to the associated empirical measure. Let $h : E \to \mathbb{R}$ be a measureable function. Then

$$\mathbb{E}_f(h(X)) = \int h(x) f(x) dx \approx \int h(x) \hat{f}(x) dx = \frac{1}{n} \sum_{i=1}^n h(x_i),$$

or, if the random sample is weighted,

$$\mathbb{E}_f(h(X)) = \int h(x) f(x) dx \approx \int h(x) \hat{f}(x) dx = \frac{\sum_{i=1}^n w_i h(x_i)}{\sum_{i=1}^n w_i}.$$

Sequential Importance Sampling exploits the idea of sequentially approximating an intractable density function f by the corresponding empirical measure $\hat{f}$ associated with a random sample from an instrumental distribution g, and properly weighted with weights $w_i \propto f(x_i)/g(x_i)$.

8.2 Sequential Importance Sampling

Sequential Importance Sampling (SIS) performs importance sampling sequentially to sample from a distribution $p_t(\mathbf{x}_t)$ in a dynamic model. Let $\left\{\mathbf{x}_t^{(i)}, w_t^{(i)}\right\}_{i=1}^n$ be a collection of samples, called *particles*, that targets $p_t(\mathbf{x}_t)$. At time $t+1$, the target distribution evolves to $p_{t+1}(\mathbf{x}_{t+1})$; for $i = 1, \ldots, n$, sample the $(t+1)$st component $x_{t+1}^{(i)}$ from an instrumental distribution, update the weight $w_{t+1}^{(i)}$, and append $x_{t+1}^{(i)}$ to $\mathbf{x}_t^{(i)}$. The desired result is a collection of samples $\left\{\mathbf{x}_{t+1}^{(i)}, w_{t+1}^{(i)}\right\}_{i=1}^n$ that targets $p_{t+1}(\mathbf{x}_{t+1})$.

The idea is to choose an instrumental distribution such that importance sampling can proceed sequentially. Let $q_{t+1}(\mathbf{x}_{t+1})$ denote the instrumental distribution, and suppose that it can be factored as follows:

$$q_{t+1}(\mathbf{x}_{t+1}) = q_1(x_1) \prod_{i=2}^{t+1} q_i(x_i|\mathbf{x}_{i-1}) = q_t(\mathbf{x}_t) q_{t+1}(x_{t+1}|\mathbf{x}_t).$$

Then, the weight $w_{t+1}^{(i)}$ can be computed incrementally from $w_t^{(i)}$. At time $t = 1$, sample $x_1^{(i)} \sim q_1(x_1)$ for $i = 1, \ldots, n$, and set $w_1^{(i)} = p_1(x_1)/q_1(x_1)$. Normalise the weights by dividing them by $\sum_{j=1}^n w_1^{(j)}$. At time $t > 1$,

$$w_t^{(i)} = \frac{p_t(\mathbf{x}_t^{(i)})}{q_t(\mathbf{x}_t^{(i)})},$$

so, at the following time step, we sample $x_{t+1}^{(i)} \sim q_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})$, and update the weight

$$w_{t+1}^{(i)} = \frac{p_{t+1}(\mathbf{x}_{t+1}^{(i)})}{q_{t+1}(\mathbf{x}_{t+1}^{(i)})} = \frac{p_{t+1}(\mathbf{x}_{t+1}^{(i)})}{q_t(\mathbf{x}_t^{(i)}) q_{t+1}(x_{t+1}^{(i)}|\mathbf{x}_t^{(i)})} = w_t^{(i)} \frac{p_{t+1}(\mathbf{x}_{t+1}^{(i)})}{p_t(\mathbf{x}_t^{(i)}) q_{t+1}(x_{t+1}^{(i)}|\mathbf{x}_t^{(i)})}. \tag{8.2}$$

Normalise the weights. The term

$$\frac{p_{t+1}(\mathbf{x}_{t+1}^{(i)})}{p_t(\mathbf{x}_t^{(i)}) q_{t+1}(x_{t+1}^{(i)}|\mathbf{x}_t^{(i)})}$$

is known as the *incremental weight*. The intuition is that if the weighted sample $\left\{\mathbf{x}_t^{(i)}, w_t^{(i)}\right\}_{i=1}^n$ is a good approximation to the target distribution at time t, $p_t(\mathbf{x}_t)$, then, for appropriately chosen instrumental distribution $q_{t+1}(x_{t+1}|\mathbf{x}_t)$, the weighted sample $\left\{\mathbf{x}_{t+1}^{(i)}, w_{t+1}^{(i)}\right\}_{i=1}^n$ is also a good approximation to $p_{t+1}(\mathbf{x}_{t+1})$. For details, see Liu and Chen (1998).

8.2.1 Optimal instrumental distribution

Kong et al. (1994) and Doucet et al. (2000) prove that the unconditional variance of the weights increases over time. So in the long run, a few of the weights contain most of the probability mass, while most of the particles have normalised weights with near zero values. This phenomenon is know in the literature as *weight degeneracy*. Chopin (2004) argues that the SIS algorithm suffers from the curse of dimensionality, in the sense that weight degeneracy grows exponentially in the dimension t.

So it is natural to seek an instrumental distribution that minimises the variance of the weights. Doucet et al. (2000) show that the optimal instrumental distribution is $q_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)}) = p_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})$, in the sense that the variance of $w_{t+1}^{(i)}$ conditional upon $\mathbf{x}_t^{(i)}$ is zero. This result appears in the following proposition.

Proposition 8.2. *The instrumental distribution $q_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)}) = p_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})$ minimises the variance of the weight $w_{t+1}^{(i)}$ conditional upon $\mathbf{x}_t^{(i)}$.*

Proof. From (8.2),

$$\begin{aligned} \operatorname{Var}_{q_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})}\left(w_{t+1}^{(i)}\right) &= \left(w_t^{(i)}\right)^2 \operatorname{Var}_{q_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})}\left(\frac{p_{t+1}(\mathbf{x}_t^{(i)}, x_{t+1})}{p_t(\mathbf{x}_t^{(i)}) q_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})}\right) \\ &= \left(w_t^{(i)}\right)^2 \left\{\int \left[\frac{p_{t+1}(\mathbf{x}_t^{(i)}, x_{t+1})}{p_t(\mathbf{x}_t^{(i)})}\right]^2 \frac{1}{q_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})} dx_{t+1} - \left[\frac{p_{t+1}(\mathbf{x}_t^{(i)})}{p_t(\mathbf{x}_t^{(i)})}\right]^2\right\} \\ &= 0, \end{aligned}$$

if $q_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)}) = p_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})$. □

More intuitively, Liu and Chen (1998) rewrite the incremental weight as

$$\frac{p_{t+1}(\mathbf{x}_{t+1}^{(i)})}{p_t(\mathbf{x}_t^{(i)}) q_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})} = \frac{p_{t+1}(\mathbf{x}_t^{(i)})}{p_t(\mathbf{x}_t^{(i)})} \frac{p_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})}{q_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})},$$

and interpret the second ratio on the right hand side as correcting the discrepancy between $q_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})$ and $p_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})$, when they are different. Hence the optimal instrumental distribution is $p_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)})$.

In practice, however, sampling from the optimal instrumental distribution is usually not possible, so other choices of instrumental distributions are considered. Oftentimes it is possible to find good approximations to the optimal instrumental distribution; in such instances, the variance of the corresponding weights is low for small t, but weight degeneracy still occurs at t increases.

When $q_{t+1}(x_{t+1}|\mathbf{x}_t^{(i)}) = p_t(x_{t+1}|\mathbf{x}_t^{(i)})$, i.e., the distribution $p_t(\mathbf{x}_t)$ is used to predict x_{t+1}, then the incremental weight simplifies to $p_{t+1}(\mathbf{x}_{t+1}^{(i)})/p_t(\mathbf{x}_{t+1}^{(i)})$. The resulting SIS algorithm is known as the *bootstrap filter*. It was first introduced by Gordon et al. (1993) in the context of Bayesian filtering for non-linear, non-Gaussian state-space models.

8.2.2 SIS for state-space models

Recall the state-space model introduced in Section 7.2. For simplicity, assume that all parameters are known.

$$\begin{aligned} \text{observation:} \quad & y_t = a(x_t, u_t) \sim g(\cdot|x_t) \\ \text{hidden state:} \quad & x_t = b(x_{t-1}, v_t) \sim f(\cdot|x_{t-1}). \end{aligned}$$

We present the SIS algorithm to sample approximately from the filtering distribution $p_{t+1}(x_{t+1}) = p(x_{t+1}|y_{1:t+1})$, and the smoothing distribution $p_{t+1}(\mathbf{x}_{t+1}) = p(x_{1:t+1}|y_{1:t+1})$.

The instrumental distribution is $q_{t+1}(x_{t+1}|x_{1:t}^{(i)})$, where the subscript $t+1$ indicates that the distribution may incorporate all the data up to time $t+1$: $y_{1:t+1}$. For the bootstrap filter, $q_{t+1}(x_{t+1}|x_{1:t}^{(i)}) = p(x_{t+1}|x_{1:t}^{(i)}, y_{1:t}) =$

$f(x_{t+1}|x_t^{(i)})$, i.e., the instrumental distribution does not incorporate the most recent observation y_{t+1}, and the weight is

$$w_{t+1}^{(i)} = w_t^{(i)} \frac{p(x_{1:t+1}^{(i)}|y_{1:t+1})}{p(x_{1:t}^{(i)}|y_{1:t})f(x_{t+1}^{(i)}|x_t^{(i)})} \propto w_t^{(i)} \frac{p(x_{1:t}^{(i)}|y_{1:t})f(x_{t+1}^{(i)}|x_t^{(i)})g(y_{t+1}|x_{t+1}^{(i)})}{p(x_{1:t}^{(i)}|y_{1:t})f(x_{t+1}^{(i)}|x_t^{(i)})} = w_t^{(i)} g(y_{t+1}|x_{t+1}^{(i)})$$

by equation (7.1). In this case, the incremental weight does not depend on past trajectories $x_{1:t}^{(i)}$, but only on the likelihood function of the most recent observation.

Gordon et al. (1993) introduce the bootstrap filter in a very intuitive way, without reference to SIS, but rather as a two-stage recursive process with a propagate step, followed by an update step (similar in spirit to the recursions in the Kalman filter algorithm). Write the filtering density as follows:

$$\begin{aligned} p(x_{t+1}|y_{1:t+1}) = \frac{g(y_{t+1}|x_{t+1})p(x_{t+1}|y_{1:t})}{p(y_{t+1}|y_{1:t})} \quad &\propto \quad g(y_{t+1}|x_{t+1}) \int p(x_{t+1}, x_t|y_{1:t})dx_t \\ &\propto \quad g(y_{t+1}|x_{t+1}) \int f(x_{t+1}|x_t)p(x_t|y_{1:t})dx_t. \end{aligned} \tag{8.3}$$

Now, let $\left\{x_t^{(i)}, w_t^{(i)}\right\}_{i=1}^n$ be a weighted sample representing the filtering density at time t, such that $\sum_{j=1}^n w_t^{(j)} = 1$. Then, $\hat{p}(x_t|y_{1:t}) = \sum_{i=1}^n w_t^{(i)} \delta_{x_t^{(i)}}(x_t)$ is the empirical measure approximating $p(x_t|y_{1:t})$, and $\hat{p}(x_{t+1}|y_{1:t}) = \sum_{i=1}^n w_t^{(i)} f(x_{t+1}|x_t^{(i)})$. Furthermore, by (8.3), we have the approximation

$$\hat{p}(x_{t+1}|y_{1:t+1}) = \sum_{i=1}^n w_t^{(i)} g(y_{t+1}|x_{t+1}) f(x_{t+1})|x_t^{(i)}),$$

and sampling proceeds in two steps:

Propagate step: for $i = 1:n$, sample $x_{t+1}^{(i)} \sim f(x_{t+1}|x_t^{(i)})$.

Update step: for $i = 1:n$, weigh $x_{t+1}^{(i)}$ with weight $w_{t+1}^{(i)} = w_t^{(i)} g(y_{t+1}|x_{t+1}^{(i)})$. Normalise the weights.

In contrast to the instrumental distribution of the bootstrap filter, the optimal instrumental distribution incorporates the most recent observation:

$$\begin{aligned} q_{t+1}(x_{t+1}|x_{1:t}^{(i)}) &= p(x_{t+1}|x_{1:t}^{(i)}, y_{1:t+1}) \\ &= \frac{p(y_{t+1}|x_{1:t}^{(i)}, x_{t+1}, y_{1:t})p(x_{t+1}|x_{1:t}^{(i)}, y_{1:t})}{\int p(y_{t+1}|x_{1:t}^{(i)}, x_{t+1}, y_{1:t})p(x_{t+1}|x_{1:t}^{(i)}, y_{1:t})dx_{t+1}} \\ &= \frac{g(y_{t+1}|x_{t+1})f(x_{t+1}|x_t^{(i)})}{\int g(y_{t+1}|x_{t+1})f(x_{t+1}|x_t^{(i)})dx_{t+1}}, \end{aligned}$$

where the normalising constant equals the predictive distribution of y_{t+1} conditional on x_t, i.e., $p(y_{t+1}|x_t)$. So the weight function becomes $w_{t+1}^{(i)} \propto w_t^{(i)} p(y_{t+1}|x_t)$.

Example 8.1 (AR(1) model observed with noise (continued from example 7.1)). The optimal instrumental distribution is

$$\begin{aligned} q_{t+1}(x_{t+1}|x_{1:t}) &\propto g(y_{t+1}|x_{t+1})f(x_{t+1}|x_t) \\ &\propto \exp\left\{-\frac{1}{2\sigma_V^2}(y_{t+1} - x_{t+1})^2\right\} \exp\left\{-\frac{1}{2\sigma_U^2}(x_{t+1} - \phi x_t)^2\right\} \\ &= \exp\left\{-\frac{1}{2}\left[x_{t+1}^2\left(\frac{1}{\sigma_V^2} + \frac{1}{\sigma_U^2}\right) - 2x_{t+1}\left(\frac{y_{t+1}}{\sigma_V^2} + \frac{\phi x_t}{\sigma_U^2}\right) + \frac{y_{t+1}^2}{\sigma_V^2} + \frac{\phi^2 x_t^2}{\sigma_U^2}\right]\right\} \\ &\propto \exp\left\{-\frac{1}{2\sigma^2}(x_{t+1} - \mu)^2\right\}, \end{aligned}$$

implying that the distribution is $\mathsf{N}(\mu, \sigma^2)$ with

$$\begin{aligned} \mu &= \frac{\sigma_U^2 \sigma_V^2}{\sigma_U^2 + \sigma_V^2}\left(\frac{y_{t+1}}{\sigma_V^2} + \frac{\phi x_t}{\sigma_U^2}\right) \\ \sigma^2 &= \frac{\sigma_U^2 \sigma_V^2}{\sigma_U^2 + \sigma_V^2}. \end{aligned}$$

The normalising constant of this optimal instrumental distribution is

$$\begin{aligned} p(y_{t+1}|x_t) &= \int g(y_{t+1}|x_{t+1})f(x_{t+1}|x_t)dx_{t+1} \\ &\propto \exp\left(-\frac{1}{2\sigma_V^2}y_{t+1}^2\right) \exp\left\{\frac{1}{2}\left(\frac{1}{\sigma_U^2} + \frac{1}{\sigma_V^2}\right)\left(\frac{\sigma_U^2\sigma_V^2}{\sigma_U^2 + \sigma_V^2}\right)^2 \left(\frac{y_{t+1}}{\sigma_V^2} + \frac{\phi x_t}{\sigma_U^2}\right)^2\right\} \times \\ &\int \frac{1}{\sqrt{2\pi}}\left(\frac{1}{\sigma_U^2} + \frac{1}{\sigma_V^2}\right)^{1/2} \exp\left\{-\frac{1}{2}\left(\frac{1}{\sigma_U^2} + \frac{1}{\sigma_V^2}\right)\left[x_{t+1} - \frac{\sigma_U^2\sigma_V^2}{\sigma_U^2 + \sigma_V^2}\left(\frac{y_{t+1}}{\sigma_V^2} + \frac{\phi x_t}{\sigma_U^2}\right)\right]^2\right\} dx_{t+1} \\ &\propto \exp\left\{-\frac{1}{2}\left[y_{t+1}^2 \frac{1}{\sigma_U^2 + \sigma_V^2} - \frac{2\phi x_t}{\sigma_U^2 + \sigma_V^2} y_{t+1}\right]\right\}, \end{aligned}$$

i.e., $y_{t+1}|x_t \sim \mathsf{N}(\phi x_t, \sigma_U^2 + \sigma_V^2)$. ◁

Let $\left\{x_{1:t+1}^{(i)}, w_{t+1}^{(i)}\right\}_{i=1}^n$ be a weighted sample, normalised such that $\sum_{j=1}^n w_{t+1}^{(j)} = 1$, from $p(x_{1:t+1}|y_{1:t+1})$. Then the filtering and smoothing densities are approximated by the corresponding empirical measures:

$$\hat{p}(x_{t+1}|y_{1:t+1}) = \sum_{i=1}^n w_{t+1}^{(i)} \delta_{x_{t+1}^{(i)}}(x_{t+1}), \quad \hat{p}(x_{1:t+1}|y_{1:t+1}) = \sum_{i=1}^n w_{t+1}^{(i)} \delta_{x_{1:t+1}^{(i)}}(x_{1:t+1}).$$

Algorithm 2 is the general SIS algorithm for state-space models. The computational complexity to generate n particles representing $p(x_{1:t}|y_{1:t})$ is $O(nt)$.

Algorithm 2 The SIS algorithm for state-space models

```
1: Set $t = 1$.
2: For $i = 1:n$, sample $x_1^{(i)} \sim q_1(x_1)$.
3: For $i = 1:n$, set $w_1^{(i)} \propto p(x_1^{(i)})g(y_1|x_1^{(i)})/q_1(x_1^{(i)})$. Normalise such that $\sum_{j=1}^n w_1^{(j)} = 1$.
4: At time $t+1$, do:
5: For $i = 1:n$, sample $x_{t+1}^{(i)} \sim q_{t+1}(x_{t+1}|x_{1:t}^{(i)})$.
6: For $i = 1:n$, set $w_{t+1}^{(i)} \propto w_t^{(i)} f(x_{t+1}^{(i)}|x_t^{(i)})g(y_{t+1}|x_{t+1}^{(i)})/q_{t+1}(x_{t+1}^{(i)}|x_{1:t}^{(i)})$. Normalise such that $\sum_{j=1}^n w_{t+1}^{(j)} = 1$.
7: The filtering and smoothing densities at time $t+1$ may be approximated by
       $\hat{p}(x_{t+1}|y_{1:t+1}) = \sum_{i=1}^n w_{t+1}^{(i)} \delta_{x_{t+1}^{(i)}}(x_{t+1}), \quad \hat{p}(x_{1:t+1}|y_{1:t+1}) = \sum_{i=1}^n w_{t+1}^{(i)} \delta_{x_{1:t+1}^{(i)}}(x_{1:t+1}).$
8: Set $t = t+1$. Go to step 4.
```

8.3 Sequential Importance Sampling with Resampling

One approach to limiting the weight degeneracy problem is to choose an instrumental distribution that lowers the variance of the weights; a second approach is to introduce a *resampling* step after drawing and weighing the particles at time $t+1$. This idea of *rejuvenating* the particles by resampling was first suggested by Gordon et al. (1993).

The idea is as follows: let $\left\{x_t^{(i)}, w_t^{(i)}\right\}_{i=1}^n$ be a weighted sample from $p_t(x_t)$, obtained by importance sampling, and normalised such that $\sum_{j=1}^n w_t^{(j)} = 1$. The empirical measure is $\hat{p}_t(x_t) = \sum_{i=1}^n w_t^{(i)} \delta_{x_t^{(i)}}(x_t)$. Under suitable regularity conditions, the law of large number states that, for any fixed measurable function h, as $n \to \infty$,

$$\int h(x_t)\hat{p}_t(x_t)dx_t = \sum_{i=1}^n w_t^{(i)} h(x_t^{(i)}) \to \int h(x_t)p_t(x_t)dx_t.$$

Suppose now that we draw a sample of size n' from $\hat{p}_t(x_t)$ with replacement, i.e., for $j = 1, \ldots, n$, $\tilde{x}_t^{(j)} = x_t^{(i)}$ with probability $w_t^{(i)}$. The new particles have equal weights: $\tilde{w}_t^{(i)} = 1/n'$. Again, invoking the law of large numbers, as $n' \to \infty$,

$$\frac{1}{n'} \sum_{j=1}^{n'} h(\tilde{x}_t^{(j)}) \to \sum_{i=1}^{n} w_t^{(i)} h(x_t^{(i)}).$$

So, for n' large, the integral of the function h with respect to the new empirical measure based on $\left\{\tilde{x}_t^{(j)}, 1/n'\right\}_{j=1}^{n'}$ is a good approximation to the integral of h with respect to $\hat{p}_t(x_t)$.

In the SIS algorithm, resampling is applied to the entire trajectory $x_{1:t+1}^{(i)}$, not simply to the last value x_{t+1}; the new algorithm is known as SIS with Resampling (SISR). The advantage of resampling is that it eliminates particle trajectories with low weights, and replicates those with large weights; all of the resampled particles then contribute significantly to the importance sampling estimates.

On the other hand, replicating trajectories with large weights reduces diversity by depleting the number of distinct particle values at any time step in the past. At time $t+1$, new values $x_{t+1}^{(i)}$ are sampled and appended to the particle trajectories; resampling then eliminates some of these trajectories. Since the values at $t' < t+1$ are not rejuvenated as t increases, their diversity decreases due to resampling. In the extreme case, the smoothing density $p_{t+1}(x_{1:t+1})$ is approximated by a system of particle trajectories with a single common acestor. Figure 8.1 displays this situation graphically.

In general, at the current time step $t+1$, we can obtain a good approximation to the filtering density $p_{t+1}(x_{t+1})$ from the particles and their corresponding weights, provided the number of particles is large enough. However, approximations to the smoothing density $p_{t+1}(x_{1:t+1})$ and fixed interval smoothing densities $p_{t+1}(x_{1:t'})$, for $t' \ll t$, will be poor. Chopin (2004) argues that for smoothing the first state x_1, i.e., approximating $p_{t+1}(x_1)$, the SIS algorithm is more efficient than the SISR algorithm, but that the latter can be expected to be more efficient in filtering the states, i.e., approximating $p_{t+1}(x_{t+1})$. In particular, if the instrumental distribution of the SISR algorithm has a certain abilitiy to "forget the past" (i.e., to forget its initial condition), then the asymptotic variance of the estimator is bounded from above in t.

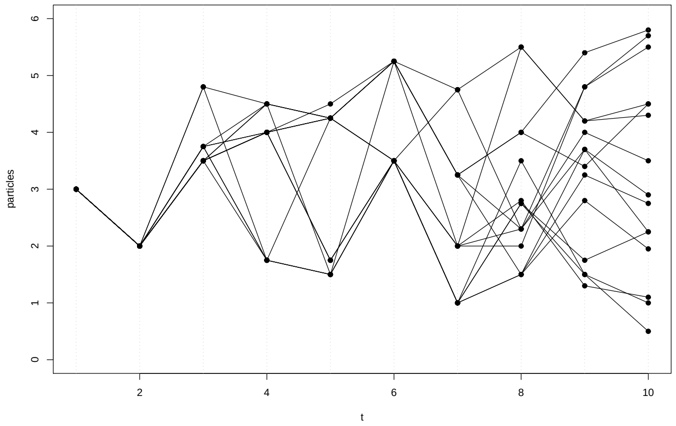


Figure 8.1. Plot of particle trajectories with a single common ancestor after resampling.

Moreover, the resampled trajectories are no longer independent, so resampling has the additional effect of increasing the Monte Carlo variance of an estimator at the current time step (Chopin, 2004). However, it can reduce the variance of estimators at later times. So, if one is interested in estimating the integral $\int h(x_{t+1})p_{t+1}(x_{t+1})dx_{t+1}$, for some measurable function h, then the estimator $\sum_{i=1}^n w_{t+1}^{(i)} h(x_{t+1}^{(i)})$ must be computed before resampling (as it will have lower Monte Carlo error than if it were computed after the resampling step). So far we discussed resampling via multinomial sampling; other resampling schemes exist that introduce lower Monte Carlo variance, and no additional bias, such as *stratified sampling* (Carpenter et al., 1999), and *residual sampling* (Liu and Chen, 1998). Chopin (2004) shows that residual sampling always outperforms multinomial sampling: the resulting estimator using the former sampling scheme has lower asymptotic variance.

8.3.1 Effective sample size

Resampling at every time step introduces unnecesary variation, so a trade-off is required between reducing the Monte Carlo variance in the future, and increasing the variance at the recent time step. Following Kong et al. (1994), we define the *effective sample size* (ESS), a measure of the efficiency of estimation based on a given SIS sample, compared to estimation based on a sample of i.i.d. draws from the target distribution $p_{t+1}(x_{t+1})$.

Let $h(x_{t+1})$ be a measurable function, and suppose that we're interested in estimating the mean $\mu = \mathbb{E}_{p_{t+1}(x_{t+1})}\left(h(X_{t+1})\right)$. Let $\left\{x_{t+1}^{(i)}, w_{t+1}^{(i)}\right\}_{i=1}^n$ be a weighted sample approximating $p_{t+1}(x_{t+1})$, obtained by SIS, and let $\left\{y_{t+1}^{(i)}\right\}_{i=1}^n$ be a sample of i.i.d. draws from $p_{t+1}(x_{t+1})$.

The SIS estimator of μ is

$$\hat{\mu}^{IS} = \frac{\sum_{i=1}^n w_{t+1}^{(i)} h(x_{t+1}^{(i)})}{\sum_{j=1}^n w_{t+1}^{(j)}},$$

and the Monte Carlo estimator is

$$\hat{\mu}^{MC} = \frac{1}{n} \sum_{i=1}^n h(y_{t+1}^{(i)}).$$

Then the relative efficiency of SIS in estimating μ can be measured by the ratio

$$\frac{\mathrm{Var}\left(\hat{\mu}^{IS}\right)}{\mathrm{Var}\left(\hat{\mu}^{MC}\right)},$$

which, in general, cannot be computed exactly. Kong et al. (1994) propose the following approximation that has the advantage of being independent of the function h:

$$\frac{\mathrm{Var}\left(\hat{\mu}^{IS}\right)}{\mathrm{Var}\left(\hat{\mu}^{MC}\right)} \approx 1 + \mathrm{Var}_{q_{t+1}(x_{t+1})}(\bar{w}_{t+1}),$$

where $q_{t+1}(x_{t+1})$ is the instrumental distribution in SIS, and $\bar{w}_{t+1}$ is the normalised version of w_{t+1}, i.e., $\int \bar{w}_{t+1} q_{t+1}(x_{t+1}) dx_{t+1} = 1$. In general, $\mathrm{Var}_{q_{t+1}(x_{t+1})}(\bar{w}_{t+1})$ is impossible to obtain, but can be approximated by the sample variance of $\left\{\bar{w}_{t+1}^{(i)}\right\}_{i=1}^n$, where $\bar{w}_{t+1}^{(i)} = w_{t+1}^{(i)} / \sum_{j=1}^n w_{t+1}^{(j)}$ are the normalised weights.

In practice, the ESS is defined as follows:

$$ESS = \frac{n}{1 + \mathrm{Var}_{q_{t+1}(x_{t+1})}(\bar{w}_{t+1})} = \frac{n}{\mathbb{E}_{q_{t+1}(x_{t+1})}(\bar{w}_{t+1})^2} \approx \frac{n}{n \sum_{i=1}^n \left(\bar{w}_{t+1}^{(i)}\right)^2} = \frac{\left(\sum_{j=1}^n w_{t+1}^{(j)}\right)^2}{\sum_{i=1}^n \left(w_{t+1}^{(i)}\right)^2},$$

since the weights are normalised to sum to 1, and

$$\mathbb{E}_{q_{t+1}(x_{t+1})}(\bar{w}_{t+1})^2 = \mathbb{E}_{q_{t+1}(x_{t+1})} \left(\frac{w_{t+1}}{C}\right)^2 = \frac{1}{C^2} \mathbb{E}_{q_{t+1}(x_{t+1})}(w_{t+1})^2 \approx \frac{n^{-1} \sum_{i=1}^n \left(w_{t+1}^{(i)}\right)^2}{n^{-2} \left(\sum_{j=1}^n w_{t+1}^{(j)}\right)^2},$$

where C is the normalising constant. ESS is interpreted as the number of i.i.d. samples from the target distribution $p_{t+1}(x_{t+1})$ that would be required to obtain an estimator with the same variance as the SIS estimator. Since $ESS \leq$

n (Kong et al., 1994), then an ESS value close to n indicates that the SIS sample of size n is approximately as "good" as an i.i.d. sample of size n from $p_{t+1}(x_{t+1})$. In practice, a fixed threshold is chosen (in general, half of the sample size n), and if the ESS falls below that threshold, then a resampling step is performed.

A word of caution is required at this point. Via the ESS approach, we're using the importance sampling weights to evaluate how well the weighted sample $\left\{x_{t+1}^{(i)}, w_{t+1}^{(i)}\right\}_{i=1}^{n}$ approximates the target distribution $p_{t+1}(x_{t+1})$. It is possible that the instrumental distribution matches poorly the target distribution, but the weights are similar in value, and thus have small variance. The ESS would then be large, thus incorrectly indicating a good match between the instrumental and target distributions. This, for example, could happen if the target distribution places most of its mass on a small region where the instrumental distribution is flat, and the target distribution is flat in the region of the mode of the instrumental distribution. Hence, we expect sampling from this instrumental distribution to result in weights that are similar in value. With small probability, a draw would fall under the mode of the target distribution, resulting in a strikingly different weight value. This example highlights the importance of sampling a large enough number of particles such that all modes of the target distribution are explored via sampling from the instrumental distribution. So choosing an appropriate sample size n depends on knowing the shape of the target distribution, which, of course, is not known; in practice, we use as many particles as computationally feasible.

8.3.2 SISR for state-space models

Figure 3 presents the SISR algorithm with multinomial sampling for state-space models.

Algorithm 3 The SISR algorithm for state-space models

```
1: Set $t = 1$.
2: For $i = 1 : n$, sample $x_1^{(i)} \sim q_1(x_1)$.
3: For $i = 1 : n$, set $w_1^{(i)} \propto p_1(x_1^{(i)})g(y_1|x_1^{(i)})/q_1(x_1^{(i)})$. Normalise such that $\sum_{j=1}^{n} w_1^{(j)} = 1$.
4: At time $t + 1$, do:
5: Resample step: compute $ESS = 1/\sum_{j=1}^{n}(w_t^{(j)})^2$.
6: If $ESS < threshold$, then resample: for $i = 1 : n$, set $\tilde{x}_{1:t}^{(i)} = x_{1:t}^{(j)}$ with probability $w_t^{(j)}$, $j = 1, \ldots, n$. Finally, for
   $i = 1 : n$, set $x_{1:t}^{(i)} = \tilde{x}_{1:t}^{(i)}$ and $w_t^{(i)} = 1/n$.
7: For $i = 1 : n$, sample $x_{t+1}^{(i)} \sim q_{t+1}(x_{t+1}|x_{1:t}^{(i)})$.
8: For $i = 1 : n$, set $w_{t+1}^{(i)} \propto w_t^{(i)} f(x_{t+1}^{(i)}|x_t^{(i)})g(y_{t+1}|x_{t+1}^{(i)})/q_{t+1}(x_{t+1}^{(i)}|x_t^{(i)})$. Normalise such that $\sum_{i=1}^{n} w_{t+1}^{(i)} = 1$.
9: The filtering and smoothing densities at time $t + 1$ may be approximated by
   $$\hat{p}(x_{t+1}|y_{1:t+1}) = \sum_{i=1}^{n} w_{t+1}^{(i)} \delta_{x_{t+1}^{(i)}}(x_{t+1}), \quad \hat{p}(x_{1:t+1}|y_{1:t+1}) = \sum_{i=1}^{n} w_{t+1}^{(i)} \delta_{x_{1:t+1}^{(i)}}(x_{1:t+1}).$$
10: Set $t = t + 1$. Go to step 4.
```

For the SISR algorithm for state-space models, Crisan and Doucet (2002) prove that the empirical distributions converge to their true values almost surely as $n \to \infty$, under weak regularity conditions. Furthemore, they show convergence of the mean square error for bounded, measurable functions, provided that the weights are upper bounded; moreover, the rate of convergence is proportional to $1/n$. However, only under restrictive assumptions can they prove that approximation errors do not accumulate over time, so careful implementation and interpretation is required when dealing with SMC methods. More generally, Chopin (2004) proves a Central Limit Theorem result for the SISR algorithm under both multinomial sampling, and residual sampling, not restricted to applications to state-space models.

Example 8.2 (A nonlinear time series model (Cappé et al., 2007)). Consider the following nonlinear time series model:

$$\begin{aligned} y_t &= \frac{x_t^2}{20} + v_t \sim g(y_t|x_t) \\ x_t &= \frac{x_{t-1}}{2} + 25\frac{x_{t-1}}{1+x_{t-1}^2} + 8\cos(1.2t) + u_t \sim f(x_t|x_{t-1}), \end{aligned}$$

where $v_t \sim \mathsf{N}(0, \sigma_v^2)$, $u_t \sim \mathsf{N}(0, \sigma_u^2)$, and parameters $\sigma_v^2 = 1$, $\sigma_u^2 = 10$. Let $x_1 \sim p(x_1) = \mathsf{N}(0, 10)$. The densities are

$$\begin{aligned} f(x_t|x_{t-1}) &= \mathsf{N}\left(\frac{x_{t-1}}{2} + 25\frac{x_{t-1}}{1+x_{t-1}^2} + 8\cos(1.2t), 10\right) \\ g(y_t|x_t) &= \mathsf{N}\left(\frac{x_t^2}{20}, 1\right). \end{aligned}$$

◁

Figure 8.2 shows 100 states x_t and corresponding observations y_t generated from this model.

Using these 100 observations, we begin by running the SISR algorithm until $t = 9$, with $n = 10000$ particles and resampling whenever $ESS < 0.6 \times n$. The instrumental distribution is the state transition distribution: $q_{t+1}(x_{t+1}|x_{1:t}^{(i)}) = f(x_{t+1}|x_t^{(i)})$. Figure 8.3 shows the weighted samples $\left\{x_9^{(i)}, w_9^{(i)}\right\}_{i=1}^{n}$ as small dots (with weights unnormalised), and the ***kernel density estimate*** of the filtering distribution as a continuous line. Kernel density estimation is a non-parameteric approach to estimating the density of a random variable from a (possibly weighted) sample of values. For details, see Silverman (1986). We use a Gaussian kernel with fixed width of 0.5. The kernel density estimator takes into account both the value of the weights, and the local density of the particles.

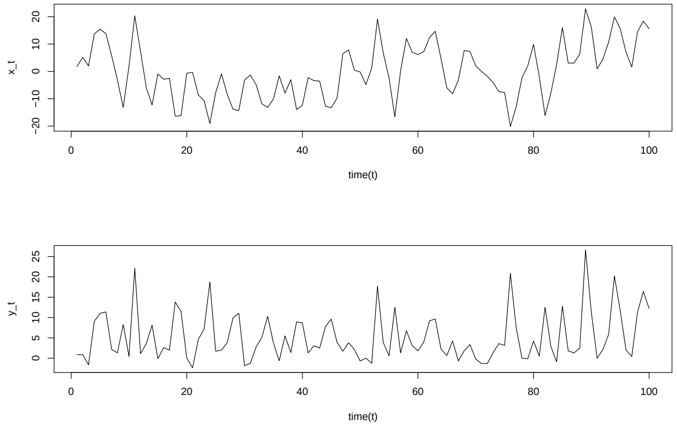


Figure 8.2. Plot of 100 observations y_t and hidden states x_t generated from the above state-space model.

To analyse the effect of resampling, we run the SISR algorithm up to time $t = 100$ with $n = 10000$ and resampling whenever $ESS < 0.6 \times n$, and the SIS algorithm with $n = 10000$. Figures 8.4 and 8.5 show the image intensity plots of the kernel density estimates based on the filter outputs, with the true state sequence overlaid. In general, the true state value falls in the high density regions of the density estimate in Figure 8.4, indicating good performance of the SISR algorithm. Moreover, it is interesting to notice that there is clear evidence of multimodality and non-Gaussianity. In Figure 8.5, however, we remark that the particle distributions are highly degenerate, and do not track the correct state sequence. Hence, resampling is required for good performance of the SIS algorithm.

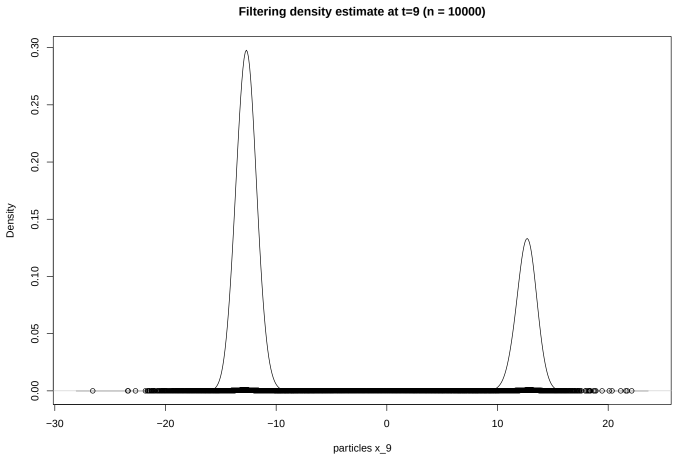


Figure 8.3. Filtering density estimate at $t = 9$ from SISR algorithm with $n = 10000$, and ESS $threshold = 6000$. Weighted samples $\left\{x_9^{(i)}, w_9^{(i)}\right\}_{i=1}^{n}$ shown as small dots, and kernel density estimate as continuous line.

To make this last point more clear, we look at histograms of the base 10 logarithm of the normalised weights at various time steps in the SIS algorithm. Figure 8.6 shows that the weights quickly degenerate as t increases; by $t = 5$, we already observe weights on the order of 10^{-300}. Recall that these weights are a measure of the adequacy of the simulated trajectory, drawn from an instrumental distribution, to the target distribution. In this particular example, the instrumental distribution is the state transition distribution, which is highly variable ($\sigma_u^2 = 10$) compared to the observation distribution ($\sigma_v^2 = 1$). Hence, draws from the state transition distribution are given low weights under the observation distribution, and, with no resampling to eliminate the particles with very low weights, there is a quickly growing accumulation of very low weights.

8.4 Sequential Importance Sampling with Resampling and MCMC moves

Gilks and Berzuini (2001) introduce the idea of using MCMC moves to reduce sample impoverishment, and call their proposed algorithm *resample-move*. The algorithm performs sequential importance resampling with an MCMC move after the resampling step that rejuvenates the particles. Let $\left\{x_{1:t+1}^{(i)}, w_{t+1}^{(i)}\right\}_{i=1}^{n}$ be a weighted set of particles that targets the distribution $p_{t+1}(x_{1:t+1})$. Let $q_{t+1}(x_{1:t+1})$ denote the instrumental distribution from which the particles are generated. During resampling, some particles will be replicated (possibly many times), to produce the set $\left\{\tilde{x}_{1:t+1}^{(i)}, 1/n\right\}_{i=1}^{n}$. Let $\mathbf{K}_{t+1}$ be a $p_{t+1}(x_{1:t+1})$-invariant Markov kernel, i.e., $p_{t+1}\mathbf{K}_{t+1} = p_{t+1}$. The MCMC move is as follows: for $i = 1, \ldots, n$, draw $z_{1:t+1}^{(i)} \sim \mathbf{K}_{t+1}(\tilde{x}_{1:t+1}^{(i)}, \cdot)$. Then $\left\{z_{1:t+1}^{(i)}, 1/n\right\}_{i=1}^{n}$ is a rejuvenated, weighted set of particles that targets the distribution $p_{t+1}(z_{1:t+1})$. If $\left\{\tilde{x}_{1:t+1}^{(i)}, 1/n\right\}_{i=1}^{n}$ is a good particle representation of $p_{t+1}(x_{1:t+1})$, then, provided that the kernel $\mathbf{K}_{t+1}$ is fast mixing, each $\tilde{x}_{1:t+1}^{(i)}$ will tend to move to a distinct point in a high density region of the target distribution, thus improving the particle representation. In the words of Gilks and Berzuini (2001), the MCMC step helps the particles track the moving target.

Just as in Section 8.1.1 we interpreted rejection sampling as importance sampling on an enlarged space, so can importance sampling with an MCMC move be interpreted as importance sampling on an enlarged space with instrumental distribution $q_{t+1}(x_{1:t+1})\mathbf{K}_{t+1}(x_{1:t+1}, z_{1:t+1})$ and target distribution $p_{t+1}(x_{1:t+1})\mathbf{K}_{t+1}(x_{1:t+1}, z_{1:t+1})$,

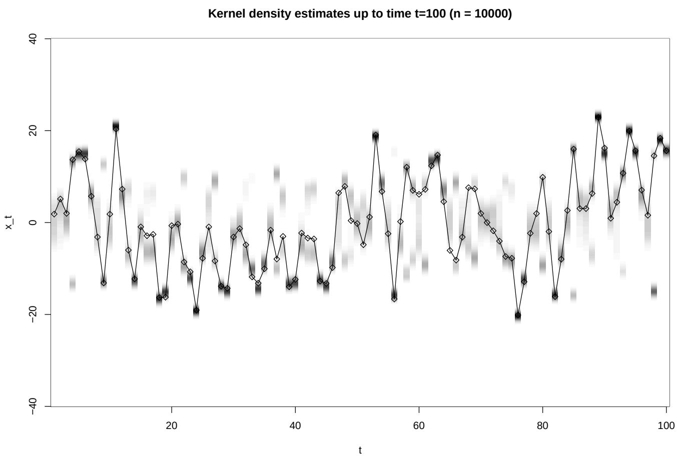


Figure 8.4. Image intensity plot of the kernel density estimates up to $t = 100$, with diamond symbol indicating the true state sequence (SISR algorithm).

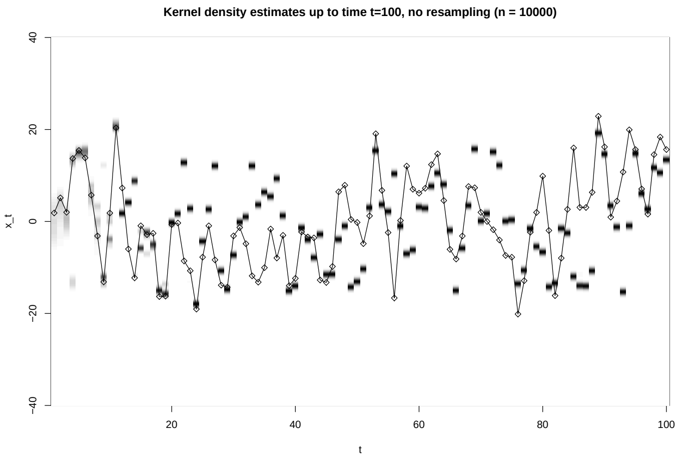


Figure 8.5. Image intensity plot of the kernel density estimates up to $t = 100$, with diamond symbol indicating the true state sequence (SIS algorithm).

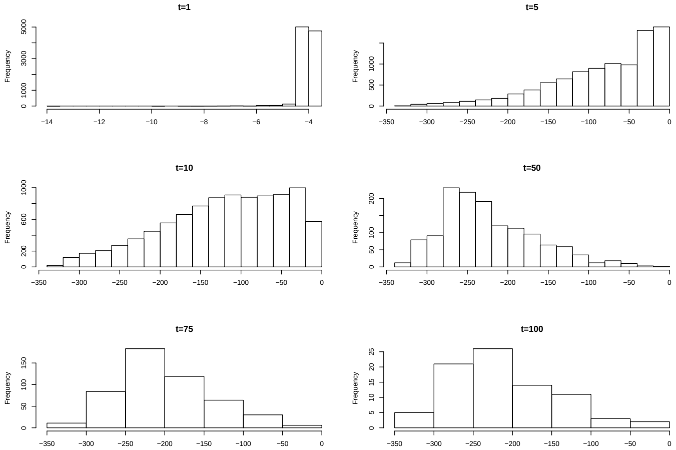


Figure 8.6. Histogram of the base 10 logarithm of the normalised weights for the filtering distributions (SIS algorithm).

where $\int p_{t+1}(x_{1:t+1})\mathbf{K}_{t+1}(x_{1:t+1}, z_{1:t+1})dx_{1:t+1} = p_{t+1}(z_{1:t+1})$ by invariance of $\mathbf{K}_{t+1}$. This explanation omits the resampling step, but the latter does not alter the justification of the algorithm. Gilks and Berzuini (2001) present a Central Limit Theorem result in the number of particles, and an explicit formulation for the asymptotic variance as the number of particles tends to infinity, for t fixed. They argue that, in the extreme case, rejuvenation via a perfectly mixing kernel at step t, i.e., $\mathbf{K}_t(x_{1:t}, z_{1:t}) = p_t(z_{1:t}|x_{1:t})$, can reduce the asymptotic variance of estimators at later time steps. This is similar to the idea from Section 8.3 that resampling, although it introduces extra Monte Carlo variation at the current time step, can reduce the variance at later times.

Chopin (2004) states that MCMC moves may lead to more stable algorithms for the filtering problem in terms of the asymptotic variance of estimators (although theoretical results to support this are lacking); however, he is not as hopeful regarding the smoothing problem. He suggests investigating the degeneracy of a given particle filter algorithm by running m, say $m = 10$, independent particle filters in parallel, computing the estimates from each output, and monitoring the empirical variance of these m estimates as t increases.

8.4.1 SISR with MCMC moves for state-space models

Algorithm 4 is the SISR algorithm with MCMC moves for state-space models. First, notice that except for the requirement of invariance, there are no constraints on the choice of kernel; it can be a Gibbs sampling, or a Metropolis-Hastings kernel. Second, only one MCMC step will suffice (i.e., no burn-in period is required), since it is assumed that the particle set at step t has "converged", i.e., it is a good representation of $p(x_{1:t}|y_{1:t})$. Third, notice that the MCMC move is applied to the entire state trajectory up to time t. Hence, the dimension of the kernel increases with t, thus increasing the computational cost of performing the move. Also, as t increases, it is increasingly difficult to construct a fast-mixing Markov kernel of dimension t. In practice, the MCMC move is applied only to the last component x_{t+1} with kernel $\mathbf{K}_{t+1}$ that is invariant with respect to $p(x_{t+1}|y_{1:t+1})$.

Algorithm 4 The SISR algorithm with MCMC moves for state-space models

```
1: Set $t = 1$.
2: For $i = 1 : n$, sample $x_1^{(i)} \sim q_1(x_1)$.
3: For $i = 1 : n$, set $w_1^{(i)} \propto p_1(x_1^{(i)})g(y_1|x_1^{(i)})/q_1(x_1^{(i)})$. Normalise such that $\sum_{j=1}^{n} w_1^{(j)} = 1$.
4: At time $t + 1$, do:
5: Resample step: compute $ESS = 1/\sum_{j=1}^{n}(w_t^{(j)})^2$.
6: If $ESS < threshold$, then resample: for $i = 1 : n$, set $\tilde{x}_{1:t}^{(i)} = x_{1:t}^{(j)}$ with probability $w_t^{(j)}$, $j = 1, \ldots, n$. Finally, for
   $i = 1 : n$, set $x_{1:t}^{(i)} = \tilde{x}_{1:t}^{(i)}$ and $w_t^{(i)} = 1/n$.
7: MCMC step: for $i = 1 : n$, sample $z_{1:t}^{(i)} \sim \mathbf{K}_t(x_{1:t}^{(i)}, \cdot)$. Then, for $i = 1 : n$, set $x_{1:t}^{(i)} = z_{1:t}^{(i)}$.
8: For $i = 1 : n$, sample $x_{t+1}^{(i)} \sim q_{t+1}(x_{t+1}|x_{1:t}^{(i)})$.
9: For $i = 1 : n$, set $w_{t+1}^{(i)} \propto w_t^{(i)} f(x_{t+1}^{(i)}|x_t^{(i)}) g(y_{t+1}|x_{t+1}^{(i)})/q_{t+1}(x_{t+1}^{(i)}|x_t^{(i)})$. Normalise such that $\sum_{i=1}^{n} w_{t+1}^{(i)} = 1$.
10: The filtering and smoothing densities at time $t + 1$ may be approximated by
    $$\hat{p}(x_{t+1}|y_{1:t+1}) = \sum_{i=1}^{n} w_{t+1}^{(i)} \delta_{x_{t+1}^{(i)}}(x_{t+1}), \quad \hat{p}(x_{1:t+1}|y_{1:t+1}) = \sum_{i=1}^{n} w_{t+1}^{(i)} \delta_{x_{1:t+1}^{(i)}}(x_{1:t+1}).$$
11: Set $t = t + 1$. Go to step 4.
```

8.5 Smoothing density estimation

As we have seen, the SISR algorithm is prone to suffer from sample impoverishment as t grows; hence the particle trajectories do not offer reliable approximations to the smoothing density. This section presents a Monte Carlo smoothing algorithm for state-space models that is carried out in a forward-filtering, backward-smoothing procedure, i.e., a filtering procedure such as SISR is applied forward in time, followed by a smoothing procedure applied backward in t (Godsill et al., 2004).

Godsill et al. (2004) consider the joint smoothing density

$$\begin{aligned} p(x_{1:T}|y_{1:T}) &= p(x_T|y_{1:T}) \prod_{t=1}^{T-1} p(x_t|x_{t+1:T}, y_{1:T}) = p(x_T|y_{1:T}) \prod_{t=1}^{T-1} p(x_t|x_{t+1}, y_{1:t}) \\ &\propto p(x_T|y_{1:T}) \prod_{t=1}^{T-1} p(x_t|y_{1:t}) f(x_{t+1}|x_t). \end{aligned}$$

Let $\left\{x_t^{(i)}, w_t^{(i)}\right\}_{i=1}^{n}$ be a particle representation to the filtering density $p(x_t|y_{1:t})$, that is, $\hat{p}(x_t|y_{1:t}) = \sum_{i=1}^{n} w_t^{(i)} \delta_{x_t^{(i)}}(x_t)$. Then, it is possible to approximate $p(x_t|x_{t+1}, y_{1:t})$ by

$$\hat{p}(x_t|x_{t+1}, y_{1:t}) = \frac{\sum_{i=1}^{n} w_t^{(i)} f(x_{t+1}|x_t^{(i)}) \delta_{x_t^{(i)}}(x_t)}{\sum_{j=1}^{n} w_t^{(j)} f(x_{t+1}|x_t^{(j)})}. \tag{8.4}$$

So the idea is to run the particle filter (e.g., the SISR algorithm) forward in time, to obtain particle approximations to $p(x_t|y_{1:t})$, for $t = 1, \ldots, T$, and then to apply the backward smoothing recursion (8.4) for $t = T - 1$ to $t = 1$. The draws $(\tilde{x}_1, \ldots, \tilde{x}_T)$ form an approximate realization from $p(x_{1:T}|y_{1:T})$.

Algorithm 5 returns one realization from $p(x_{1:T}|y_{1:T})$ via this forward-filtering, backward-smoothing approach. The computational complexity is $O(nT)$, for filtering density approximations with n particles. So for n realizations, the computational complexity is $O(n^2T)$, i.e., it is quadratic in n, compared to the computational complexity of SISR, for n realizations, which is linear in n. This smoothing algorithm has the advantage that it uses particle approximations to the filtering densities (under the assumption that good approximations can be obtained via SISR, for example), as opposed to resampled particle trajectories $x_{1:t}$ which suffer from sample impoverishment as t increases.

Example 8.3 (A nonlinear time series model (continued)). We continue the example of the nonlinear time series model. We carry out smoothing via Algorithm 5, implementing the SISR algorithm as before, with $n = 10000$

Algorithm 5 The forward-filtering, backward-smoothing algorithm for state-space models

```
1: Run a particle filter algorithm to obtain weighted particle approximations $\left\{x_t^{(i)}, w_t^{(i)}\right\}_{i=1}^n$ to the filtering distributions $p(x_t|y_{1:t})$ for $t = 1, \ldots, T$.
2: Set $t = T$.
3: Choose $\tilde{x}_T = x_T^{(i)}$ with probability $w_T^{(i)}$.
4: Set $t = t - 1$.
5: At $t \geq 1$
6: For $i = 1 : n$, compute $w_{t|t+1}^{(i)} \propto w_t^{(i)} f(\tilde{x}_{t+1}|x_t^{(i)})$. Normalise the weights.
7: Choose $\tilde{x}_t = x_t^{(i)}$ with probability $w_{t|t+1}^{(i)}$.
8: Go to step 3.
9: $(\tilde{x}_1, \ldots, \tilde{x}_T)$ is an approximate realization from $p(x_{1:T}|y_{1:T})$.
```

particles. Figure 8.7 displays the 10000 smoothing trajectories drawn from $p(x_{1:100}|y_{1:100})$ with the true state sequence overlaid. Multimodality in the smoothing distributions is shown in Figure 8.8.

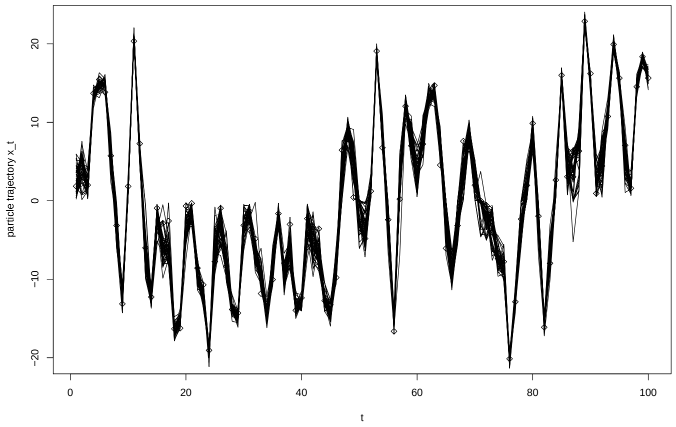


Figure 8.7. 10000 smoothing trajectories drawn from $p(x_{1:100}|y_{1:100})$ with dimond symbol indicating the true sequence.

Finally, since the algorithm returns entire smoothing trajectories, as opposed to simply returning smoothing marginals, it is possible to visualize characteristics of the multivariate smoothing distribution. Figure 8.9 shows the kernel density estimate for $p(x_{11:12}|y_{1:100})$.

◁

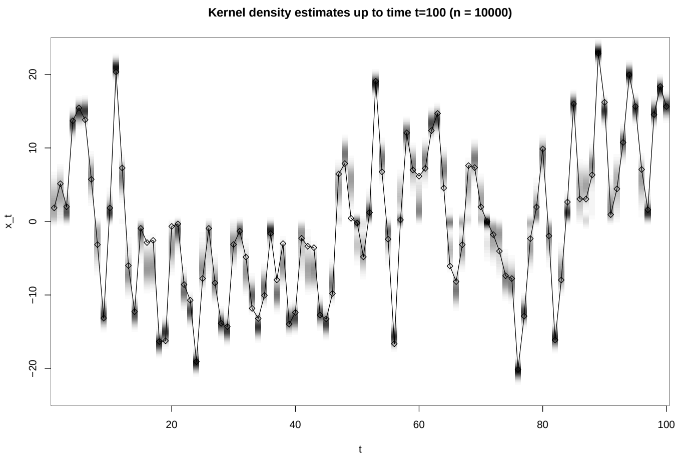


Figure 8.8. Image intensity plot of the kernel density estimates of smoothing densities with dimond symbol indicating the true state sequence.

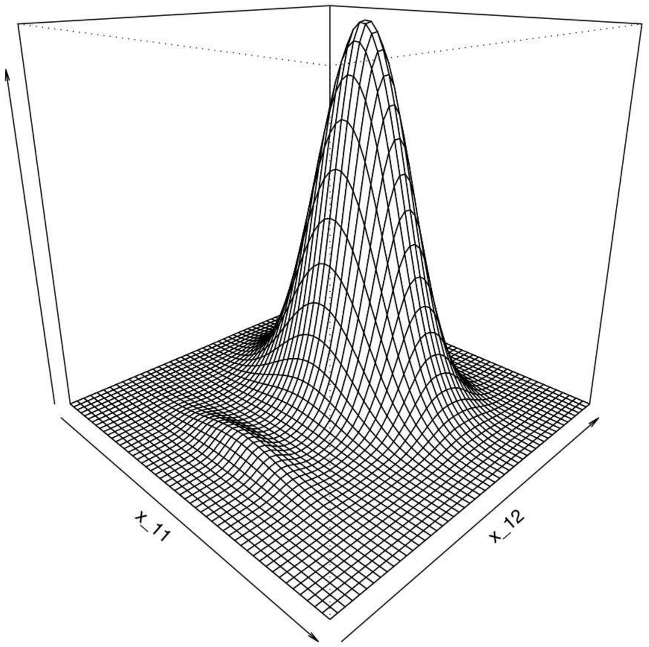


Figure 8.9. Kernel density estimate for $p(x_{11:12}|y_{1:100})$.