Stochastic Simulation

Ulm University
Institute of Stochastics

Lecture Notes
Dr. Tim Brereton
revised by Dr. Kirsten Schorning
Summer Term 2017

Ulm, 2017

Contents

Chapter 1

Discrete-time Markov Chains

1.1 Discrete-Time Markov Chains

In this course, we will consider techniques for simulating many interesting (and often complicated) random objects. In doing so, we will review the mathematical theory that makes these techniques work. (Arguably) the most important tools we will use are *discrete-time Markov chains*. As we will learn, these will allow us to simulate an incredibly large number of random objects. Discrete-time Markov chains are a particular type of *discrete-time stochastic process* with a number of very useful features.

Definition 1.1.1 (Discrete-Time Stochastic Process)**.** A discrete-time stochastic process with *state space* $\mathcal{X}$ is a collection of $\mathcal{X}$-valued random variables $\{X_n\}_{n \in \mathbb{N}}$.

In this course, I will take $\mathbb{N}$ to be the set of natural numbers including 0. In general, as the term "discrete-time" implies, n will represent a point in time. However, it could also represent a point in space for example.

Definition 1.1.2 (Discrete-Time Markov Chain)**.** A discrete-time Markov chain is a discrete-time stochastic process, $\{X_n\}_{n \in \mathbb{N}}$, with a *countable* state space, $\mathcal{X}$, that satisfies

$$\mathbb{P}(X_n = j \mid X_0 = i_0, \ldots, X_{n-1} = i_{n-1}) = \mathbb{P}(X_n = j \mid X_{n-1} = i_{n-1})$$

for all $n \geq 1$ and events $\{X_0 = i_0, \ldots, X_{n-1} = i_{n-1}\}$ with

$$\mathbb{P}(X_0 = i_0, \ldots, X_{n-1} = i_{n-1}) > 0.$$

If $\mathbb{P}(X_n = j \mid X_{n-1} = i)$ does not depend on n for all $i, j \in \mathcal{X}$, we say $\{X_n\}_{n \in \mathbb{N}}$ is *time homogenous*. In this course, unless otherwise stated, we will assume that Markov chains are time homogenous.

Example 1.1.3 (Random walk on the corners of a square)**.** Consider a random walker that jumps around on the corners of a square (pictured in Figure 1.1.1). At each time step, it selects one of the adjacent corners (uniformly) and jumps to it.

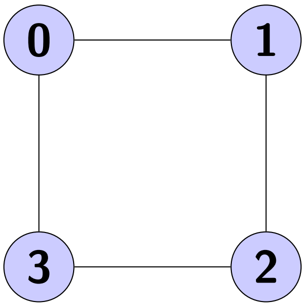

Figure 1.1.1: State space of the random walk, showing possible transitions.

The state space of this Markov chain is $\mathcal{X} = \{0, 1, 2, 3\}$. The transition probabilities are given by

$$\mathbb{P}(X_n = j \mid X_{n-1} = i) = \begin{cases} 1/2 & \text{if } j = i + 1 \mod 4, \\ 1/2 & \text{if } j = i - 1 \mod 4, \\ 0 & \text{otherwise.} \end{cases}$$

Example 1.1.4 (A jumping flea)**.** A flea lives in a house with 3 dogs. Each day it either stays where it is (with probability 1/2) or jumps (with probability 1/2) to one of the other dogs (selected uniformly).

The state space of this Markov chain is $\mathcal{X} = \{0, 1, 2\}$. The transition probabilities are given by

$$\mathbb{P}(X_n = j \mid X_{n-1} = i) = \begin{cases} 1/2 & \text{if } j = i, \\ 1/4 & \text{if } j = i + 1 \mod 3, \\ 1/4 & \text{if } j = i - 1 \mod 3. \end{cases}$$

Example 1.1.5. Random walk on the integers Consider a random walker taking integer values. At each turn, it goes up one with probability 1/2 and goes down one with probability 1/2.

The state space of this Markov chain is $\mathcal{X} = \mathbb{Z}$ and the transition probabilities are given by

$$\mathbb{P}(X_n = j \mid X_{n-1} = i) = \begin{cases} 1/2 & \text{if } j = i + 1, \\ 1/2 & \text{if } j = i - 1, \\ 0 & \text{otherwise.} \end{cases}$$

Definition 1.1.6. Let $\{X_n\}_{\in \mathbb{N}}$ be a discrete-time homogenous Markov chain. The $|\mathcal{X}| \times |\mathcal{X}|$ matrix P defined by

$$\mathbb{P}(X_n = j \mid X_{n-1} = i) = (P)_{i,j} \quad \forall i, j \in \mathcal{X},$$

is called *transition matrix* .

Properties of transition matrices

In order for P to be a transition matrix, it must satisfy three properties:

(i) It is real-valued (i.e., $P \in \mathbb{R}^{|\mathcal{X}| \times |\mathcal{X}|}$).

(ii) It is non-negative (i.e., $(P)_{i,j} \geq 0$ for all $i, j \in \mathcal{X}$).

(iii) It has rows that sum to one (i.e. $\sum_{j \in \mathcal{X}} (P)_{i,j} = 1$ for all $i \in \mathcal{X}$).

A matrix that satisfies these properties is called a *stochastic matrix.*

A transition matrix can be represented by a directed weighted graph with vertex set $\mathcal{X}$ and an edge (with weight $(P)_{i,j}$) placed between $i \in \mathcal{X}$ and $j \in \mathcal{X}$ if and only if $(P)_{i,j} > 0$. Such a graph is called a *transition graph.*

Example 1.1.7 (Example 1.1.3 continued: random walk on the corners of a square)**.** The transition matrix of this Markov chain is

$$P = \left[\begin{array}{cccc} 0 & 1/2 & 0 & 1/2 \\ 1/2 & 0 & 1/2 & 0 \\ 0 & 1/2 & 0 & 1/2 \\ 1/2 & 0 & 1/2 & 0 \end{array}\right].$$

The corresponding transition graph is illustrated in Figure 1.1.2.

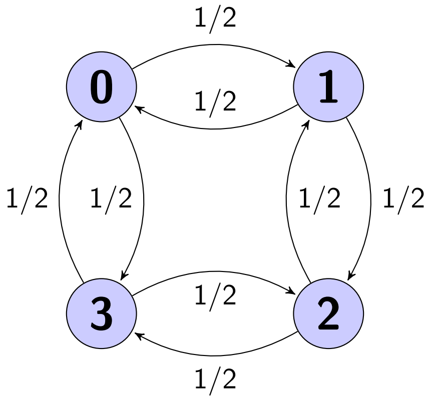

Figure 1.1.2: Transition graph of the random walk on the corners of the square.

Example 1.1.8 (Example 1.1.4 continued: a jumping flea)**.** The transition matrix of this Markov chain is

$$P = \left[\begin{array}{ccc} 1/2 & 1/4 & 1/4 \\ 1/4 & 1/2 & 1/4 \\ 1/4 & 1/4 & 1/2 \end{array}\right].$$

The corresponding transition graph is illustrated in Figure 1.1.3.

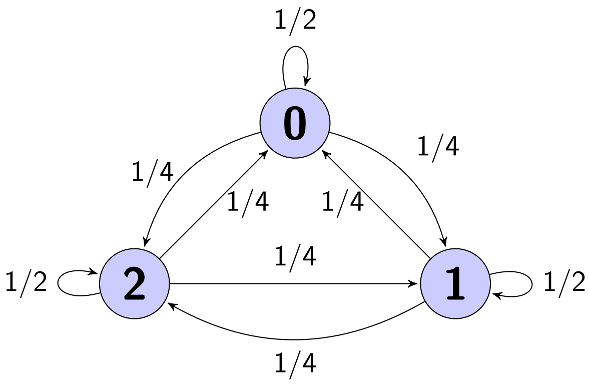

Figure 1.1.3: Transition graph of the Markov chain describing the flea hopping from dog to dog.

Using a transition matrix, we are able to describe the dynamics of a Markov chain (i.e. how it moves at each step). There is one more piece of information we need, however, in order to completely characterize a discrete-time Markov chain $\{X_n\}_{n\in\mathbb{N}}$: we need to describe how it begins.

Definition 1.1.9 (Measure (on a countable space))**.** We say a $|\mathcal{X}|$-dimensional (row) vector, $\boldsymbol{\lambda}$, is a *measure* on a countable state space $\mathcal{X}$ if

$$(\boldsymbol{\lambda})_i \geq 0 \text{ for all } i \in \mathcal{X}.$$

Given a set $A \subset \mathcal{X}$ and a measure $\boldsymbol{\lambda}$, we can calculate the measure of the set by

$$\boldsymbol{\lambda}(A) = \sum_{i\in A} (\boldsymbol{\lambda})_i.$$

Definition 1.1.10 (Distribution (on a countable space))**.** We say a measure, $\boldsymbol{\mu}$, on a countable state space $\mathcal{X}$, is a *distribution* if

$$\sum_{i\in\mathcal{X}} (\boldsymbol{\mu})_i = 1.$$

We will say a discrete-time Markov chain, $\{X_n\}_{n\in\mathbb{N}}$ is Markov$(\boldsymbol{\mu}, P)$ if it has transition matrix P and $X_0 \sim \boldsymbol{\mu}$ (i.e. $\mathbb{P}(X_0 = i) = (\boldsymbol{\mu})_i$). In particular, we will often want to start the Markov chain at a specific value (with probability 1). In order to do this, we define the $|\mathcal{X}|$-dimensional vector $\boldsymbol{\delta}_i$, where

$$(\boldsymbol{\delta}_i)_j = \begin{cases} 1 & \text{if } j = i, \\ 0 & \text{otherwise.} \end{cases}$$

If $X_0 \sim \boldsymbol{\delta}_i$ then $\mathbb{P}(X_0 = i) = 1$.

In these notes, we will write $\mathbb{P}_{\boldsymbol{\mu}}$ and $\mathbb{E}_{\boldsymbol{\mu}}$ for probabilities and expectations given the Markov chain starts from $\boldsymbol{\mu}$. We will write $\mathbb{P}_i$ and $\mathbb{E}_i$ for probabilities and expectations given the Markov chain starts from state i (i.e. $X_0 \sim \boldsymbol{\delta}_i$).

Theorem 1.1.11. A discrete-time stochastic process, $\{X_n\}_{n\in\mathbb{N}}$ is Markov$(\boldsymbol{\mu}, P)$ if and only if

$$\mathbb{P}(X_0 = i_0, X_1 = i_1, \ldots, X_n = i_n) = (\boldsymbol{\mu})_{i_0}(P)_{i_0,i_1} \cdots (P)_{i_{n-1},i_n}$$

for all $i_0, \ldots, i_n \in \mathcal{X}$ such that $\mathbb{P}(X_0 = i_0, \ldots, X_n = i_{n-1}) > 0$.

Proof. Assuming that $\{X_n\}_{n\in\mathbb{N}}$ is Markov $(\boldsymbol{\mu}, P)$, observe that

$$\begin{aligned}&\mathbb{P}(X_0 = i_0, X_1 = i_1, \ldots, X_n = i_n)\\ &= \mathbb{P}(X_0 = i_0)\mathbb{P}(X_1 \,|\, X_0 = i_0) \times \cdots\\ &\times \mathbb{P}(X_{n-1} \,|\, X_0 = i_0, \ldots, X_{n-2} = i_{n-2})\mathbb{P}(X_n \,|\, X_0 = i_0, \ldots, X_{n-1} = i_{n-1}).\end{aligned}$$

Because the chain is Markov $(\boldsymbol{\mu}, P)$, the first probability is an entry in $\boldsymbol{\mu}$ and all the conditional probabilities are just entries in the transition matrix. Thus, we can write

$$\mathbb{P}(X_0 = i_0, X_1 = i_1, \ldots, X_n = i_n) = (\boldsymbol{\mu})_{i_0}(P)_{i_0,i_1} \cdots (P)_{i_{n-1},i_n}$$

In the converse direction, we recall that (because P is a stochastic matrix) $\sum_{j\in\mathcal{X}}(P)_{i,j} = 1$ for all $i \in \mathcal{X}$. Thus, we have

$$\begin{aligned}&\mathbb{P}(X_0 = i_0, X_1 = i_1, \ldots, X_{n-1} = i_{n-1})\\ &= \sum_{i_n\in\mathcal{X}} (\boldsymbol{\mu})_{i_0}(P)_{i_0,i_1} \cdots (P)_{i_{n-1},i_n}\\ &= (\boldsymbol{\mu})_{i_0}(P)_{i_0,i_1} \cdots (P)_{i_{n-2},i_{n-1}}\end{aligned}$$

We can continue in this way to recover all the probabilities up to

$$\mathbb{P}(X_0 = i_0) = (\boldsymbol{\mu})_{i_0},$$

which proves that $X_0 \sim \boldsymbol{\mu}$. In order to demonstrate that $\{X_n\}_{n\in\mathbb{N}}$ is Markov with transition matrix P, observe that

$$\begin{aligned}&\mathbb{P}(X_n = i_n \,|\, X_0 = i_0, \ldots, X_{n-1} = i_{n-1})\\ &= \frac{\mathbb{P}(X_0 = i_0, X_1 = i_1, \ldots, X_n = i_n)}{\mathbb{P}(X_0 = i_0, X_1 = i_1, \ldots, X_{n-1} = i_{n-1})}\\ &= \frac{(\boldsymbol{\mu})_{i_0}(P)_{i_0,i_1} \cdots (P)_{i_{n-1},i_n}}{(\boldsymbol{\mu})_{i_0}(P)_{i_0,i_1} \cdots (P)_{i_{n-2},i_{n-1}}}\\ &= (P)_{i_{n-1},i_n}.\end{aligned}$$

Note that this implies that

$$\mathbb{P}(X_n = i_n \,|\, X_0 = i_0, \ldots, X_{n-1} = i_{n-1}) = \mathbb{P}(X_n = i_n \,|X_{n-1} = i_{n-1}),$$

because

$$\begin{aligned}
&\mathbb{P}(X_n = i_n \,| X_{n-1} = i_{n-1}) \\
&= \sum_{i_0,\dots,i_{n-2}\in\mathcal{X}} \Big[\mathbb{P}(X_n = i_n \,|\, X_0 = i_0, \dots, X_{n-1} = i_{n-1}) \\
&\quad \times \mathbb{P}(X_0 = i_0, \dots X_{n-2} = i_{n-2} \,|\, X_{n-1} = i_{n-1})\Big] \\
&= (P)_{i_{n-1},i_n} \sum_{i_0,\dots,i_{n-2}\in\mathcal{X}} \mathbb{P}(X_0 = i_0, \dots X_{n-2} = i_{n-2} \,|\, X_{n-1} = i_{n-1}) \\
&= (P)_{i_{n-1},i_n}.
\end{aligned}$$

□

Theorem 1.1.12 (Markov property)**.** Let $\{X_n\}_{n\in\mathbb{N}}$ be Markov $(\boldsymbol{\mu}, P)$. Then, conditional on $\{X_m = i\}$, $\{X_{m+n}\}_{n\geq 0}$ is Markov $(\boldsymbol{\delta}_i, P)$ and is independent of the random variables $X_0, \dots, X_m$.

Proof. If we show that, for any event A determined by $X_0, \dots, X_m$,

$$\begin{aligned}
&\mathbb{P}(\{X_m = i_m, \dots, X_{m+n} = i_{m+n}\} \cap A \,|\, X_m = i) \\
&\quad = (\boldsymbol{\delta}_i)_{i_m}(P)_{i_m,i_{m+1}}, \dots, (P)_{i_{m+n-1},i_{m+n}}\mathbb{P}(A \,|\, X_m = i),
\end{aligned} \tag{1.1}$$

then we have shown the result as Theorem 1.1.11 shows that $\{X_{m+n}\}_{n\in\mathbb{N}}$ is Markov $(\boldsymbol{\delta}_i, P)$ and the fact that the probability is a product demonstrates the independence.

In order to show (1.1) is true, we first show it for simple events of the form $A_k = \{X_0 = i_{0_k}, \dots, X_m = i_{m_k}\}$. In such cases, repeatedly using the definition of a Markov chain, we have that

$$\begin{aligned}
&\mathbb{P}(X_0 = i_{0_k}, \dots X_m = i_{m_k}, X_{m+1} = i_{m+1}, \dots, X_{m+n} = i_{m+n} \,|\, X_m = i_m) \\
&= \mathbb{P}(X_m = i_m, \dots, X_{m+n} = i_{m+n} \,|\, X_0 = i_{0_k}, \dots, X_{m-1} = i_{m-1}, X_m = i) \\
&\quad \times \mathbb{P}(X_0 = i_{0_k}, \dots X_m = i_{m_k} \,|\, X_m = i) \\
&= \mathbb{P}(X_m = i_m \,|\, X_0 = i_{0_k}, \dots, X_{m-1} = i_{m-1}, X_m = i) \\
&\quad \times \mathbb{P}(X_{m+1} = i_{m+1} \,|\, X_0 = i_{0_k}, \dots, X_{m-1} = i_{m-1}, X_m = i) \\
&\quad \times \cdots \times \mathbb{P}(X_{m+n} = i_{m+n} \,|\, X_0 = i_{0_k}, \dots, X_m = i, \dots X_{m+n-1} = i_{m+n-1}) \\
&\quad \times \mathbb{P}(X_0 = i_{0_k}, \dots X_m = i_{m_k} \,|\, X_m = i) \\
&= (\boldsymbol{\delta}_i)_{i_m}(P)_{i_m,i_{m+1}} \cdots (P)_{i_{m+n-1},i_{m+n}}\mathbb{P}(A_k \,|\, X_m = i),
\end{aligned}$$

As any event A that only depends on $X_0, \dots, X_m$ can be written as a union of disjoint A_k, the result follows. □

1.2 Random Mappings and Simulation

1.2.1 Random Mapping Representations

The key to simulating a Markov chain is (usually) to find a suitable *random mapping representation*. This is a representation of a Markov chain that allows us to simulate it using a stream of *independent and identically distributed* (iid) random variables.

Definition 1.2.1 (Random mapping representation)**.** A random mapping representation of a transition matrix P on state space $\mathcal{X}$ is a function $f : \mathcal{X} \times \Lambda \to \mathcal{X}$ and a Λ-valued random variable Z such that

$$\mathbb{P}(f(i,Z) = j) = (P)_{i,j} \text{ for all } i,j \in \mathcal{X}.$$

The following theorem shows that we can construct a Markov chain with the desired transition matrix and initial distribution using a random mapping representation.

Theorem 1.2.2. Let f and Z be a random mapping representation of the transition matrix P. Then, given a sequence of iid random variables $\{Z_n\}_{n\geq 1}$ with $Z_n \overset{D}{=} Z$, the sequence $\{X_n\}_{n\in\mathbb{N}}$ defined by $X_0 \sim \boldsymbol{\mu}$, where X_0 is independent of the $\{Z_n\}_{n\geq 1}$, and

$$X_n = f(X_{n-1}, Z_n) \text{ for all } n \geq 1,$$

is Markov $(\boldsymbol{\mu}, P)$.

Proof. That $\{X_n\}_{n\in\mathbb{N}}$ has initial distribution $\boldsymbol{\mu}$ follows from the definition. Now, observe that

$$\begin{aligned}
&\mathbb{P}(X_n = i_n \,|\, X_0 = i_0, \dots, X_{n-1} = i_{n-1}) \\
&\quad = \mathbb{P}(f(X_{n-1}, Z_n) = i_n \,|\, X_0 = i_0, \dots, X_{n-1} = i_{n-1}) \\
&\quad = \mathbb{P}(f(i_{n-1}, Z_n) = i_n \,|\, X_0 = i_0, \dots, X_{n-1} = i_{n-1}) \\
&\quad = \mathbb{P}(f(i_{n-1}, Z_n) = i_n) \\
&\quad = \mathbb{P}(f(i_{n-1}, Z) = i_n) \\
&\quad = (P)_{i_{n-1}, i_n},
\end{aligned}$$

where we use the fact that Z_n is independent of $X_0, \dots, X_{n-1}$ and that the $\{Z_n\}_{n\geq 1}$ are iid with the same distribution as Z. □

The next theorem tell us that given a transition matrix, P, on a countable state space $\mathcal{X}$, we can always find a random mapping representation.

Theorem 1.2.3. Every transition matrix has a random mapping representation.

Proof. Consider an arbitrary transition matrix P on a countable state space $\mathcal{X}$. Let $\Lambda = (0,1)$ and $Z \sim \mathcal{U}(0,1)$. Define the values $\{F_{i,j}\}_{i,j\in\mathcal{X}}$ by

$$F_{i,j} = \sum_{k=1}^{j} (P)_{i,k}.$$

Observe that $0 \leq F_{i,j} \leq 1$ for all $i,j \in \mathcal{X}$ and that $F_{i,j} \leq F_{i,j+1}$ for all $i \in \mathcal{X}$ and j such that $j, j+1 \in \mathcal{X}$. Define the function $f : \mathcal{X} \times (0,1) \to \mathcal{X}$ by

$$f(i,z) = j \text{ such that } F_{i,j-1} < z \leq F_{i,j}.$$

Then

$$\mathbb{P}(f(i,Z) = j) = \mathbb{P}(F_{i,j-1} < Z \leq F_{i,j}) = F_{i,j} - F_{i,j-1} = (P)_{i,j}.$$

□

Theorem 1.2.3 is in some sense a proof that for any transition matrix, P, a Markov chain with the specified transition probabilities exists (after all, we can easily cook up a space of iid uniform random variables).

It is important to note that random mapping representations are not unique. Consider the following example.

Example 1.2.4 (A random mapping representation of a random walk on the integers)**.** Consider the random walk defined in Example 1.1.5. We can construct a random mapping representation of it using the random variable $Z \sim \mathsf{Ber}(1/2)$ and the function

$$f(i,z) = \begin{cases} i+1 & \text{if } z = 1, \\ i-1 & \text{if } z = 0, \\ 0 & \text{otherwise.} \end{cases}$$

1.2.2 Simulation

The proof of Theorem 1.2.3 gives a generic recipe for simulating Markov chains on finite spaces (on infinite spaces another random mapping representation is usually needed – see, e.g., Example 1.2.4). The basic algorithm is as follows:

Algorithm 1.2.1 (Simulating a Markov chain on a finite state space)**.**

(i) Draw $X_0 \sim \boldsymbol{\mu}$.

(ii) Draw $U \sim \mathcal{U}(0,1)$. Set $X_{n+1} = j$ where j is such that

$$\sum_{k=1}^{j-1} (P)_{i,k} < U \leq \sum_{k=1}^{j} (P)_{i,k}.$$

(iii) Set $n = n+1$ and repeat from step 2.

The code below demonstrates how to implement this in Matlab.

Example 1.2.5 (Example 1.1.3 continued: random walk on the corners of a square)**.** We can simulate the random walk on the corners of a square as follows.

Listing 1.1: Simulating a random walk on the corners of a square

```
n = 10
X= c()
P= rbind(c(0,0.5,0,0.5),c(0.5,0,0.5,0),c(0,0.5,0,0.5),c(0.5,0,0.5,0))
mu=rep(1/4, 4)
X_0= sample(1:4, size=1, prob= mu)
Z=runif(n=1, min=0, max=1)
X[1]=min(which(Z<=cumsum(P[X_0,]) ))
for(i in 1:(n-1))
{
  Z= runif(n=1, min=0, max=1)
  X[i+1]= min(which(Z<=cumsum(P[X[i],]) ))
}

```

```
# plot of sample
par(mfrow=c(1,1))
plot(1:n, X , xlim=c(0, n), type="n", ylim=c(0, 4), xlab="n",
ylab=expression(X[n]))
segments(x0=0, y0= X_0, x1=1, y1=X_0, lwd=3)
for(i in 1:(n-1))
{
  segments(x0=i, y0= X[i], x1=i+1, y1=X[i], lwd=3)
}
```

Example 1.2.6 (Example 1.1.5 continued: random walk on the integers)**.** Using the random mapping representation given in Example 1.2.4, we can simulate a random walk on the integers as follows.

Listing 1.2: Simulating a random walk on the integers

```
n=20
X=c()
p=0.5  # transition probability
X_0 = 0 # starting point
Z= rbinom(n=1,size=1, prob=p) # sample of bernoulli distributed variable
X[1]= X_0 + 1*Z + 1*(Z-1)   # X= i+1 if Z=1 and X=i-1 if Z=0
for(i in 1:(n-1))           # continue for i=2, ... , n
{
  Z= rbinom(n=1, size=1, prob=0.5)
  X[i+1]= X[i] + 1*Z + 1*(Z-1)
}
# plot
par(mfrow=c(1,1))
plot(1:n, X , xlim=c(0, n), type="n", ylim=c(min(X), max(X)), xlab="n",
ylab=expression(X[n]))
segments(x0=0, y0= X_0, x1=1, y1=X_0, lwd=3)
for(i in 1:(n-1))
{
  segments(x0=i, y0= X[i], x1=i+1, y1=X[i], lwd=3)
}
```

1.3 The distribution of X_n

We now know how to calculate a number of probabilities. In particular,

(i) $\mathbb{P}(X_0 = i)$

(ii) $\mathbb{P}(X_n = i_n \,|\, X_0 = i_0, \dots, X_{n-1} = i_{n-1})$

(iii) $\mathbb{P}(X_n = i_n \,|\, X_{n-1} = i_{n-1})$

(iv) $\mathbb{P}(X_0 = i_0, \dots, X_n = i_n)$.

Finding the distribution of X_n for a fixed $n \in \mathbb{N}$ is a little bit more work.

To answer this we first consider the *n-step probability*

$$(P^{(n)})_{i,j} = \mathbb{P}(X_n = j | X_0 = i)$$

if $\mathbb{P}(X_0 = i) > 0$.
$P^{(n)} = (P^{(n)})_{i,j \in \mathcal{X}}$ is called the n-step transition matrix of the Markov chain $\{X_n\}_{n \in \mathbb{N}}$.

Theorem 1.3.1. The equation $P^{(n)} = P^n$ holds for arbitrary $n = 0, 1, 2, \ldots$ and thus for arbitrary $n, m = 0, 1, 2, \ldots$

$$P^{(n+m)} = P^{(n)} P^{(m)}.$$

Corollary 1.3.2. For arbitrary $n, m, r = 0, 1, 2, \ldots$ and $i, j, k \in \mathcal{X}$

$$P_{i,i}^{(n+m)} \geq P_{i,j}^{(n)} P_{j,i}^{(m)}$$
$$P_{i,i}^{(n+m+r)} \geq P_{i,j}^{(n)} P_{j,k}^{(m)} P_{k,i}^{(r)}$$

Theorem 1.3.3. Let $\{X_n\}_{n \in \mathbb{N}}$ be Markov $(\boldsymbol{\mu}, P)$. Then, for $n \in \mathbb{N}$,

$$\mathbb{P}(X_n = j) = (\boldsymbol{\mu} P^n)_j.$$

Proof. Use the n-step probability matrix.

$$\begin{aligned}
\mathbb{P}(X_n = j) &= \sum_{i \in \mathcal{X}} \mathbb{P}(X_0 = i) \mathbb{P}(X_n = j \,|\, X_0 = i) \\
&= \sum_{i \in \mathcal{X}} P_{i,j}^{(n)} (\boldsymbol{\mu})_i \\
&= (\boldsymbol{\mu} P^{(n)})_j.
\end{aligned}$$

□

Thus, when $\{X_n\}_{n \in \mathbb{N}}$ is Markov$(\boldsymbol{\mu}, P)$, the distribution of X_n is given by

$$X_n \sim \boldsymbol{\mu} P^n.$$

Note that this is not always so nice to work out by hand but, so long as $|\mathcal{X}|$ is not too big, it can be easily computed on a computer.

Example 1.3.4 (Calculating the distribution of X_n for Example 1.1.3)**.** In Matlab, it is straightforward to take powers of matrices. The following code does it for Example 1.1.3 with an initial distribtuion $\boldsymbol{\mu} = (1, 0, 0, 0)$.

Listing 1.3: Calculating the distribution of X_n for a random walk on the corners of a square

```
matpot= function(M, n)
{
  result= diag(rep(1,nrow(M)))
  for(i in 1:n)
  {
    result= result %*% M
  }
  return(result)
}
P= rbind(c(0,0.5,0,0.5),c(0.5,0,0.5,0),c(0,0.5,0,0.5),c(0.5,0,0.5,0))
n= c(2, 5, 100, 101, 102)

mu1=c(1, 0, 0, 0) # starting in corner 1
DIST1 = matrix(0, nrow=length(n), ncol= nrow(P))
for(i in 1:length(n))
{
  DIST1[i,]= mu1%*% matpot(M=P, n=n[i])
}
DIST1=cbind(n, DIST1)

mu2= c(0, 1, 0, 0) # starting in corner 2
DIST2= matrix(0, nrow=length(n), ncol= nrow(P))
for(i in 1:length(n))
{
  DIST2[i,]= mu2%*% matpot(M=P, n=n[i])
}
DIST2=cbind(n, DIST2)

mu3= c(1/2, 1/2, 0, 0) # starting in corner 1 or 2
DIST3= matrix(0, nrow=length(n), ncol= nrow(P))
for(i in 1:length(n))
{
  DIST3[i,]= mu3%*% matpot(M=P, n=n[i])
}
DIST3=cbind(n, DIST3)

mu4= c(1/4, 1/4, 1/4, 1/4) # starting randomly in one corner
DIST4= matrix(0, nrow=length(n), ncol= nrow(P))
for(i in 1:length(n))
{
  DIST4[i,]= mu4%*% matpot(M=P, n=n[i])
}
DIST4=cbind(n, DIST4)
```

Let us look at the output of this code for a number of different choices of the initial distribution $\boldsymbol{\mu}$. For $\boldsymbol{\mu} = (1, 0, 0, 0)$ we have

$$
\begin{aligned}
X_2 &\sim (1/2, 0, 1/2, 0)\\
X_5 &\sim (0, 1/2, 0, 1/2)\\
X_{100} &\sim (1/2, 0, 1/2, 0)\\
X_{101} &\sim (0, 1/2, 0, 1/2)\\
X_{102} &\sim (1/2, 0, 1/2, 0).
\end{aligned}
$$

For $\boldsymbol{\mu} = (0, 1, 0, 0)$ we have

$$\begin{aligned}
X_2 &\sim (0, 1/2, 0, 1/2)\\
X_5 &\sim (1/2, 0, 1/2, 0)\\
X_{100} &\sim (0, 1/2, 0, 1/2)\\
X_{101} &\sim (1/2, 0, 1/2, 0)\\
X_{102} &\sim (0, 1/2, 0, 1/2).
\end{aligned}$$

For $\boldsymbol{\mu} = (1/2, 1/2, 0, 0)$ we have

$$\begin{aligned}
X_2 &\sim (1/4, 1/4, 1/4, 1/4)\\
X_5 &\sim (1/4, 1/4, 1/4, 1/4)\\
X_{100} &\sim (1/4, 1/4, 1/4, 1/4)\\
X_{101} &\sim (1/4, 1/4, 1/4, 1/4)\\
X_{102} &\sim (1/4, 1/4, 1/4, 1/4).
\end{aligned}$$

Lastly, for $\boldsymbol{\mu} = (1/4, 1/4, 1/4, 1/4)$ we have

$$\begin{aligned}
X_2 &\sim (1/4, 1/4, 1/4, 1/4)\\
X_5 &\sim (1/4, 1/4, 1/4, 1/4)\\
X_{100} &\sim (1/4, 1/4, 1/4, 1/4)\\
X_{101} &\sim (1/4, 1/4, 1/4, 1/4)\\
X_{102} &\sim (1/4, 1/4, 1/4, 1/4).
\end{aligned}$$

Example 1.3.5 (Calculating the distribution of X_n for Example 1.1.4)**.** The following code calculates the distribution of X_n for Example 1.1.4.

Listing 1.4: Calculating the distribution of X_n for a jumping flea

```r
P= rbind(c(0.5,0.25,0.25),c(0.25,0.5,0.25),c(0.25,0.25,0.5))
n= c(2, 5, 100, 101, 102)

mu1=c(1, 0, 0) # starting on dog 1
DIST1 = matrix(0, nrow=length(n), ncol= nrow(P))
for(i in 1:length(n))
{
  DIST1[i,]= mu1%*% matpot(M=P, n=n[i])
}
DIST1=cbind(n, DIST1)

mu2= c(0, 1, 0) # starting on dog 2
DIST2= matrix(0, nrow=length(n), ncol= nrow(P))
for(i in 1:length(n))
{
  DIST2[i,]= mu2%*% matpot(M=P, n=n[i])
}
DIST2=cbind(n, DIST2)

```

```
mu3= c(0,1/3,2/3) # starting on dog 2 or 3
DIST3= matrix(0, nrow=length(n), ncol= nrow(P))
for(i in 1:length(n))
{
  DIST3[i,]= mu3%*% matpot(M=P, n=n[i])
}
DIST3=cbind(n, DIST3)

mu4= c(1/3,1/3,1/3) # starting on dog 1, 2, 3
DIST4= matrix(0, nrow=length(n), ncol= nrow(P))
for(i in 1:length(n))
{
  DIST4[i,]= mu4%*% matpot(M=P, n=n[i])
}
DIST4=cbind(n, DIST4)
```

Let us again look at the output of this code for a number of different choices of the initial distribution $\boldsymbol{\mu}$. For $\boldsymbol{\mu} = (1, 0, 0)$ we have (at least approximately)

$$\begin{aligned} X_2 &\sim (3/8, 5/16, 5/16) \\ X_5 &\sim (171/512, 341/1024, 341/1024) \\ X_{100} &\sim (1/3, 1/3, 1/3) \\ X_{101} &\sim (1/3, 1/3, 1/3) \\ X_{102} &\sim (1/3, 1/3, 1/3). \end{aligned}$$

For $\boldsymbol{\mu} = (0, 1, 0)$ we have (at least approximately)

$$\begin{aligned} X_2 &\sim (5/16, 3/8, 5/16) \\ X_5 &\sim (341/1024, 171/512, 341/1024) \\ X_{100} &\sim (1/3, 1/3, 1/3) \\ X_{101} &\sim (1/3, 1/3, 1/3) \\ X_{102} &\sim (1/3, 1/3, 1/3). \end{aligned}$$

For $\boldsymbol{\mu} = (0, 1/3, 2/3)$ we have (at least approximately)

$$\begin{aligned} X_2 &\sim (5/16, 1/3, 17/48) \\ X_5 &\sim (341/1024, 1/3, 342/1025) \\ X_{100} &\sim (1/3, 1/3, 1/3) \\ X_{101} &\sim (1/3, 1/3, 1/3) \\ X_{102} &\sim (1/3, 1/3, 1/3). \end{aligned}$$

And lastly, for $\boldsymbol{\mu} = (1/3, 1/3, 1/3)$,we have

$$\begin{aligned} X_2 &\sim (1/3, 1/3, 1/3) \\ X_5 &\sim (1/3, 1/3, 1/3) \\ X_{100} &\sim (1/3, 1/3, 1/3) \\ X_{101} &\sim (1/3, 1/3, 1/3) \\ X_{102} &\sim (1/3, 1/3, 1/3). \end{aligned}$$

Looking at the distributions computed in Examples 1.3.4 and 1.3.5, we can observe a number of things.

(i) In Example 1.3.4, if one starts with probability 1 in either $\{0\}$ or $\{2\}$, then the distribution of X_n is $(1/2, 0, 1/2, 0)$ for even times and $(0, 1/2, 0, 1/2)$ for odd times. If one starts with probability 1 in either $\{1\}$ or $\{3\}$, the distribution of X_n is $(0, 1/2, 0, 1/2)$ for even times and$(1/2, 0, 1/2, 0)$ for odd times. Thus, the distribution of X_n seems to be *periodic* for certain choices of initial distribution.

(ii) In both examples, there are distributions — $(1/4, 1/4, 1/4, 1/4)$ and $(1/3, 1/3, 1/3)$ — where, if the chain is started in this distribution, X_n has the same distribution for all $n \geq 0$.

(iii) In Example 1.3.5 there is a distribution — $(1/3, 1/3, 1/3)$ — to which $\{X_n\}_{n\in\mathbb{N}}$ seems to converge, no matter what its initial distribution is.

This leads to a number of natural questions:

(i) When does a Markov chain have an initial distribution, $\boldsymbol{\pi}$, that does not change as n increases (in other words, when can we find a $\boldsymbol{\pi}$ such that $\boldsymbol{\pi} = \boldsymbol{\pi}P$)?

(ii) If such a $\boldsymbol{\pi}$ exists, when is it unique?

(iii) Given a unique $\boldsymbol{\pi}$, when does a Markov chain converge to it (if it starts from an initial distribution that is not $\boldsymbol{\pi}$?

In order to answer these questions, we need to be able to classify a number of different types of Markov chains.

1.4 Class Structure

An important question to ask, when dealing with a Markov chain, is if it is possible to move from any state to any other state.

Definition 1.4.1. We say $i \in \mathcal{X}$ *leads* to $j \in \mathcal{X}$ $(i \to j)$ if there exists an $n \in \mathbb{N}$ with

$$P^n_{i,j} = \mathbb{P}_i(X_n = j | X_0 = i) > 0.$$

We say $i \in \mathcal{X}$ *communicates* with $j \in \mathcal{X}$ $(i \leftrightarrow j)$ if $i \to j$ and $j \to i$.

The relation $\leftrightarrow$ is an *equivalence relation* because it is

(i) *reflexive*: $i \leftrightarrow i$ (set $n = 0$),

(ii) *symmetric*: $i \leftrightarrow j$ if and only if $j \leftrightarrow i$ (by definition),

(iii) *transitive*: $i \leftrightarrow j$ and $j \leftrightarrow k$ implies $i \leftrightarrow k$, use extension of Corollary 1.3.2.

Using the equivalence relation $\leftrightarrow$ we can partition $\mathcal{X}$ into disjoint equivalence classes, called *communicating classes*. Note that the communicating classes of a Markov chain are determined by its transition matrix, P, and do not depend on the initial distribution $\boldsymbol{\mu}$.

Example 1.4.2. Consider the Markov chain with transition matrix

$$P = \begin{pmatrix} 0 & 1/2 & 1/2 \\ 0 & 1/2 & 1/2 \\ 0 & 1 & 0 \end{pmatrix}$$

Its transition graph is given in Figure 1.4.1.

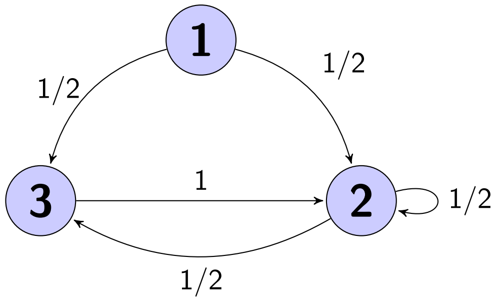

Figure 1.4.1: Transition graph of a Markov chain with two communicating classes.

The transition matrix P has two communicating classes $C_1 = \{1\}$ and $C_2 = \{2,3\}$. This is because $2 \leftrightarrow 3$ but 1 does not communicate with the other two.

Theorem 1.4.3. *For $i, j \in \mathcal{X}$ with $i \neq j$ the following are equivalent*

(i) $i \to j$,

(ii) $(P)_{i_0,i_1}(P)_{i_1,i_2}\cdots(P)_{i_{n-1},i_n} > 0$ *for some* $i_0, i_1, \ldots, i_n \in \mathcal{X}$ *with* $i_0 = i$ *and* $i_n = j$.

Proof. We have $i \to j$, then there exists an $n \in \mathbb{N}$ such that

$$(P^n)_{i,j} > 0.$$

We also have that

$$(P^n)_{i,j} = \sum_{i_1,\ldots,i_{n-1}\in\mathcal{X}} P_{i,i_1}\cdot\ldots P_{i_{n-1},i_n}$$

so that $(i) \Leftrightarrow (ii)$. □

Definition 1.4.4. A communicating class, C, is *closed* if $i \in C$ and $i \to j$ implies that $j \in C$. In other words, if the chain is in C, it cannot leave C.

In Example 1.4.2, C_2 is a closed class and C_1 is not.

Definition 1.4.5 (Irreducibility)**.** A transition matrix, P, is called *irreducible* if $\mathcal{X}$ is a single communicating class.

Example 1.4.6 (An irreducible Markov chain)**.** Consider the Markov chain with transition matrix

$$P = \begin{pmatrix} 1/4 & 3/4 & 0 \\ 1/2 & 0 & 1/2 \\ 1 & 0 & 0 \end{pmatrix}$$

Its transition graph is given in Figure 1.4.2.

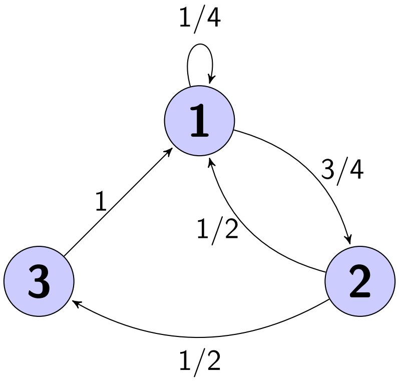

Figure 1.4.2: Transition graph of an irreducible Markov chain.

The transition matrix P is irreducible, as it is possible to get from any state to any other state (though not always in a single step).

Example 1.4.7 (A Markov chain that is not irreducible)**.** Consider the Markov chain with transition matrix

$$P = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1/2 & 1/2 \\ 0 & 1/2 & 1/2 \end{pmatrix}$$

Its transition graph is given in Figure 1.4.3.

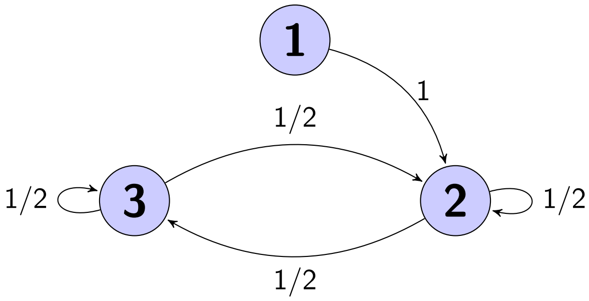

Figure 1.4.3: Transition graph of a Markov chain that is not irreducible.

The transition matrix P is not irreducible, as it is not possible to get from $\{2\}$ to $\{1\}$ (i.e., there is more than one communicating class).

1.5 Stopping Times and the Strong Markov Property

It is often useful to consider the times at which various events involving a discrete-time stochastic process $\{X_n\}_{n\in\mathbb{N}}$ occur. For example, we might like to know the first time that $\{X_n\}$ hits a certain state. Because $\{X_n\}_{n\in\mathbb{N}}$ is random process, such times will also be random. We usually require that the random times we consider satisfy certain technical conditions. Such random times are called *stopping times*.

Definition 1.5.1 ((Discrete-time) Stopping Time)**.** A $\mathbb{N}$-valued random variable, τ, is a (discrete-time) stopping time if and only if for all $n \geq 0$ there is a set $A_n \in \mathcal{X}^{n+1}$ such that $\{\tau = n\} = \{(X_0, \dots, X_n) \in A_n\}$. In other words, τ is a stopping time if and only if $\mathbb{I}(\tau = n)$ is a function of $(X_0, \dots, X_n)$

Intuitively, a random time τ is a stopping time if and only if we can determine at time n whether $\tau = n$ or not. Two important stopping times are the following:

(i) The *hitting time* of a set $A \subset \mathcal{X}$ is given by

$$\tau_A^H = \inf\{n \geq 0 : X_n \in A\}.$$

(ii) The *first passage time* into state i is given by

$$\tau_i^F = \inf\{n \geq 1 : X_n = i\}.$$

Note that, in both cases, we will take $\inf\{\emptyset\} = \infty$.

Note that, if the chain starts in a state i then the hitting time $\tau_{\{i\}}^H$ will be 0, while the first passage time will be strictly large than 0.

Example 1.5.2 (Random times that are not stopping times)**.** The random time $\tau = \inf\{n \geq 0 : X_{n+1} = i\}$ is not a stopping time, because the event $\{\tau = n\}$ depends on $(X_0, \dots, X_{n+1})$ rather than just $(X_0, \dots, X_n)$. Likewise, the *last exit time* of a set $A \subset \mathcal{X}$, $\tau_A^L = \sup\{n \geq 0 : X_n \in A\}$, is not, in general, a stopping time. This is because we may not have any way of knowing at time n whether $\{X_n\}_{n\in\mathbb{N}}$ will return to A or not.

The following theorem is an analogue of the Markov property for times that are random (rather than the deterministic times considered in Theorem 1.1.12).

Theorem 1.5.3. *Let $i \in \mathcal{X}$ be such that $\mathbb{P}(X_0 = i) > 0$. $i \to j$ if and only if $\mathbb{P}(\tau_j^F < \infty | X_0 = i) > 0$.*

Proof. 1. Let $i \to j$ that means that there exists $n \in \mathbb{N}$: $P_{i,j}^n > 0$.
It is:

$$\{X_n = j\} \subset \{\tau_j^F \leq n\} \subset \{\tau_j^F < \infty\}$$

and thus

$$\mathbb{P}(\tau_j^F < \infty | X_0 = i) \geq \mathbb{P}(X_n = j | X_0 = i) = P_{i,j}^n > 0.$$

2. Let $i \nrightarrow j$ that means for all $n \geq 0$ $P^n_{i,j} = 0$.

$$\begin{aligned}
\mathbb{P}(\tau^F_j < \infty | X_0 = i) &= \lim_{n\to\infty} \mathbb{P}(\tau^F_j < n | X_0 = i) \\
&= \lim_{n\to\infty} \mathbb{P}(\cup_{k=1}^{n-1} \{X_k = j\} | X_0 = i) \\
&\leq \lim_{n\to\infty} \sum_{k=1}^{n-1} \mathbb{P}(X_k = j | X_0 = i) \\
&= \lim_{n\to\infty} \sum_{k=1}^{n-1} P^k_{i,j} = 0
\end{aligned}$$

□

Theorem 1.5.4 (Strong Markov Property)**.** Let $\{X_n\}_{n\in\mathbb{N}}$ be Markov $(\boldsymbol{\mu}, P)$ and τ be a stopping time of $\{X_n\}_{n\in\mathbb{N}}$. Then, conditional on $\{\tau < \infty\}$ and $\{X_\tau = i\}$, $\{X_{\tau+n}\}_{n\in\mathbb{N}}$ is Markov$(\boldsymbol{\delta}_i, P)$ and independent of $X_0, X_1, \ldots, X_\tau$.

Proof. Let A be an event determined by $X_0, \ldots, X_\tau$. Then $A \cap \{\tau = m\}$ is determined by $X_0, \ldots, X_m$. By the Markov property (Theorem 1.1.12), we have that

$$\begin{aligned}
&\mathbb{P}(\{X_\tau = j_0, \ldots, X_{\tau+n} = j_n\} \cap A \cap \{\tau = m\} \cap \{X_\tau = i\}) \\
&= \mathbb{P}_i(\{X_0 = j_0, \ldots, X_n = j_n\})\mathbb{P}(A \cap \{\tau = m\} \cap \{X_\tau = i\})
\end{aligned}$$

Then, summing over m, we have

$$\begin{aligned}
&\mathbb{P}(\{X_\tau = j_0, \ldots, X_{\tau+n} = j_n\} \cap A \cap \{X_\tau = i\}) \\
&= \mathbb{P}_i(\{X_0 = j_0, \ldots, X_n = j_n\})\mathbb{P}(A \cap \{X_\tau = i\})
\end{aligned}$$

Dividing by $\mathbb{P}(\{\tau < \infty\} \cap \{X_\tau = i\})$ we have

$$\begin{aligned}
&\mathbb{P}(\{X_\tau = j_0, \ldots, X_{\tau+n} = j_n\} \cap A \,|\, \tau < \infty, X_\tau = i) \\
&= \mathbb{P}_i(\{X_0 = j_0, \ldots, X_n = j_n\})\mathbb{P}(A \cap \{X_\tau = i\})
\end{aligned}$$

□

1.6 Recurrence

We have already seen that Markov chains can be divided into those that are irreducible and those that are not. Another important property is what is called *recurrence.*

Definition 1.6.1. Let $\{X_n\}_{n\in\mathbb{N}}$ be a Markov chain with transition matrix P. A state $i \in \mathcal{X}$ is *recurrent* if

$$\mathbb{P}_i(X_n = i \text{ for infinitely many } n) = 1.$$

We say a state $i \in \mathcal{X}$ is *transient* if

$$\mathbb{P}_i(X_n = i \text{ for infinitely many } n) = 0.$$

Notation
$V_i = \sum_{n=0}^{\infty} \mathbb{1}(X_n = i)$ is the number of visits to i.

$f_i = \mathbb{P}(\tau_i^F < \infty | X_0 = i)$ is the probability to return to i.
Thus: $i \in \mathcal{X}$ is recurrent if and only if $\mathbb{P}(V_i = \infty | X_0 = i) = 1$ and ($i \in \mathcal{X}$ is transient if $\mathbb{P}(V_i = \infty | X_0 = i) < 1$).

Lemma 1.6.2. For all $k \geq 0$ $\mathbb{P}(V_i \geq k+1 | X_0 = i) = (f_i)^k$.

Proof. Induction principle:
Basis: $k = 0$ ✓
Inductive hypothesis: For $0, 1, \ldots, k-1$ it is $\mathbb{P}(V_i \geq k | X_0 = i) = f_i^{k-1}$
Inductive step: $k-1 \to k$

$$\begin{aligned}
\mathbb{P}(V_i \geq k+1 | X_0 = i) &= \mathbb{P}(V_i \geq k+1 | V_i \geq k, X_0 = i)\mathbb{P}(V_i \geq k | X_0 = i) \\
&= \mathbb{P}(\tau_i^F < \infty | X_0 = i)(f_i)^{k-1} \\
&= f_i (f_i)^{k-1} = (f_i)^k.
\end{aligned}$$

□

Theorem 1.6.3. We have that:

(i) $i \in \mathcal{X}$ is recurrent if and only if $f_i = 1$.

(ii) $i \in \mathcal{X}$ is transient if and only if $f_i < 1$.

Proof. Observe

$$\begin{aligned}
\mathbb{P}(V_i < \infty | X_0 = i) &= \mathbb{P}(\cup_{k \geq 1} \{V_i = k\} | X_0 = i) \\
&= \sum_{k=1}^{\infty} \mathbb{P}(V_i = k | X_0 = i) \\
&= \sum_{k=1}^{\infty} (1 - f_i) f_i^{k-1} \\
&= \begin{cases} 0 & f_i = 1 \\ 1 & f_i < 1 \end{cases}
\end{aligned}$$

where we use that V_i is geometrically distributed with probability f_i.
So $i \in \mathcal{X}$ is recurrent if $f_i = 1$ and transient if $f_i < 1$. □

Theorem 1.6.4. For a given $i \in \mathcal{X}$, we have

(i) i is recurrent if and only if $\sum_{n=0}^{\infty} (P^n)_{i,i} = \infty$.

(ii) i is transient if and only if $\sum_{n=0}^{\infty} (P^n)_{i,i} < \infty$.

Proof. 1. $i \in \mathcal{X}$ is recurrent $\leftrightarrow \mathbb{P}(V_i = \infty | X_0 = i) = 1$.

$$\begin{aligned}\sum_{n=0}^{\infty} P_{i,j}^n &= \sum_{n=0}^{\infty} \mathbb{E}[\mathbb{1}(X_n = i) | X_0 = i] && (1.2)\\ &= \mathbb{E}[\sum_{n=0}^{\infty} \mathbb{1}(X_n = i) | X_0 = i] && (1.3)\\ &= \mathbb{E}[V_i | X_0 = i] = \infty && (1.4)\end{aligned}$$

2. $i \in \mathcal{X}$ is transient $\leftrightarrow f_i < \infty$.

$$\begin{aligned}\sum_{n=0}^{\infty} P_{i,j}^n &= \mathbb{E}[V_i | X_0 = i] && (1.5)\\ &= \sum_{r=0}^{\infty} \mathbb{P}(V_i > r | X_0 = i) && (1.6)\\ &= \sum_{r=0}^{\infty} f_i^r = \frac{1}{1 - f_i} < \infty && (1.7)\end{aligned}$$

where we again use that V_i is geometrically distributed with probability f_i. □

The following theorem shows that recurrence (or transience) is a class property. That is, the states in a communicating class are either all recurrent or all transient.

Theorem 1.6.5. For a transition matrix P, we have the following:

(i) Let C be a communicating class. Then, either all states in C are transient or all states are recurrent.

(ii) Every recurrent class is closed.

(iii) Every finite closed class is recurrent.

Proof. Only (i), transient. For the rest see J. Norris (1997) Markov Chains.
Take any pair of states $i, j \in C$ and suppose $i \in C$ is transient.
Since C is a communicating class there exists $n, m \geq 0$ with $(P^n)_{i,j}) > 0$ and $(P^m)_{j,i}) > 0$. With Corollary 1.3.2 for $r \geq 0$ we get:

$$(P^{n+m+r})_{i,i} \geq (P^n)_{i,j} (P^r)_{j,j} (P^m)_{j,i}$$

so that

$$\sum_{r=0}^{\infty} P_{j,j}^r \leq \frac{1}{P_{i,j}^n P_{j,i}^m} \sum_{r=0}^{\infty} P_{i,i}^{n+m+r} < \infty$$

since i is transient. □

Theorem 1.6.6. Let P be irreducible and recurrent. Then, for all $j \in \mathcal{X}$, $\mathbb{P}(\tau_j^F < \infty) = 1$.

Proof. We have that

$$\mathbb{P}(\tau_j^F < \infty) = \sum_{i \in \mathcal{X}} \mathbb{P}(X_0 = i, \tau_j^F < \infty) = \sum_{i \in \mathcal{X}} \mathbb{P}(X_0 = i)\mathbb{P}_i(\tau_j^F < \infty).$$

Because $\sum_{i \in \mathcal{X}} \mathbb{P}(X_0 = i) = 1$, the result follows if we can show

$$\mathbb{P}_i(\tau_j^F < \infty) = 1$$

for all $i, j \in \mathcal{X}$.
Choosing m such that $(P^m)_{i,j} > 0$, we have that

$$\begin{aligned}
1 &= \mathbb{P}_j(X_n = j \text{ for infinitely many } n) \\
&\leq \mathbb{P}_j(X_n = j \text{ for some } n \geq m+1) \\
&= \sum_{k \in \mathcal{X}} \mathbb{P}_j(X_n = j \text{ for some } n \geq m+1 \,|\, X_m = k)\mathbb{P}_j(X_m = k) \\
&= \sum_{k \in \mathcal{X}} \mathbb{P}_k(\tau_j^F < \infty)(P^m)_{j,k}
\end{aligned}$$

Since $\sum_{k \in \mathcal{X}} P^m_{j,k} = 1$ it follows: $\mathbb{P}_k(\tau_j^F < \infty) = 1$ for all $j, k \in mathcalX$. □

Note that $\mathbb{P}_i(\tau_i < \infty) = 1$ does not guarantee that $\mathbb{E}_i \tau_i^F < \infty$. For this reason, we distinguish between two types of recurrence.

Definition 1.6.7. We say a recurrent state $i \in \mathcal{X}$ is *positive recurrent* if $\mathbb{E}\tau_i^F < \infty$. If, instead, $\mathbb{E}\tau_i^F = \infty$, we say i is *null recurrent.*

Example 1.6.8 (The simple random walk on $\mathbb{Z}$ is null recurrent)**.** It is possible to show that the simple random walk on $\mathbb{Z}$ is recurrent (i.e. it returns to 0 with probability 1). However, the expected time it takes to return to zero is infinite.

1.7 Interlude: Estimating Expected Stopping Times and Related Probabilities

There are not always closed-form formulas for the expected values of stopping times. For that reason, it is often necessary to estimate such expected values using Monte Carlo methods.

The fundamental theorem that justifies the use of Monte Carlo methods is the *Strong Law of Large Numbers.*

Theorem 1.7.1 (Strong Law of Large Numbers (SLLN))**.** Let $\{Y_n\}_{n \in \mathbb{N}\setminus\{0\}}$ be a sequence of iid random variables with $\mathbb{E}|Y_1| < \infty$. Then

$$\frac{S_n}{n} \to \mathbb{E}Y_1 \text{ almost surely,}$$

as $n \to \infty$, where $S_n = Y_1 + \ldots + Y_n$.

This theorem tells us that we can estimate the expectation of a random variable Y by generating a sequence of N iid replicates of it, $\{Y_n\}_{n=1}^N$, and taking their sample average. As N grows larger, this sample average will converge almost surely to $\mathbb{E}Y$. We will talk a lot more about the properties of such estimators (and their errors) later in this course.

We can use these results to estimate the expected return time to a state i (that is, $\mathbb{E}_i\tau_i^F$). We will see later that it is often easy to calculate this value exactly, but it is a good place to start learning about how to estimate things. In order to be able to use the SLLN, we need to make sure that $\mathbb{E}_i|\tau_i^F| < \infty$. Because $\tau_i^F \geq 0$, this is equivalent to the condition that $\mathbb{E}\tau_i^F < \infty$. In other words, we need to make sure that the state i is positive recurrent. If i is positive recurrent, when then simply need to find a way to generate independent realizations of τ_i^F. We can do this by repeatedly starting the Markov chain at state i (i.e. with initial distribution $\boldsymbol{\delta}_i$) and recording the time taken until it returns of i. The average of these values will be a (hopefully quite good) estimate of $\mathbb{E}\tau_i^F$.

Example 1.7.2 (Estimating an expected return time)**.** Consider a Markov chain with transition matrix

$$P = \begin{pmatrix} 0 & 3/4 & 0 & 1/4 \\ 0 & 1/2 & 1/2 & 0 \\ 1/8 & 5/8 & 0 & 2/8 \\ 0 & 0 & 3/4 & 1/4 \end{pmatrix}$$

The corresponding transition graph is illustrated in Figure 1.7.1.

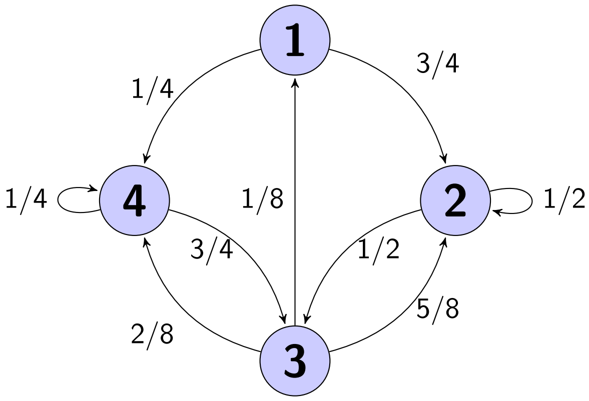

Figure 1.7.1: Transition graph of the Markov chain.

We wish to estimate the expected time it takes to return to state 1 when we start in state 1. We do this using the following Matlab code.

Listing 1.5: Estimating $\mathbb{E}_i\tau_i^F$

```
N=10000
Y=c()
P=rbind(c(0,3/4,0,1/4), c(0,1/2,1/2,0), c(1/8,5/8,0,2/8), c(0,0,3/4,1/4))
i=1

for(j in 1:N)
```

```
{
  t=0
  X=i
  Z=runif(1, min=0, max=1)
  X=min(which(Z<=cumsum(P[X,])))
  t=t+1
  while(X!=i)
  {
    Z=runif(1, min=0, max=1)
    X=min(which(Z<=cumsum(P[X,])))
    t=t+1
  }
  Y[j]=t
}
mean(Y)
```

As well as estimating the expected value of stopping times, it is often useful to estimate probabilities concerning stopping times. For example, imagine that $\{X_n\}_{n\in\mathbb{N}}$ represents the daily price of a share in Apple. We might want to know the probability it reaches \$200, say, before it falls to 50. We can write this in terms of stopping times as

$$\mathbb{P}(\tau_{200}^F < \tau_{50}^F).$$

In order to estimate such probabilities (using the idea from the SLLN) we need to be able to write them as expectations of random variables. Thankfully, this is very straightforward, as we have

$$\mathbb{P}(A) = \mathbb{E}\mathbb{1}(A),$$

where A is some event and $\mathbb{I}(A)$ is the indicator function which takes the value 1 when the event occurs and 0 otherwise. In other words, we can estimate this probability by repeatedly running a Markov chain starting at say $X_0 = 100$ until $\tau = \min\{\tau_{200}^F, \tau_{50}^F\}$ and recording a 1 if $X_\tau = 200$ and a 0 if $X_\tau = 50$.

Example 1.7.3. The probability a simple random walk hits 10 before -7 We can use this approach to estimate the probability a simple random walk, $\{X_n\}_{n\in\mathbb{N}}$, starting at 0 hits 10 before it hits -7. That is,

$$\ell = \mathbb{P}_0(\tau_{10}^F < \tau_{-7}^F).$$

Listing 1.6: Estimating $\mathbb{P}_0(\tau_{10}^F < \tau_{-7}^F)$.

```
N=10^6
Y=c()
A=10
B=-7
p= 0.5

for(j in 1:n)
{
  X=0
  Z= rbinom(n=1,size=1, prob=p)
  X= X + 1*Z + 1*(Z-1)
  while(X!=A & X!=B)
  {
    Z= rbinom(n=1,size=1, prob=p)
    X= X + 1*Z + 1*(Z-1)
  }
  Y[j]= X==A
}
mean(Y)
```

1.8 Reminder: Eigenvalues

It is perhaps now a good time to start thinking about eigenvalues. We will not start to use these straight away, but we will see that they are very important later when we begin to analyze finite state Markov chains a bit more closely.

Definition 1.8.1 (Left eigenvectors and eigenvalues)**.** A non-negative row vector $\boldsymbol{v}^L$ is a *left eigenvector* of a square matrix A if and only if there exists a $\lambda^L \in \mathbb{C}$ such that

$$\boldsymbol{v}^L A = \lambda^L \boldsymbol{v}^L.$$

The corresponding λ^L is called a *(left) eigenvalue* of A.

Definition 1.8.2 (Right eigenvectors and eigenvalues)**.** A non-negative column vector $\boldsymbol{v}^R$ is a *right eigenvector* of a square matrix A if and only if there exists a $\lambda^R \in \mathbb{C}$ such that

$$A\boldsymbol{v}^R = \lambda^R \boldsymbol{v}^R.$$

The corresponding λ^R is called a *(right) eigenvalue* of A.

In order to find the (left) eigenvalues of A we need to find the λ such that

$$\boldsymbol{v}^L A = \lambda \boldsymbol{v}^L \rightarrow \boldsymbol{v}^L(A - \lambda I) = \mathbf{0}.$$

Note that this says we need to choose λ so that it is possible to take a linear combination of the rows of $A - \lambda I$ and get 0. This is only possible if $A - \lambda I$ is not of full rank (i.e., $\det(A - \lambda I) = 0$). Thus, the eigenvalues of A are given by the so-called *characteristic equation*

$$\det(A - \lambda I) = 0.$$

Likewise, in order to find the (right) eigenvalues of A we need to find the λ such that

$$A\boldsymbol{v}^R = \lambda \boldsymbol{v}^R \rightarrow (A - \lambda I)\boldsymbol{v}^R = \mathbf{0}.$$

This implies we need to choose λ so that it is possible to take a linear combination of the columns of $A - \lambda I$ and get 0. Again, this means that λ must be a solution of the characteristic equation

$$\det(A - \lambda I) = 0.$$

It follows from this that the left and right eigenvalues of a matrix are the same.

1.9 Invariant Measures and Invariant Distributions

We saw in Examples 1.3.4 and 1.3.5 that, when the Markov chains started from certain distributions, their distributions did not change as n increased. This lead naturally to a few questions. We will focus on two in this section:

(i) Given a Markov chain, $\{X_n\}_{n\in\mathbb{N}}$, when does there exist a distribution $\boldsymbol{\pi}$ such that if $X_0 \sim \boldsymbol{\pi}$ then $X_n \sim \boldsymbol{\pi}$ for all $n \geq 1$?

(ii) If such a $\boldsymbol{\pi}$ exists, when is it unique?

In order to answer these questions, we need a name for such distributions.

Definition 1.9.1 (Invariant measure)**.** We say a non-trivial (i.e., not all 0s) measure, $\boldsymbol{\lambda}$, is an *invariant measure* of the matrix P if

$$\boldsymbol{\lambda}P = \boldsymbol{\lambda}.$$

Note that, if $\boldsymbol{\lambda}$ is an invariant measure, then so is $\alpha\boldsymbol{\lambda}$ for all $\alpha > 0$.

Definition 1.9.2 (Invariant distribution)**.** We say a distribution, $\boldsymbol{\pi}$, is an *invariant distribution* of the matrix P if

$$\boldsymbol{\pi}P = \boldsymbol{\pi}.$$

An invariant distribution is also called a *stationary distribution* and an *equilibrium distribution.* The following lemma explains why we call $\boldsymbol{\pi}$ a stationary distribution (once a Markov chain has distribution $\boldsymbol{\pi}$ it will never stop having distribution $\boldsymbol{\pi}$).

Lemma 1.9.3. If $\boldsymbol{\pi}$ is a stationary distribution of the transition matrix P and $\{X_n\}_{n\in\mathbb{N}}$ is Markov($\boldsymbol{\pi}$,P), then $X_n \sim \boldsymbol{\pi}$ for all $n \geq 1$.

Proof. We have, by Theorem 1.3.3, that $X_n \sim \boldsymbol{\pi}P^n$. But,

$$\boldsymbol{\pi}P^n = \boldsymbol{\pi}PP^{n-1} = \boldsymbol{\pi}P^{n-1} = \cdots = \boldsymbol{\pi}P = \boldsymbol{\pi}.$$

□

When $\mathcal{X}$ is finite, we know that a stochastic matrix P has a right eigenvector of 1s, with corresponding eigenvalue 1 as

$$P(1,\ldots,1)^\intercal = (1,\ldots,1)^\intercal.$$

This implies that there must exist a corresponding left eigenvector for the eigenvalue 1. That is, a row vector $\boldsymbol{v}$ such that

$$\boldsymbol{v}P = \boldsymbol{v}.$$

Unfortunately, this result does not help us too much. This is because we have no guarantee that the elements of $\boldsymbol{v}$ are non-negative, which is what we need in order for $\boldsymbol{v}$ to be a measure. However, with a little bit more theory from linear algebra we can in fact establish when an invariant measure (and even a stationary distribution) exist. We will come back to this later, but first we will see that we can establish conditions for the existence and uniqueness of invariant measures and distributions using purely probabilistic tools (and without the restriction that the state space is finite). We will not give the proofs of all these theorems, but you can find them all in the book by Norris [6].

Definition 1.9.4. For a fixed $k \in \mathcal{X}$ define $\boldsymbol{\gamma}^k$ by

$$(\gamma^k)_i = \mathbb{E}_k \sum_{n=0}^{\tau_k^F - 1} \mathbb{I}(\{X_n = i\}).$$

Notation/Reminder:

$$\begin{aligned} V_i(n) &= \sum_{l=0}^{n-1} \mathbb{1}\{X_l = i\} \quad \text{number of visits to i before time n} \\ V_i^k &= V_i(\tau_k^F) \quad \text{number of visits to i before first return to k} \\ (\gamma^k)_i &= \mathbb{E}[V_i^k] \quad \text{mean number of visits to i between successive visits to k} \end{aligned}$$

If you think about it, you should be able to see that $\boldsymbol{\gamma}^k$ is a vector where, for $i \neq k$, the ith element corresponds to the expected number of visits to state i in between visits to state k. Also, $(\boldsymbol{\gamma}^k)_k = 1$. The following theorem says that, when P is irreducible and recurrent, $\boldsymbol{\gamma}^k$ is an invariant measure (i.e., it is invariant, not all 0s, and has non-negative elements).

Theorem 1.9.5. Let P be irreducible and recurrent. Then,

(i) $(\boldsymbol{\gamma}^k)_k = 1$.

(ii) $\boldsymbol{\gamma}^k$ satisfies $\boldsymbol{\gamma}^k P = \boldsymbol{\gamma}^k$.

(iii) $0 < (\boldsymbol{\gamma}^k)_i < \infty$ for all $i \in \mathcal{X}$.

Proof. (i) obvious (see Remark).
(ii) For $n = 1, 2, \ldots$ the event $\{\tau_k^F \geq n\}$ only depends on $X_0, \ldots, X_{n-1}$. So by the strong Markov property we have:

$$\mathbb{P}_k(X_{n-1} = i, X_n = j, \tau_k^F \geq n) = P_{i,j}\mathbb{P}_k(X_{n-1} = i, \tau_k^F \geq n). \tag{1.8}$$

Since P is recurrent, we have (proof 1.6.6)

$$\mathbb{P}_k(\tau_k^F < \infty) = 1.$$

Moreover with probability 1 it is: $X_{\tau_k^F} = X_0 = k$.
So for all $j \in \mathcal{X}$(including $j = k$):

$$\begin{aligned}
\gamma_j^k &= \mathbb{E}[V_i^k] \\
&= \mathbb{E}_k[\sum_{n=1}^{\tau_k^F} \mathbb{1}\{X_n = j\}] \\
&= \sum_{n=1}^{\infty} \mathbb{E}_k[\mathbb{1}\{X_n = j\}\mathbb{1}\{\tau_k^F \geq n\}] \\
&= \sum_{i\in\mathcal{X}} \sum_{n=1}^{\infty} \mathbb{P}_k(X_n = j, X_{n-1} = i, \tau_k^F \geq n) \\
&= \sum_{i\in\mathcal{X}} \sum_{n=1}^{\infty} P_{i,j}\mathbb{P}_k(X_{n-1} = i, \tau_k^F \geq n) \\
&= \sum_{i\in\mathcal{X}} P_{i,j} \sum_{m=0}^{\infty} \mathbb{P}_k(X_m = i, \tau_k^F - 1 \geq m) \\
&= \sum_{i\in\mathcal{X}} P_{i,j}\mathbb{E}_k[\sum_{m=0}^{\tau_k^F-1} \mathbb{1}\{X_m = j\}] \\
&= \sum_{i\in\mathcal{X}} P_{i,j}\gamma_i^k,
\end{aligned}$$

where the fourth equality follows by (1.8).
(iii) P is irreducible, so for all $i \in \mathcal{X}$ there exist $m, n \geq 0$ with $P_{i,k}^m > 0$ and $P_k^n, i > 0$. It follows that:

$$\gamma_i^k \geq \gamma_k^k P_{k,i}^n = P_{k,i}^n > 0$$

and

$$\gamma_i^k P_{i,k}^m \leq \gamma_k^k = 1.$$

□

The next theorem gives the conditions (irreducibility and recurrence) under which $\boldsymbol{\gamma}^k$ is a unique invariant measure (up to scaling by a constant).

Theorem 1.9.6. Let P be irreducible and $\boldsymbol{\lambda}$ an invariant measure for P with $(\boldsymbol{\lambda})_k = 1$. Then $\boldsymbol{\lambda} \geq \boldsymbol{\gamma}^k$ (element-wise). If, in addition, P is recurrent then $\boldsymbol{\lambda} = \boldsymbol{\gamma}^k$.

Proof. For each $j \in \mathcal{X}$ we have that

$$\begin{aligned}
\lambda_j &= \sum_{i_0 \in \mathcal{X}} \lambda_{i_0} P_{i_0,j} \\
&= P_{k,j} + \sum_{i_0 \neq k} \lambda_{i_0} P_{i_0,j} \\
&= ... = P_{k,j} + \sum_{i_0 \neq k} P_{k,i_0} P_{i_0,j} + \dots \sum_{i_0,i_1,\dots,i_n \neq k} \lambda_{i_n} P_{i_n,i_{n-1}} \dots P_{i_1,i_0} \\
&\geq \mathbb{P}_k(X_1 = j, \tau_k^F \geq 1) + \mathbb{P}_k(X_2 = j, \tau_k^F \geq 2) + \dots + \mathbb{P}_k(X_n = j, \tau_k^F \geq n) \\
&\to \mathbb{E}_k[\sum_{n=0}^{\tau_k^F - 1} \mathbb{1}\{X_n = j\}] = \gamma_j^k.
\end{aligned}$$

as $n \to \infty$.
So $\lambda \geq \gamma^k$ (element-wise).
If P is recurrent, then γ^k is invariant by 1.9.5(ii). So $\mu = \lambda - \gamma^k$ is invariant and $\mu \geq 0$. Since P is irreducible for arbitrary (fixed) $i \in \mathcal{X}$ there exists an $n \geq 0$ with $P_{i,k}^n > 0$ and therefore:

$$0 = \lambda_k - \gamma_k^k = \mu_k = \sum_{j \in \mathcal{X}} \mu_j P_{j,k}^n \geq \mu_i P_{i,k}^n$$

which results in $\mu_i = 0$. □

The existence of a unique invariant measure, $\boldsymbol{\lambda}$, does not guarantee that we have a stationary distribution. In order for that to be true, we also need $\sum_{j \in \mathcal{X}} (\boldsymbol{\lambda})_j < \infty$. Then, we can define a stationary distribution by

$$(\boldsymbol{\pi})_i = \frac{(\boldsymbol{\lambda})_i}{\sum_{j \in \mathcal{X}} (\boldsymbol{\lambda})_j} \quad \text{for all } i \in \mathcal{X}.$$

In order for this to be true, we need P to satisfy one last condition: positive recurrence.

Theorem 1.9.7. Let P be irreducible. Then the following are equivalent:

(i) Every state is positive recurrent.

(ii) Some state $i \in \mathcal{X}$ is positive recurrent.

(iii) P has an invariant distribution $\boldsymbol{\pi}$.

Furthermore, $(\boldsymbol{\pi})_i = 1/\mathbb{E}_i \tau_i^F$ for all $i \in \mathcal{X}$.

Proof.

Part 1. It is clear that (i) implies (ii).

Part 2. Assuming (ii) holds, we have that some $i \in \mathcal{X}$ is positive recurrent. If i is positive recurrent then it is recurrent. Thus, P is recurrent (as it is irreducible). Theorem 1.9.5 then implies that $\boldsymbol{\gamma}^i$ is invariant. Thus, if we can show

$$\sum_{j \in \mathcal{X}} (\boldsymbol{\gamma}^i)_j < \infty,$$

then we have established that P has an invariant distribution. Now, again swapping sums, we have

$$\begin{aligned}
\sum_{j \in \mathcal{X}} (\boldsymbol{\gamma}^i)_j &= \sum_{j \in \mathcal{X}} \mathbb{E}_i \sum_{n=0}^{\tau_i^F - 1} \mathbb{I}(\{X_n = j\}) \\
&= \mathbb{E}_i \sum_{j \in \mathcal{X}} \sum_{n=0}^{\infty} \mathbb{I}(\{X_n = j\}) \mathbb{I}(\{n \le \tau_i^F\}) \\
&= \mathbb{E}_i \sum_{n=0}^{\infty} \sum_{j \in \mathcal{X}} \mathbb{I}(\{X_n = j\}) \mathbb{I}(\{n \le \tau_i^F\}) \\
&= \mathbb{E}_i \sum_{n=0}^{\infty} \mathbb{I}(\{n \le \tau_i^F\}) \\
&= \mathbb{E}_i \tau_i^F < \infty,
\end{aligned}$$

where the inequality follows from the fact that i is positive recurrent. Thus, we have a stationary distribution $\boldsymbol{\pi}$ defined by

$$(\boldsymbol{\pi})_j = \frac{(\boldsymbol{\gamma}^i)_j}{\sum_{k \in \mathcal{X}} (\boldsymbol{\gamma}^i)_k} \quad \text{for all } j \in \mathcal{X},$$

so (ii) implies (iii).

Part 3. Assuming (iii) holds, we have an invariant distribution $\boldsymbol{\pi}$ for P. In addition, as P is irreducible, for a given $k \in \mathcal{X}$ we can find an n such that

$$(\boldsymbol{\pi})_k = \sum_{i \in \mathcal{X}} (\boldsymbol{\pi})_i (P^n)_{i,k} > 0.$$

So, we know that $(\boldsymbol{\pi})_k > 0$ for all $k \in \mathcal{X}$. Thus, choosing an arbitrary but fixed $k \in \mathcal{X}$, we can define a measure $\boldsymbol{\lambda}$ by

$$(\boldsymbol{\lambda})_i = \frac{(\boldsymbol{\pi})_i}{(\boldsymbol{\pi})_k} \quad \text{for all } i \in \mathcal{X}.$$

We know that $\boldsymbol{\lambda}$ is invariant because it is a scalar multiple of $\boldsymbol{\pi}$ (an invariant measure). In addition, we have $(\boldsymbol{\lambda})_k = 1$. So, using Theorem 1.9.6, we have that $\boldsymbol{\lambda} \ge \boldsymbol{\gamma}^k$ (element-wise). Thus,

$$\sum_{i \in \mathcal{X}} (\boldsymbol{\gamma}^k)_i \le \sum_{i \in \mathcal{X}} (\boldsymbol{\lambda})_i = \sum_{i \in \mathcal{X}} \frac{(\boldsymbol{\pi})_i}{(\boldsymbol{\pi})_k} = \frac{1}{(\boldsymbol{\pi})_k} < \infty. \tag{1.9}$$

As $\sum_{i \in \mathcal{X}} (\boldsymbol{\gamma}^k)_i = \mathbb{E}_k \tau_k^F$, this implies $\mathbb{E}_k \tau_k^F < \infty$. That is, k is positive recurrent. As k was arbitrary, this implies all states are positive recurrent. That is, (iii) implies (ii).

Part 4. As P is recurrent, we have by Theorem 1.9.6 that $\lambda = \gamma^k$ and the inequality in (1.9) is actually an equality. Thus $\mathbb{E}_k \tau_k^F = 1/(\boldsymbol{\pi})_k$.

□

In summary, if we have an irreducible and positive recurrent P matrix, we have a unique stationary distribution $\boldsymbol{\pi}$.

Example 1.9.8 (Finding the stationary distribution of a chain)**.** Say we have a Markov chain with the following transition matrix

$$P = \begin{pmatrix} 1/3 & 2/3 & 0 \\ 0 & 1/4 & 3/4 \\ 1/3 & 1/3 & 1/3 \end{pmatrix}.$$

The corresponding transition graph is illustrated in Figure 1.9.1.

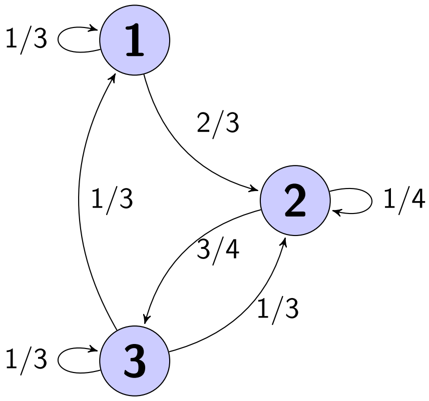

Figure 1.9.1: Transition graph of the Markov chain.

This chain is clearly irreducible and positive recurrent. We can solve for the stationary distribution by solving $\boldsymbol{\pi} P = \boldsymbol{\pi}$. That is, we have the following three equations

$$\begin{aligned} &1/3(\boldsymbol{\pi})_1 + 1/3(\boldsymbol{\pi})_3 = (\boldsymbol{\pi})_1 \\ &2/3(\boldsymbol{\pi})_1 + 1/4(\boldsymbol{\pi})_2 + 1/3(\boldsymbol{\pi})_3 = (\boldsymbol{\pi})_2 \\ &3/4(\boldsymbol{\pi})_2 + 1/3(\boldsymbol{\pi})_3 = (\boldsymbol{\pi})_3. \end{aligned}$$

Solving these, we find $(\boldsymbol{\pi})_2 = 16/9(\boldsymbol{\pi})_1$ and $(\boldsymbol{\pi})_3 = 2(\boldsymbol{\pi})_1$. We need $(\boldsymbol{\pi})_1 + (\boldsymbol{\pi})_2 + (\boldsymbol{\pi})_3 = 1$, so we choose

$$(\boldsymbol{\pi})_1 = \frac{1}{1 + 16/9 + 2} = \frac{9}{43}.$$

This gives $\boldsymbol{\pi} = (9/43, 16/43, 18/43)$.

1.10 Markov Chains and Reversibility

What happens if instead of thinking about a Markov chain starting at time 0 and running forward, we think about a Markov chain starting at time N and running backwards? In general, this might not be a Markov chain. However, it turns out that if we start an irreducible Markov chain from a stationary distribution at time N, then the reversed process is also a Markov chain.

First, let us think about what the transition probabilities of a Markov chain running backwards would look like:

$$\begin{aligned}\mathbb{P}(X_{n-1} = i_{n-1} \,|\, X_n = i_n) &= \frac{\mathbb{P}(X_{n-1} = i_{n-1}, X_n = i_n)}{\mathbb{P}(X_n = i_n)} \\ &= \frac{\mathbb{P}(X_n = i_n \,|\, X_{n-1} = i_{n-1})\mathbb{P}(X_{n-1} = i_{n-1})}{\mathbb{P}(X_n = i_n)}.\end{aligned}$$

If the Markov chain starts from a stationary distribution $\boldsymbol{\pi}$ then we have a good idea what $\mathbb{P}(X_n = i_n)$ and $\mathbb{P}(X_{n-1} = i_{n-1})$ should look like. This motivates the following theorem.

Theorem 1.10.1. Let P be irreducible with invariant distribution $\boldsymbol{\pi}$ Let $\{X_n\}_{n=0}^N$ be Markov$(\boldsymbol{\pi}, P)$ and set $Y_n = X_{N-n}$. Then $\{Y_n\}_{n=0}^N$ is Markov $(\boldsymbol{\pi}, \widehat{P})$, where $\widehat{P}$ is given by

$$(\boldsymbol{\pi})_j(\widehat{P})_{j,i} = (\boldsymbol{\pi})_i(P)_{i,j} \quad \text{for all } i, j \in \mathcal{X},$$

with $\widehat{P}$ irreducible with stationary distribution $\boldsymbol{\pi}$. We call $\{Y_n\}_{n=0}^N$ the *time reversal* of $\{X_n\}_{n=0}^N$.

Proof. Note that we have

$$(\widehat{P})_{j,i} = \frac{(\boldsymbol{\pi})_i}{(\boldsymbol{\pi})_j}(P)_{i,j} \quad \text{for all } i, j \in \mathcal{X},$$

and

$$(P)_{i,j} = \frac{(\boldsymbol{\pi})_j}{(\boldsymbol{\pi})_i}(\widehat{P})_{j,i} \quad \text{for all } i, j \in \mathcal{X},$$

Part 1. First, we show $\widehat{P}$ is a stochastic matrix. As it clearly has non-negative entries, this reduces to showing that its row sums are 1. Observe that, using the fact that $\boldsymbol{\pi}$ is invariant for P,

$$\sum_{i\in\mathcal{X}}(\widehat{P})_{j,i} = \frac{1}{(\boldsymbol{\pi})_j}\sum_{i\in\mathcal{X}}(\boldsymbol{\pi})_i(P)_{i,j} = \frac{1}{(\boldsymbol{\pi})_j}(\boldsymbol{p}i P)_j = \frac{(\boldsymbol{\pi})_j}{(\boldsymbol{\pi})_j} = 1.$$

Part 2. Next, we show $\boldsymbol{\pi}$ is invariant for $\widehat{P}$. We have

$$(\boldsymbol{\pi}\widehat{P})_i = \sum_{j\in\mathcal{X}}(\boldsymbol{\pi})_j(\widehat{P})_{j,i} = \sum_{j\in\mathcal{X}}(\boldsymbol{\pi})_j\frac{(\boldsymbol{\pi})_i}{(\boldsymbol{\pi})_j}(P)_{i,j} = (\boldsymbol{\pi})_i\sum_{j\in\mathcal{X}}(P)_{i,j} = (\boldsymbol{\pi})_i.$$

Part 3. We now need to show $\{Y_n\}_{n=0}^N$ is Markov$(\boldsymbol{\pi}, P)$. We can use Theorem 1.1.11 and the fact that $\{X_n\}_{n=0}^N$ is Markov$(\boldsymbol{\pi}, P)$ to do this. Observe that, for all paths with non-zero probability,

$$\begin{aligned}
&\mathbb{P}(Y_0 = i_0, Y_1 = i_1, \ldots, Y_n = i_n) \\
&= \mathbb{P}(X_{N-n} = i_n, \ldots, X_{N-1} = i_1, X_N = i_0) \\
&= (\boldsymbol{\pi})_{i_n}(P)_{i_n,i_{n-1}} \cdots (P)_{i_1,i_0} \\
&= (\boldsymbol{\pi})_{i_n}\frac{(\boldsymbol{\pi})_{i_{n-1}}}{(\boldsymbol{\pi})_{i_n}}(\widehat{P})_{i_{n-1},i_n} \cdots \frac{(\boldsymbol{\pi})_{i_0}}{(\boldsymbol{\pi})_{i_1}}(\widehat{P})_{i_0,i_1} \\
&= (\boldsymbol{\pi})_{i_0}(\widehat{P})_{i_0,i_1} \cdots (\widehat{P})_{i_{n-1},i_n}
\end{aligned}$$

Part 4. Finally, we need to show that $\widehat{P}$ is irreducible. To do this, for arbitrary i and j in $\mathcal{X}$ we need to be able to find an n and path $\{i_1, \ldots, i_{n-1}\}$ such that

$$(\widehat{P})_{i,i_1} \ldots (\widehat{P})_{i_{n-2},i_{n-1}}(\widehat{P})_{i_{n-1},j} > 0.$$

We know P is irreducible, so we can find a path from j to i such that

$$(P)_{j,i_{n-1}}(P)_{i_{n-1},i_{n-2}} \ldots (P)_{i_1,i} > 0.$$

Writing this in terms of $\widehat{P}$ we have

$$\frac{(\boldsymbol{\pi})_{i_{n-1}}}{(\boldsymbol{\pi})_j}(\widehat{P})_{i_{n-1},j}\frac{(\boldsymbol{\pi})_{i_{n-2}}}{(\boldsymbol{\pi})_{i_{n-1}}}(\widehat{P})_{i_{n-2},i_{n-1}} \cdots \frac{(\boldsymbol{\pi})_i}{(\boldsymbol{\pi})_{i-1}}(\widehat{P})_{i,i_1} > 0,$$

which implies

$$(\widehat{P})_{i,i_1} \cdots (\widehat{P})_{i_{n-2},i_{n-1}}(\widehat{P})_{i_{n-1},j} > 0.$$

That is, we can find a path with non-zero probability from i to j (so $\widehat{P}$ is irreducible).

□

We have seen that the transition matrix of the time-reversal of a Markov chain starting from stationarity is determined by the equations

$$(\boldsymbol{\pi})_j(\widehat{P})_{j,i} = (\boldsymbol{\pi})_i(P)_{i,j} \quad \text{for all } i, j \in \mathcal{X}.$$

If $\widehat{P} = P$ (i.e., the reverse chain behaves exactly the same as the chain running forward) then these equations could be written as

$$(\boldsymbol{\pi})_j(P)_{j,i} = (\boldsymbol{\pi})_i(P)_{i,j} \quad \text{for all } i, j \in \mathcal{X}.$$

This motivates the following definition.

Definition 1.10.2 (Detailed Balance)**.** A stochastic matrix, P, and a measure, $\boldsymbol{\lambda}$, are said to be in *detailed balance* if

$$(\boldsymbol{\lambda})_j(P)_{j,i} = (\boldsymbol{\lambda})_i(P)_{i,j} \quad \text{for all } i, j \in \mathcal{X}.$$

If we can find a measure, $\boldsymbol{\lambda}$ such that P and $\boldsymbol{\lambda}$ satisfy the detailed balance equations, then the following result shows that this measure will be invariant.

Theorem 1.10.3. If P and $\boldsymbol{\lambda}$ are in detailed balance then $\boldsymbol{\lambda}$ is invariant for P.

Proof. We have

$$(\boldsymbol{\lambda}P)_i = \sum_{j\in\mathcal{X}}(\boldsymbol{\lambda})_j(P)_{j,i} = \sum_{j\in\mathcal{X}}(\boldsymbol{\lambda})_i P_{i,j} = (\boldsymbol{\lambda})_i.$$

□

A Markov chain that, when started from a certain distribution, looks the same running forwards and backwards is said to be *reversible*.

Definition 1.10.4 (Reversible Markov chain)**.** A Markov chain $\{X_n\}_{n\in\mathbb{N}}$ with initial distribution $\boldsymbol{\mu}$ and transition matrix P is said to be reversible if $\{X_{N-n}\}_{n=0}^N$ is Markov$(\boldsymbol{\mu}, P)$ for all $N \geq 1$.

The following theorem summarizes what is (hopefully) already clear: if a Markov chain and a distribution satisfy the detailed balance equations, then the Markov chain will be reversible when started from the distribution.

Theorem 1.10.5. Let P be an irreducible stochastic matrix and let $\boldsymbol{\mu}$ be a distribution. Suppose $\{X_n\}_{n\in\mathbb{N}}$ is Markov$(\boldsymbol{\mu}, P)$. Then, the following are equivalent.

(i) $\{X_n\}_{n\in\mathbb{N}}$ is reversible.

(ii) P and $\boldsymbol{\pi}$ are in detailed balance.

When it is possible, solving the detailed balance equations is often a very nice way to solve for the stationary distribution of a Markov chain. This can be seen in the following examples.

Example 1.10.6 (Reflected random walk)**.** Consider a biased random walk on $\{0, \ldots, N\}$ with state space $\mathcal{X} = \{0, \ldots, N\}$ and the following transition probabilities. For all $1 \leq i \leq N-1$, we have

$$\mathbb{P}(X_{n+1} = j \,|\, X_n = i) = \begin{cases} p & \text{if } j = i+1, \\ q & \text{if } j = i-1, \\ 0 & \text{otherwise.} \end{cases}$$

For the boundaries, we have

$$\mathbb{P}(X_{n+1} = j \,|\, X_n = 0) = \begin{cases} p & \text{if } j = 1, \\ q & \text{if } j = 0, \\ 0 & \text{otherwise,} \end{cases}$$

and

$$\mathbb{P}(X_{n+1} = j \,|\, X_n = N) = \begin{cases} p & \text{if } j = N, \\ q & \text{if } j = N-1, \\ 0 & \text{otherwise.} \end{cases}$$

That is, the walker moves up with probability p, down with probability q. When it tries to jump below 0 or above N it, instead, stays where it is. We can solve for the stationary distribution of this process, for arbitrary N by solving the detailed balance equations. Observe that

$$(\boldsymbol{\pi})_i(P)_{i,j} = (\boldsymbol{\pi})_j P_{j,i} \Rightarrow (\boldsymbol{\pi})_i p = (\boldsymbol{\pi})_{i+1} q \quad \text{for all } 0 \leq i \leq N-1.$$

That is,

$$(\boldsymbol{\pi})_{i+1} = \frac{p}{q}(\boldsymbol{\pi})_i \text{ for all } 0 \leq i \leq N-1.$$

If $p = q$ then all the elements of $\boldsymbol{\pi}$ are identically equal to $1/N$. Otherwise, assume without loss of generality (w.l.o.g.) that $p < q$ (just flip things to get the opposite case). Iterating from $i = 1$, this gives

$$(\boldsymbol{\pi})_i = \left(\frac{p}{q}\right)^i (\boldsymbol{\pi})_0 \text{ for all } 1 \leq i \leq N.$$

In order for $\boldsymbol{\pi}$ to be a distribution, we need its elements to sum to 1. That is, we need

$$\sum_{i=0}^{N} (\boldsymbol{\pi})_i = 1 \Rightarrow \sum_{i=0}^{N} \left(\frac{p}{q}\right)^i (\boldsymbol{\pi})_0 = 1 \Rightarrow (\boldsymbol{\pi})_0 = \frac{1}{\sum_{i=0}^{N} \left(\frac{p}{q}\right)^i}$$

As $p < q$ we have $p/q < 1$ and $\sum_{i=0}^{N} (p/q)^i$ is just a geometric series. So, we have

$$(\boldsymbol{\pi})_0 = \frac{1-(p/q)}{1-(p/q)^{N+1}}.$$

Example 1.10.7 (Example 1.10.6 with numbers)**.** Consider the simple case of the reflected random walk described above where $N = 2$ and $p = 1 - q = 1/3$ (so $p/q = 1/2$). We have the following transition matrix

$$P = \begin{pmatrix} 2/3 & 1/3 & 0 \\ 0 & 2/3 & 1/3 \\ 1/3 & 2/3 & 1/3 \end{pmatrix}.$$

The transition graph is given in Figure 1.10.1.

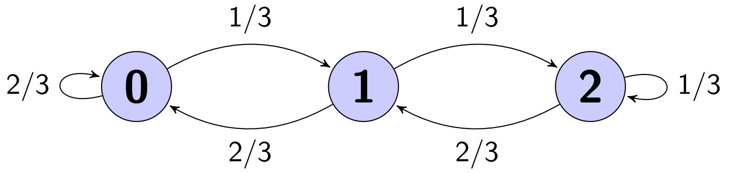

Figure 1.10.1: Transition graph of a reflected random walk with $N = 2$.

According to the result in Example 1.10.6 we have

$$(\boldsymbol{\pi})_0 = \frac{1-1/2}{1-(1/2)^3} = \frac{1/2}{7/8} = 8/14 = 4/7.$$

Then $(\boldsymbol{\pi})_1 = 1/2(\boldsymbol{\pi})_1 = (1/2)(4/7) = 2/7$ and $(\boldsymbol{\pi})_2 = 1/2(\boldsymbol{\pi})_2 = 1/7$.

Note that, as we can solve the detailed balance equations, the Markov chain in Example 1.10.6 is reversible (with respect to the stationary distribution). However, this does not mean that if we started the chain in state 3 and ran it forward for 10 steps that this would (in a statistical sense) look like the chain started in X_{10} and run backwards for 10 steps. Reversibility is a property that only makes sense when things are in stationarity.

Example 1.10.8 (Random walks on graphs)**.** Consider an undirected graph $G = (V, E)$ with at most one edge between any two vertices. For any two vertices $i, j \in V$, we will write $i \sim j$ if $(i, j) \in E$ and $i \not\sim j$ if $(i, j) \notin E$. Note that we do not distinguish between the edge (i, j) and the edge (j, i). If $i \sim j$, we say j is a neighbor of i. The number of neighbors of a vertex is its degree. For a vertex $i \in V$ we will write $\deg(i)$.

We can define a random walk on a connected graph (one where it is possible to find a path from any vertex to any other vertex) as follows. The random walker moves from vertex to vertex. At each time step he or she chooses a neighboring vertex uniformly at random and moves to it. The probability of choosing a given one of the neighboring vertices will be 1 divided by the number of neighboring vertices (i.e., the degree of the current vertex). Thus, the transition probabilities are as follows:

$$\mathbb{P}(X_{n+1} = j \mid X_n = i) = \begin{cases} \frac{1}{\deg(i)} & \text{if } i \sim j, \\ 0 & \text{otherwise.} \end{cases}$$

If $|V| < \infty$, what is the stationary of such a random walk?

We can use the detailed balance equations to solve for this. For each pair $i, j \in V$, with $i \neq j$ we have two possible cases. If $i \not\sim j$ we have

$$(\boldsymbol{\pi})_i 0 = (\boldsymbol{\pi})_j 0,$$

which trivially holds. If $i \sim j$ we have

$$(\boldsymbol{\pi})_i \frac{1}{\deg(i)} = (\boldsymbol{\pi})_j \frac{1}{\deg(j)}.$$

This would be satisfied if, for each $i \in V$, $(\boldsymbol{\pi})_i \propto \deg(i)$. We can check this. In order for everything to sum to one, we normalize $\boldsymbol{\pi}$ to get

$$(\boldsymbol{\pi})_i = \frac{\deg(i)}{\sum_{k \in V} \deg(k)} \quad \text{for all } i \in V.$$

Plugging this into the detailed balance equations, we get

$$\frac{\deg(i)}{\sum_{k \in V} \deg(k)} \cdot \frac{1}{\deg(i)} = \frac{\deg(j)}{\sum_{k \in V} \deg(k)} \cdot \frac{1}{\deg(j)},$$

which clearly holds.

Example 1.10.9 (Caution: not all Markov chains with stationary distributions are reversible)**.** It is important to remember that many irreducible and positive recurrent Markov chains (i.e., those with unique stationary distributions) are not reversible. Consider, for example, the Markov chain with transition matrix

$$P = \begin{bmatrix} 0 & 2/3 & 1/3 \\ 1/3 & 0 & 2/3 \\ 2/3 & 1/3 & 0 \end{bmatrix}$$

The transition graph is given in Figure 1.10.2.

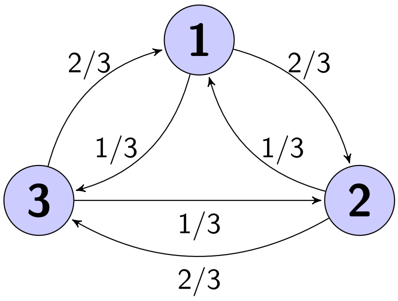

Figure 1.10.2: A Markov chain that is not reversible.

The stationary distribution of this chain is $\boldsymbol{\pi} = (1/3, 1/3, 1/3)$. However, trying to solve the detailed balance equations we have

$$(\boldsymbol{\pi})_1 (P)_{1,2} = (\boldsymbol{\pi})_2 (P)_{2,1} \Rightarrow (1/3)(2/3) = (1/3)(2/3),$$

which is definitely not true! Thus, this Markov chain is not reversible.

Lemma 1.10.10. Let P be the transition matrix of a finite Markov chain and π the stationary distribution of P. Then the following statements hold:

(i) If λ is an eigenvalue of P then $|\lambda| \leq$.

(ii) If P is irreducible, the vectorspace corresponding to the eigenvalue 1 is the one-dimensional vectorspace generated by the column-vector $\mathbf{1} = (1, \ldots, 1)^T$.

(iii) If P and π are in detailed balance the eigenvalues of P are real-valued.

Chapter 2

Markov Chain Monte Carlo

2.1 Inversion method and motivating example

We will now introduce different procedures that can be used to draw samples of random objects. For simple discrete distributions we can use *the discrete inversion method* (c.f. problem 1-4 for the continuous case):
Let X be a random variable on the countable space $\mathcal{X}$. W.L.O.G. we assume that $\mathcal{X} = \{1, \ldots, K\}$, $K \in \mathbb{N}$, or $\mathcal{X} = \mathbb{N}$. We now deduce the method for $\mathcal{X} = \{1, \ldots, K\}$, $K \in \mathbb{N}$ (this deduction can easily extended to the case $\mathcal{X} = \mathbb{N}$).
Let π define the distribution on $\mathcal{X}$, i. e.

$$\pi_k = \mathbb{P}(X = k) \quad \forall k \in \mathcal{X}.$$

Then the distribution function F of X is given by

$$F(x) = \sum_{k=1}^{K} \pi_k \mathbb{1}_{(-\infty,x]}(k) \quad \forall x \in \mathbb{R}$$

and for $m \in \mathcal{X}$ it holds

$$F(m) = \sum_{k=1}^{m} \pi_k.$$

We now calculate the *generalized inverse* of F for $r \in [0, 1]$:

$$\begin{aligned} F^{-1}(r) &= \min\{x \in \mathbb{R} : F(x) = \sum_{k=1}^{K} \pi_k \mathbb{1}_{(-\infty,x]}(k) \geq r\} \\ &= \min\{m \in \mathcal{X} : F(x) = \sum_{k=1}^{m} \pi_k \geq r\}. \end{aligned}$$

We now can use a random generator which draws a uniform distributed random sample and the generalized inverse of F to draw a random sample which has π as distribution.

Example 2.1.1. We consider a binomial distribution with $n = 5$ and $p = \frac{1}{2}$ and draw samples using the inversion method. The program code is given by

Listing 2.1: Simulating a binomial distributed variable using the inversion method

```
### The inversion method
### binomial distribution
n=5
p0= 1/2
lambda = dbinom(0:n, size=n, prob=p0)
### using a while-loop
m=0
F=0
Z=runif(n=1, min=0, max=1)
while(F <Z)
{
  m=m+1
  F=F+lambda[m] ### F is the distribution function F(m)
}
m

### using cumsum
m= min(which(Z<=cumsum(lambda)))
m
plot(pbinom(0:n, size=n, prob=p0))
abline(h=Z)
abline(v=m)
```

One of the main things we will focus on in this course is how we go about producing samples of **interesting and complicated** random objects. We will consider many such objects, arising not just in maths but also in physics, computer science, finance and image processing. Let us start with the following example problem.

Example 2.1.2 (Coloring the vertices of a graph)**.** Consider a graph $G = (V, E)$ and fix a set of colors $\mathcal{K} = \{1, \ldots, k\}$. Being curious, we might ask if it possible to assign one of the colors in $\mathcal{K}$ to each vertex in G so that no two neighboring vertices have the same color. Such a coloring is called a *(proper) k-coloring*. The smallest k such that such a coloring is possible is called the *chromatic number* of the graph G. Finding this number is a very hard problem (it is NP hard!). However, it can be shown that it is possible to find a k-coloring for a graph when $k \geq \Delta(G) + 1$, where $\Delta(G)$ is the maximum degree of a vertex in G. Although this might seem like an abstract problem, it has lots of real-world applications. Among other things, we can think of lots of scheduling and timetabling problems as graph coloring problems. For example, each vertex might be a lecture and edges between lectures could indicate lectures that cannot take place at the same time. A coloring would then give a way for the lectures to be scheduled so that there are no clashes between lectures. Another example of a graph coloring problem is Sudoku. Suppose we want to generate a k-coloring of a graph at random (such that, all k-colorings are equally likely). How would we go about doing this?

2.2 Acceptance-Rejection

Problems like that in Example 2.1.2 arise frequently. We want to generate a sample uniformly from some (complicated) set Ω or, more generally, draw from a complicated distribution that we know very little about. Fortunately, there is a generic method that can produce such samples (called the *acceptance-rejection algorithm*). Unfortunately, it is not always a very good way to generate complicated objects.

The setup is as follows. We want to generate samples from a distribution

$$\widetilde{\boldsymbol{\lambda}} = \frac{\boldsymbol{\lambda}}{\sum_{i \in \mathcal{X}} (\boldsymbol{\lambda})_i}$$

on $\mathcal{X}$. We know the denominator, $\sum_{i \in \mathcal{X}} (\boldsymbol{\lambda})_i$, is finite but we do not necessarily know its exact value. That is, we have a finite measure $\boldsymbol{\lambda}$ but might not be able to calculate the sum of its components. Now, suppose we can have a method for drawing from another distribution, $\boldsymbol{\mu}$, on $\mathcal{X}$ that satisfies the condition that $(\boldsymbol{\lambda})_i > 0 \Rightarrow (\boldsymbol{\mu})_i > 0$ for all $i \in \mathcal{X}$. We can draw from $\widetilde{\boldsymbol{\lambda}}$ as follows.

Algorithm 2.2.1 (Acceptance-Rejection)**.** Let $C \geq \max_{i \in \mathcal{X}} \frac{(\boldsymbol{\lambda})_i}{(\boldsymbol{\mu})_i}$.

(i) Draw $\widetilde{Y} \sim \boldsymbol{\mu}$.

(ii) Draw $U \sim \mathcal{U}(0,1)$.

(iii) If $U < \frac{(\boldsymbol{\lambda})_{\widetilde{Y}}}{C(\boldsymbol{\mu})_{\widetilde{Y}}}$ return $Y = \widetilde{Y}$. Otherwise, repeat from step (i).

Before showing that this algorithm works, lets look at an example to make sure we understand what it tells us to do.

Example 2.2.1 (Drawing from a distribution using acceptance-rejection)**.** This example is artificial (we certainly do not need to use acceptance-rejection here) but illustrates the basic idea. Suppose four horse are running in a race. The probabilities of the various horses winning are proportional to $\boldsymbol{\lambda} = (4,3,2,1)$. So, for example, the probability the first horse wins is given by

$$\frac{(\boldsymbol{\lambda})_1}{\sum_{i=1}^{4} (\boldsymbol{\lambda})_i} = \frac{4}{10} = \frac{2}{5}.$$

We can generate samples from $\widetilde{\boldsymbol{\lambda}}$, the normalized version of $\boldsymbol{\lambda}$, using, for example, the distribution $\boldsymbol{\mu} = (1/4, 1/4, 1/4, 1/4)$. Observe that, in this case,

$$C = \frac{4}{1/4} = 16,$$

so

$$\frac{(\boldsymbol{\lambda})_{\widetilde{Y}}}{C(\boldsymbol{\mu})_{\widetilde{Y}}} = \frac{(\boldsymbol{\lambda})_{\widetilde{Y}}}{16 \cdot (1/4)} = \frac{(\boldsymbol{\lambda})_{\widetilde{Y}}}{4}.$$

The following Matlab code implements the acceptance-rejection algorithm, drawing a sample of size N from $\widetilde{\lambda}$ and plotting a histogram of the sample.

Listing 2.2: Drawing the winning horse using acceptance-rejection

```
N= 10^5
Y=c() ### vector to be filled with the sample
lambda=c(4, 3, 2, 1)
mu= rep(1/4, 4) # we know to sample according to the discrete uniform distribution

for (i in 1:N)
{
  accept=FALSE
  while(accept==FALSE)
  {
    Y_tilde= sample(x=1:4, size=1, prob=mu) ## step one of algorithm 2.1.1
    U= runif(1, min=0, max=1)              ## step two of algorithm 2.1.1
    if(U < lambda[Y_tilde]/4)
    ## check whether U < lambda_(Y_tilde)/(C*mu_(Y_tilde))
    {
      accept=TRUE
    }
  }
  Y[i]=Y_tilde                             ## If accept=TRUE set Y=Y_tilde
}
```

The resulting histogram looks is shown in Figure 2.2.1. Note that the proportions are correct!

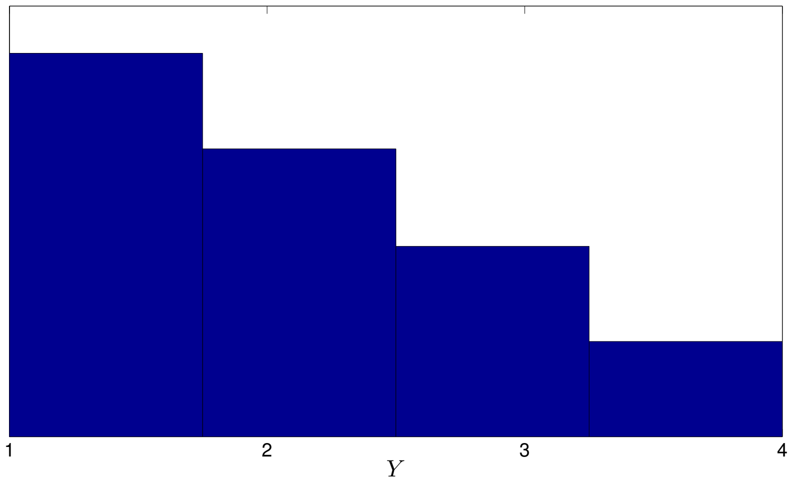

Figure 2.2.1: Histogram of the sample of winning horses generated using acceptance-rejection.

Of course, an example is not enough to prove that the acceptance-rejection algorithm works. For that, we need the following theorem.

Theorem 2.2.2. Given a finite measure, $\boldsymbol{\lambda}$, on $\mathcal{X}$ and a distribution, $\boldsymbol{\mu}$, on $\mathcal{X}$ that satisfies the condition $(\boldsymbol{\lambda})_i > 0 \Rightarrow (\boldsymbol{\mu})_i > 0$ for all $i \in \mathcal{X}$, Algorithm 2.2.1

produces samples from the distribution

$$\widetilde{\boldsymbol{\lambda}} = \frac{\boldsymbol{\lambda}}{\sum_{i \in \mathcal{X}} (\boldsymbol{\lambda})_i}$$

Proof. To keep things simple, let us assume $(\boldsymbol{\mu})_i > 0$ for all $i \in \mathcal{X}$ (everything works anyway, just replace $\mathcal{X}$ in the sums with the set on which $\boldsymbol{\mu}$ is positive). Now, in order to prove the theorem, observe that the acceptance-rejection algorithm outputs a random variable, $\widetilde{Y}$, with distribution $\boldsymbol{\mu}$, conditional on the event $\{U \leq (\boldsymbol{\lambda})_{\widetilde{Y}}/(C(\boldsymbol{\mu})_{\widetilde{Y}})\}$, where U is uniformly distributed on $(0,1)$ and independent of $\widetilde{Y}$. If $(\boldsymbol{\lambda})_i = 0$ we clearly have $\mathbb{P}(Y = i) = 0 = (\widetilde{\lambda})_i$. Otherwise,

$$\mathbb{P}(Y = i) = \mathbb{P}\left(\widetilde{Y} = i \,\middle|\, U \leq \frac{(\boldsymbol{\lambda})_{\widetilde{Y}}}{C(\boldsymbol{\mu})_{\widetilde{Y}}}\right) = \frac{\mathbb{P}\left(\widetilde{Y} = i, U \leq \frac{(\boldsymbol{\lambda})_{\widetilde{Y}}}{C(\boldsymbol{\mu})_{\widetilde{Y}}}\right)}{\mathbb{P}\left(U \leq \frac{(\boldsymbol{\lambda})_{\widetilde{Y}}}{C(\boldsymbol{\mu})_{\widetilde{Y}}}\right)}.$$

Now, for the numerator, we can write

$$\begin{aligned}
\mathbb{P}\left(\widetilde{Y} = i, U \leq \frac{(\boldsymbol{\lambda})_{\widetilde{Y}}}{C(\boldsymbol{\mu})_{\widetilde{Y}}}\right) &= \mathbb{P}\left(U \leq \frac{(\boldsymbol{\lambda})_{\widetilde{Y}}}{C(\boldsymbol{\mu})_{\widetilde{Y}}} \,\middle|\, \widetilde{Y} = i\right) \mathbb{P}\left(\widetilde{Y} = i\right) \\
&= \mathbb{P}\left(U \leq \frac{(\boldsymbol{\lambda})_i}{C(\boldsymbol{\mu})_i}\right) \mathbb{P}\left(\widetilde{Y} = i\right) \\
&= \frac{(\boldsymbol{\lambda})_i}{C(\boldsymbol{\mu})_i} (\boldsymbol{\mu})_i = \frac{(\boldsymbol{\lambda})_i}{C},
\end{aligned}$$

where we use the fact that $0 \leq (\boldsymbol{\lambda})_i/(C(\boldsymbol{\mu})_i) \leq 1$ for all $i \in \mathcal{X}$ to calculate the cumulative probability for the uniform random variable. For the denominator, using the arguments from above, we can write

$$\begin{aligned}
\mathbb{P}\left(U \leq \frac{(\boldsymbol{\lambda})_{\widetilde{Y}}}{C(\boldsymbol{\mu})_{\widetilde{Y}}}\right) &= \sum_{j \in \mathcal{X}} \mathbb{P}\left(U \leq \frac{(\boldsymbol{\lambda})_j}{C(\boldsymbol{\mu})_i}\right) \mathbb{P}\left(\widetilde{Y} = j\right) \\
&= \sum_{j \in \mathcal{X}} \frac{(\boldsymbol{\lambda})_j}{C(\boldsymbol{\mu})_j} (\boldsymbol{\mu})_j = \frac{\sum_{j \in \mathcal{X}} (\boldsymbol{\lambda})_j}{C}.
\end{aligned}$$

Putting everything together, we have

$$\mathbb{P}(Y = i) = \frac{(\boldsymbol{\lambda})_i}{\sum_{j \in \mathcal{X}} (\boldsymbol{\lambda})_j} = (\widetilde{\boldsymbol{\lambda}})_i.$$

□

2.2.1 Sampling uniformly from a complicated set

It is often the case that we want to draw uniformly from some complicated set, Ω, as in Example 2.1.2. It turns out that we can do this using acceptance-rejection (though we will also see that it is usually not a very good way to do things).

The idea is to think of Ω as a subset of a larger set $\mathcal{X}$ (with $|\mathcal{X}| < \infty$), from which it is easy to sample uniformly (Example 2.2.3 gives an example of such a set). Observe, then, that we want to draw from the distribution $\widetilde{\boldsymbol{\lambda}}$ defined by

$$(\widetilde{\boldsymbol{\lambda}})_i = \frac{\mathbb{I}(i \in \Omega)}{|\Omega|} \quad \text{for all } i \in \mathcal{X}.$$

Unfortunately, we usually do not know $|\Omega|$, the number of elements in Ω. However, we will see this is not a problem for the acceptance-rejection method. We draw our proposal variable from $\boldsymbol{\mu}$, defined by

$$(\boldsymbol{\mu})_i = \frac{\mathbb{I}(i \in \mathcal{X})}{|\mathcal{X}|}$$

Again, it turns out that we will not need to know $|\mathcal{X}|$ (though this is usually not too difficult to calculate). In order to use acceptance-rejection, we need to find some $C \geq \max_{i \in \mathcal{X}} \frac{(\widetilde{\boldsymbol{\lambda}})_i}{(\boldsymbol{\mu})_i}$. As $\widetilde{\boldsymbol{\lambda}}$ can only take one non-zero value and $\boldsymbol{\mu}$ is constant, we can choose

$$C = \max_{i \in \mathcal{X}} \frac{(\widetilde{\boldsymbol{\lambda}})_i}{(\boldsymbol{\mu})_i} = \frac{1/|\Omega|}{1/|\mathcal{X}|} = \frac{|\mathcal{X}|}{|\Omega|}.$$

Of course, we do not know what this is (but we will see it does not matter). Remember that we only accept a draw, $\widetilde{Y}$, from $\boldsymbol{\mu}$ if $U < \frac{(\widetilde{\boldsymbol{\lambda}})_{\widetilde{Y}}}{C(\boldsymbol{\mu})_{\widetilde{Y}}}$. Let us think about what the term on the right looks like. We have

$$\frac{(\widetilde{\boldsymbol{\lambda}})_{\widetilde{Y}}}{C(\boldsymbol{\mu})_{\widetilde{Y}}} = \frac{\mathbb{I}(\widetilde{Y} \in \Omega)/|\Omega|}{|\mathcal{X}|C} = \frac{\mathbb{I}(\widetilde{Y} \in \Omega)}{|\Omega||\mathcal{X}|\frac{|\Omega|}{|\mathcal{X}|}} = \mathbb{I}(\widetilde{Y} \in \Omega).$$

Thus

$$\mathbb{P}\left(U < \frac{(\widetilde{\boldsymbol{\lambda}})_{\widetilde{Y}}}{C(\boldsymbol{\mu})_{\widetilde{Y}}}\right) = \begin{cases} 1 & \text{if } \widetilde{Y} \in \Omega, \\ 0 & \text{otherwise.} \end{cases}$$

Putting everything together, we have the following, very simple, algorithm for sampling uniformly from a complicated set.

Algorithm 2.2.2 (Acceptance-Rejection for drawing uniformly from $\Omega \subset \mathcal{X}$)**.** Let $\boldsymbol{\mu}$ be the uniform distribution on $\mathcal{X}$.

(i) Draw $\widetilde{Y} \sim \boldsymbol{\mu}$.

(ii) If $\widetilde{Y} \in \Omega$, return $Y = \widetilde{Y}$. Otherwise, repeat from step (i).

Example 2.2.3 (Sampling random colorings using acceptance-rejection)**.** We are now in a position to generate the (proper) k-colorings described in Example 2.1.2. In order to do this, we need a set, $\mathcal{X}$, from which we can easily generate uniform samples. Denoting the set of all possible k-colorings by Ω, we see that $\Omega \subset \mathcal{X} = \{1, \ldots, k\}^{|V|}$. We can draw from $\mathcal{X}$ by simply assigning each vertex an independent random number, drawn uniformly from $\{1, \ldots, k\}$. We then just need to check if the coloring we produce is a proper coloring or not.

More concretely, consider the graph in Figure 2.2.2. We wish to sample uniformly from all possible 3-colorings of it. We know it is possible to find at least one such coloring (consider, for example, the coloring $(1, 2, 3, 2)$). In order to represent this graph, we use an *adjacency matrix*, E, where the entry $(E)_{i,j} = 1$ if there is an edge between vertices i and j. The code given below produces uniform samples as desired.

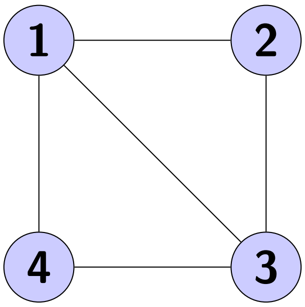

Figure 2.2.2: The graph we wish to color.

Listing 2.3: Sampling a random k-coloring

```
E=rbind(c(0, 1, 1, 1), c(1,0, 1, 0), c(1, 1, 0, 1), c(1, 0, 1, 0))
k=3  ### number of different colors
number_of_vertices=4
proper_coloring=FALSE

while(proper_coloring==FALSE)
{
  coloring = sample(1:k, size=number_of_vertices, replace=TRUE, prob=NULL)
  proper_coloring=TRUE
  for(i in 1:(number_of_vertices-1)) ### check if adjacency matrix and
                                    ### coloring fit
  {
    for(j in (i+1):number_of_vertices)
    {
      if(E[i,j]==1 & coloring[i]==coloring[j])
      {
        proper_coloring=FALSE
      }
    }
  }
}
coloring
```

2.2.2 The performance quality of the acceptance-rejection algorithm

One way to measure the efficiency of the acceptance-rejection algorithm, is to consider the amount of work we need to do to get a successful draw from the target distribution, $\boldsymbol{\lambda}$ and the probability that the algorithm will accept the draw.

Theorem 2.2.4. Let $\tilde{\lambda}$ be a distribution, μ be the distribution and C the constant used in the Acceptance-Rejection algorithm. Let $(Z_n)_{n\in\mathbb{N}}$ be a sequence of random variables defined by

$$Z_n = \begin{cases} 1 & U \le \frac{\tilde{\lambda}_{\tilde{Y}_n}}{C\mu_{\tilde{Y}_n}} \\ 0 & \text{otherwise} \end{cases}$$

and T be a random variable with

$$T = \min\{n \ge 1 : Z_n = 1\}.$$

Then it holds:

(i) $(Z_n)_{n\in\mathbb{N}}$ is an i.i.d. sequence with $p_i = \mathbb{P}(Z_n = i) = \frac{1}{C}$.

(ii) $\mathbb{E}[T] = C$.

(iii) $X_T \sim \tilde{\lambda}$.

Proof. ad (i) $p = \mathbb{P}(Z_n = 1) = \mathbb{P}(U_n \le \frac{\lambda_{\tilde{Y}_n}}{C\mu_{\tilde{Y}_n}}) = 1/C$. Since $(U_n, \tilde{Y}_n)_{n\in\mathbb{N}}$ are i.i.d., it follows that $(Z_n)_{n\in\mathbb{N}}$ are i.i.d.
ad (ii) $\mathbb{P}(T = k) = \mathbb{P}(Z_1 = 0, \ldots, Z_{k-1} = 0, Z_k = 1) = (1-p)^{k-1}p$.
Thus, T is geometrically distributed with expected value $\mathbb{E}[T] = \frac{1}{p} = C$. □

In the case where we use the uniform distribution on $\mathcal{X}$ to draw uniformly from Ω, we have

$$\mathbb{P}(\text{Accept}) = \frac{|\Omega|}{|\mathcal{X}|}.$$

In situations where we need to use acceptance-rejection, it is often the case that $|\Omega| << |\mathcal{X}|$, so many samples need to be generated before one is accepted. Usually, it is difficult to know the size of $|\Omega|$ (or the optimal value of C) a priori. However, it might be possible to get bounds on them. In some simple situations, we can calculate the acceptance probability directly.

Example 2.2.5 (Example 2.2.3 continued)**.** The acceptance probability for the algorithm in Example 2.2.3 is simply

$$\mathbb{P}(\text{Accept}) = \frac{\#\text{ of 3-colorings of } G}{|\{1,2,3\}^4|} = \frac{6}{3^4} = \frac{6}{81},$$

where the 3-colorings are $(1,2,3,2), (1,3,2,3), (2,1,3,1), (2,3,1,3), (3,1,2,1)$ and $(3,2,1,2)$.

The following example helps illustrate a phenomenon called the *curse of dimensionality*, where the efficiency of numerical methods grows small very quickly as the dimension of the problem gets large.

Example 2.2.6 (The limitations of acceptance-rejection)**.** Using a continuous state space version of the acceptance-rejection algorithm, it is possible to generate samples uniformly from the n-dimensional unit ball. We can do this by drawing samples uniformly from the n-dimensional box $[-1,1]^n$ and only accepting those which fall inside the ball. The success probability of this methods is quite high for low values of n (for $n = 2$ it is approximately 0.785). However, as n grows larger, it goes very quickly to zero. Observe that

$$\begin{aligned}\mathbb{P}(\text{Accept}) &= \frac{\text{volume of the } n\text{-dimensional ball}}{\text{volume of } [-1,1]^n} \\ &= \frac{\pi^{n/2}/\Gamma(n/2+1)}{2^n} = \frac{(\sqrt{\pi}/2)^n}{\Gamma(n/2+1)},\end{aligned}$$

which goes to zero incredibly quickly in n, as the numerator gets smaller as n increases and the denominator grows very quickly. This gives a simple picture of what happens when using acceptance-rejection in high dimensions.

2.3 The Metropolis and Metropolis-Hastings Algorithms

2.3.1 The Metropolis algorithm

Instead of sampling 'blindly' using acceptance-rejection, we will try to develop smarter ways to produce samples of complicated objects. Many of these techniques are based on Markov chains and are known, collectively, as Markov Chain Monte Carlo (MCMC). In considering Markov chains, we have so far started with a transition matrix, P, and tried to find a stationary distribution, $\boldsymbol{\pi}$. In MCMC we instead start with a distribution, $\boldsymbol{\pi}$, and try to find a transition matrix, P, that has $\boldsymbol{\pi}$ as a stationary distribution. Note that, although a given P often has a unique stationary distribution, the opposite is not true: there are many choices of P that will produce a given $\boldsymbol{\pi}$.

One way to find a P with stationary distribution $\boldsymbol{\pi}$ is to find a solution to the detailed balance equations

$$(\boldsymbol{\pi})_i(P)_{i,j} = (\boldsymbol{\pi})_j P_{j,i} \quad \text{for all } i \in \mathcal{X}.$$

This is the idea of the Metropolis algorithm.

Algorithm 2.3.1 (Metropolis Algorithm)**.** Given a distribution, $\boldsymbol{\pi}$, on $\mathcal{X}$, a symmetric $|\mathcal{X}| \times |\mathcal{X}|$ transition matrix, Q, and an initial distribution $\boldsymbol{\mu}$:

(i) Draw $X_0 \sim \boldsymbol{\mu}$. Set $n = 0$.

(ii) Draw $Y \sim (Q)_{X_n,\cdot}$.

(iii) Calculate

$$\alpha(X_n, Y) = \min\left\{1, \frac{(\boldsymbol{\pi})_Y}{(\boldsymbol{\pi})_{X_n}}\right\}.$$

(iv) With probability $\alpha(X_n, Y)$, set $X_{n+1} = Y$. Otherwise, set $X_{n+1} = X_n$.

(v) Set $n = n + 1$ and repeat from (ii).

Note that, in order to use the Metropolis algorithm, we do not need to know the normalizing constant of $\boldsymbol{\pi} = \boldsymbol{\lambda} / \sum_{k \in \mathcal{X}} (\boldsymbol{\lambda})_k$, because

$$\frac{(\boldsymbol{\pi})_Y}{(\boldsymbol{\pi})_{X_n}} = \frac{(\boldsymbol{\lambda})_Y / \sum_{k \in \mathcal{X}} (\boldsymbol{\lambda})}{(\boldsymbol{\lambda})_{X_n} / \sum_{k \in \mathcal{X}} (\boldsymbol{\lambda})}.$$

So, how do we know the algorithm works? The first thing to do is check the detailed-balance algorithms are satisfied.

Theorem 2.3.1. The Metropolis algorithm results produces a Markov chain, $\{X_n\}_{n \in \mathbb{N}}$, whose transition matrix, P, is in detailed balance with $\boldsymbol{\pi}$.

Proof. First, we need to establish what P, the transition matrix of $\{X_n\}_{n \in \mathbb{N}}$, looks like. We have that, for all $i, j \in \mathcal{X}$,

$$(P)_{i,j} = \begin{cases} \min\left\{1, \frac{(\boldsymbol{\pi})_j}{(\boldsymbol{\pi})_i}\right\} (Q)_{i,j} & \text{if } i \neq j, \\ 1 - \sum_{j \in \mathcal{X}} \min\left\{1, \frac{(\boldsymbol{\pi})_j}{(\boldsymbol{\pi})_i}\right\} (Q)_{i,j} & \text{if } i = j. \end{cases} \tag{2.1}$$

In order to see (2.1) let A be the event that we accept in step (iv) of the algorithm. Let $i \neq j$, then we have:

$$\begin{aligned} & \mathbb{P}(X_{n+1} = j | X_n = i) \\ = \; & \mathbb{P}(Y = j, A | X_n = i) \\ = \; & \mathbb{P}(A | Y = j, X_n = i) \mathbb{P}(Y = j | X_n = i) \\ = \; & \min\{1, \frac{\pi_j}{\pi_i}\} Q_{i,j}. \end{aligned}$$

Let $i = j$, then we have:

$$\begin{aligned} & \mathbb{P}(X_{n+1} = i | X_n = i) \\ = \; & \mathbb{P}(Y = i, A | X_n = i) + \mathbb{P}(A^c | X_n = i) \\ = \; & \min\{1, \frac{\pi_i}{\pi_i}\} Q_{i,i} + \sum_{j \in \mathcal{X}} \mathbb{P}(A^c | Y = j, X_n = i) \mathbb{P}(Y = j | X_n = i) \\ = \; & Q_{i,i} + \sum_{j \neq i} (1 - \min\{1, \frac{\pi_j}{\pi_i}\}) Q_{i,j} \\ = \; & 1 - \sum_{j \in \mathcal{X}} \min\{1, \frac{\pi_j}{\pi_i}\} Q_{i,j}. \end{aligned}$$

In order to show P satisfies detailed balance, we consider separate cases. First, note that if $(\boldsymbol{\pi})_i = 0$ or $(\boldsymbol{\pi})_j = 0$, the detailed balance equations are trivially satisfied. Otherwise, we first consider the case where $(\boldsymbol{\pi})_j \geq (\boldsymbol{\pi})_i$. We then have

$$\begin{aligned} (\boldsymbol{\pi})_i (P)_{i,j} &= (\boldsymbol{\pi})_i \min\left\{1, \frac{(\boldsymbol{\pi})_j}{(\boldsymbol{\pi})_i}\right\} (Q)_{i,j} = (\boldsymbol{\pi})_i (Q)_{i,j} \\ &= (\boldsymbol{\pi})_i (Q)_{j,i} \frac{(\boldsymbol{\pi})_j}{(\boldsymbol{\pi})_j} = (\boldsymbol{\pi})_j (Q)_{j,i} \frac{(\boldsymbol{\pi})_i}{(\boldsymbol{\pi})_j} \\ &= (\boldsymbol{\pi})_j \min\left\{1, \frac{(\boldsymbol{\pi})_i}{(\boldsymbol{\pi})_j}\right\} (Q)_{j,i} = (\boldsymbol{\pi})_j (P)_{j,i}. \end{aligned}$$

where we use the fact Q is symmetric, so $(Q)_{i,j} = (Q)_{j,i}$. Alternatively, in the case where $(\boldsymbol{\pi})_j < (\boldsymbol{\pi})_i$,

$$\begin{aligned}(\boldsymbol{\pi})_i(P)_{i,j} = (\boldsymbol{\pi})_i \min\left\{1, \frac{(\boldsymbol{\pi})_j}{(\boldsymbol{\pi})_i}\right\}(Q)_{i,j} &= (\boldsymbol{\pi})_i(Q)_{i,j}\frac{(\boldsymbol{\pi})_j}{(\boldsymbol{\pi})_i} \\ &= (\boldsymbol{\pi})_j(Q)_{j,i} = (\boldsymbol{\pi})_j \min\left\{1, \frac{(\boldsymbol{\pi})_i}{(\boldsymbol{\pi})_j}\right\}(Q)_{j,i} \\ &= (\boldsymbol{\pi})_j(P)_{j,i}.\end{aligned}$$

Thus, P and $\boldsymbol{\pi}$ are in detailed balance for all $i, j \in \mathcal{X}$. □

Of course, the mere fact that P and $\boldsymbol{\pi}$ are in detailed balance does not guarantee that $\boldsymbol{\pi}$ is the unique stationary distribution of P (Markov chains that are not irreducible can have more than one $\boldsymbol{\pi}$ that satisfies the detailed balance equations). However, the following ensures we have a Markov chain with unique stationary distribution $\boldsymbol{\pi}$. Let $\Omega = \{i \in \mathcal{X} : (\boldsymbol{\pi})_i > 0\}$. A Metropolis chain with transition matrix, P, will have unique stationary distribution $\boldsymbol{\pi}$ if it satisfies the conditions

(i) $\mathbb{P}_{\boldsymbol{\mu}}(X_0 \in \Omega) = 1$.

(ii) P is irreducible on Ω.

This follows from results we already have.

A more serious issue is how we know that our chain will converge to distribution $\boldsymbol{\pi}$ if we do not start with $X_0 \sim \boldsymbol{\pi}$. This will be the subject of the next section of our notes.

Example 2.3.2 (Example 2.2.1 continued.)**.** Let us use the Metropolis algorithm to sample from the distribution described in Example 2.2.1, which is given by $\boldsymbol{\lambda} = (4, 3, 2, 1)$. We use the proposal matrix

$$Q = \begin{pmatrix} 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 \end{pmatrix}.$$

This results in a P matrix that is clearly irreducible. We use the initial distribution $\boldsymbol{\mu} = \boldsymbol{\delta}_1 = (1, 0, 0, 0)$. The Matlab code is as follows.

Listing 2.4: Drawing the winning horse using the Metropolis algorithm

```
N = 10^5;
X=c()
X[1]=1
lambda= c(4, 3, 2, 1)
for (i in 2:N)
{
  Y= sample(1:4, size=1, prob=NULL)
  alpha=lambda[Y]/lambda[X[i-1]]
  U = runif(1, min=0, max=1)
  if(U < alpha) # with prob alpha set X[i]=Y
```

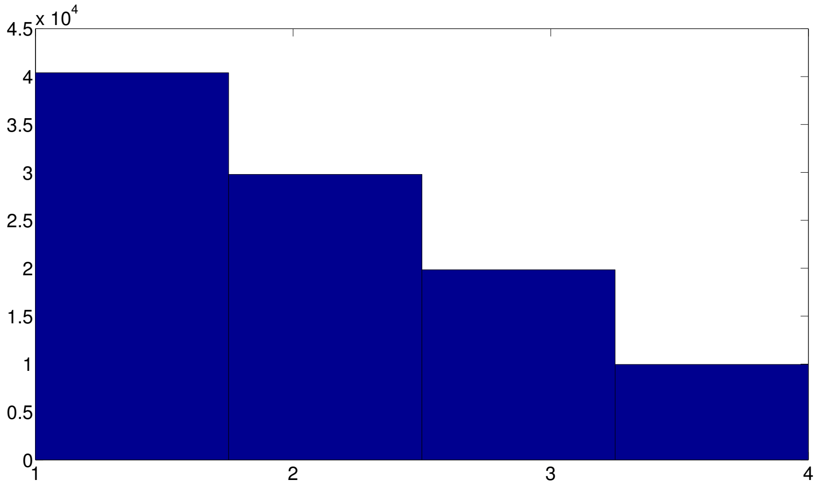

Figure 2.3.1: Histogram of the sample of winning horses generated using the Metropolis algorithm.

```
  {
    X[i]=Y
  }
  else X[i]=X[i-1]
}
hist(X, 4)
```

The resulting histogram is shown in Figure 2.3.1. Note that the proportions seem to be right!

The above example is quite artificial. It is straightforward to sample the winning horses without using the Metropolis algorithm. However, in the next example, where we again consider colorings of graphs, it is not clear how to sample using a simpler method (other than acceptance-rejection, which does not work well for large numbers of vertices).

Example 2.3.3 (Sampling random colorings using the Metropolis algorithm)**.** We can come up with some quite clever Markov chains that allow us to sample from the k-colorings of a graph $G = (V, E)$. We use the following simple (but quite elegant) approach. Starting from a coloring, we make a new proposal as follows:

(i) Choose a vertex of G uniformly at random.

(ii) Change the value of the current coloring at the vertex by drawing a color uniformly from $\{1, \ldots, k\}$.

Remember that we can write the uniform distribution on Ω, the set of proper k-colorings, in terms of indicator functions, so $(\boldsymbol{\pi})_i = \mathbb{I}(i \in \Omega)/|\Omega|$ for all $i \in \Omega$. Then, given the current value of the Metropolis chain is the coloring x (which is a proper k-coloring) and we propose the changed coloring y, the acceptance probability is given by

$$\alpha(x,y) = \min\{1, \frac{(\boldsymbol{\pi})_y}{(\boldsymbol{\pi})_x}\} = \min\left\{1, \frac{\mathbb{I}(i \in \Omega)/|\Omega|}{1/|\Omega|}\right\} = \begin{cases} 1 & \text{if } y \in \Omega, \\ 0 & \text{otherwise.} \end{cases}$$

Note that, if x is a proper coloring, we can check if y is a proper coloring simply by making sure that the changed color does not cause any clashes.

We also need to check that, provided we start with a proper k-coloring, the resulting chain is irreducible. This is not, in general, the case. But if we make sure $k \geq \Delta(G) + 2$, where $\Delta(G)$ is the maximum degree of a vertex in G, the Metropolis chain will be irreducible on Ω. To see this, observe that the condition $k \geq \Delta(G) + 2$ ensures that any vertex can be given at least two colors without clashing with neighboring vertices. We can then move to any proper coloring as follows. We first fix one vertex (call it v_1). We then 'flip' vertices around this v_1 until we can give v_1 the color we want. Then holding v_1 constant, we choose another vertex v_2 and 'flip' colors until we get the one we want. Continuing in this manner, we can achieve any proper coloring.

Lets consider the concrete example shown in Figure 2.3.2.

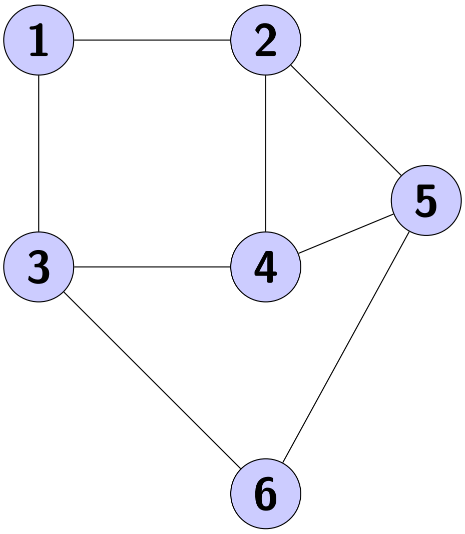

Figure 2.3.2: The graph we wish to color.

If we consider 5-colorings of this graph, we are guaranteed the Metropolis chain will be irreducible. This is because the maximum degree in the graph is 3 and $5 \geq 3+2$. We use the initial coloring $(1, 4, 4, 3, 5, 2)$. The Matlab code for sampling is as follows.

Listing 2.5: R-code for coloring a graph using the Metropolis algorithm

```
N=10
number_of_vertices= 6
k= 4
E= rbind(c(0, 1, 1, 0, 0, 0),
        c(1, 0, 0, 1, 1, 0),
        c(1, 0, 0, 1, 0, 1),
        c(0, 1, 1, 0, 1, 0),
        c(0, 1, 0, 1, 0, 1),
        c(0, 0, 1, 0, 1, 0))

X=matrix(0, nrow=N, ncol=number_of_vertices)
X[1,]=c(1, 4, 4, 3, 1, 2)
for(i in 1:(N-1))
{
  random_vertex= sample(1:number_of_vertices, size=1)
  random_color= sample(1:k, size=1)
  proper_coloring= TRUE
  for (j in 1:number_of_vertices)
  {
    if(E[random_vertex, j]==1 & X[i, j]==random_color)
    {
      proper_coloring=FALSE
    }
  }
  X[i+1, ]=X[i,]
  if(proper_coloring==TRUE)
  {
    X[i+1, random_vertex]= random_color
  }
}
X
```

A nice property of the algorithm we have implemented is that, even if we do not start with a proper k-coloring, the algorithm will eventually reach one and will then only produce proper k-colorings.

2.3.2 The Metropolis-Hastings algorithm

The requirement that Q be symmetric in the Metropolis algorithm is a bit limiting. Fortunately, an extension to the Metropolis algorithm allows us to drop this assumption.

Algorithm 2.3.2 (Metropolis-Hastings Algorithm)**.** Given a distribution, $\boldsymbol{\pi}$, on $\mathcal{X}$, a (not necessarily symmetric) $|\mathcal{X}| \times |\mathcal{X}|$ transition matrix, Q, and an initial distribution $\boldsymbol{\mu}$:

(i) Draw $X_0 \sim \boldsymbol{\mu}$. Set $n = 0$.

(ii) Draw $Y \sim (Q)_{X_n,\cdot}$.

(iii) Calculate

$$\alpha(X_n, Y) = \min\left\{1, \frac{(\boldsymbol{\pi})_Y}{(\boldsymbol{\pi})_{X_n}} \frac{(Q)_{Y,X_n}}{(Q)_{X_n,Y}}\right\}.$$

(iv) With probability $\alpha(X_n, Y)$, set $X_{n+1} = Y$. Otherwise, set $X_{n+1} = X_n$.

(v) Set $n = n + 1$ and repeat from (ii).

2.4 Convergence of Markov Chains

We know the Metropolis algorithm (or the Metropolis-Hastings algorithm) produces a Markov chain, $\{X_n\}_{n \in \mathbb{N}}$, with a transition matrix, P, that is in detailed balance with $\boldsymbol{\pi}$. This means that, if $X_0 \sim \boldsymbol{\pi}$, then $X_n \sim \boldsymbol{\pi}$ for all $n \geq 0$. We also know that, defining $\Omega = \{i \in \mathcal{X} : (\boldsymbol{\pi})_i > 0\}$, if $\mathbb{P}(X_0 \in \Omega) = 1$ and P is irreducible on Ω, then $\boldsymbol{\pi}$ will be the unique stationary distribution of $\{X_n\}_{n \in \mathbb{N}}$. A problem in practice, however, is that we usually do not know how to start the chain from the initial distribution $\boldsymbol{\pi}$ (after all, if it was easy to sample from $\boldsymbol{\pi}$ we would not be using Markov Chain Monte Carlo). What we need to know is that $\mathbb{P}_{\boldsymbol{\mu}}(X_n = i) = (\boldsymbol{\mu} P^n)_i \to (\boldsymbol{\pi})_i$ as $n \to \infty$ for distributions, $\boldsymbol{\mu}$, other than the stationary distribution, $\boldsymbol{\pi}$. We will see that this will be true provided the Markov chain satisfies a number of conditions.

2.4.1 Total variation distance

In order to talk about convergence, we need to be able to measure the distance between two probability distributions. There are lots of ways to measure such a distance. The one that will be most useful for us is called *total variation distance.*

Definition 2.4.1 (Total variation distance)**.** The total variation distance between two probability distributions, $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$, on $\mathcal{X}$ is given by

$$\|\boldsymbol{\mu} - \boldsymbol{\nu}\|_{\mathsf{TV}} = \sup_{A \subset \mathcal{X}} |\boldsymbol{\mu}(A) - \boldsymbol{\nu}(A)|.$$

There are lots of equivalent ways to define total variation distance. One of the most useful is given in the following lemma.

Lemma 2.4.2. For two probability distributions, $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$, defined on $\mathcal{X}$, we have

$$\|\boldsymbol{\mu} - \boldsymbol{\nu}\|_{\mathsf{TV}} = \frac{1}{2} \sum_{i \in \mathcal{X}} |(\boldsymbol{\mu})_i - (\boldsymbol{\nu})_i| = \frac{1}{2} \|\boldsymbol{\mu} - \boldsymbol{\nu}\|_1.$$

Proof. First, define the set $B = \{i \in \mathcal{X} : (\boldsymbol{\mu})_i \geq (\boldsymbol{\nu})_i\}$. Let $A \subset \mathcal{X}$ be an event. Then,

$$\boldsymbol{\mu}(A) - \boldsymbol{\nu}(A) \leq \boldsymbol{\mu}(A \cap B) - \boldsymbol{\nu}(A \cap B) \leq \boldsymbol{\mu}(B) - \boldsymbol{\nu}(B).$$

The first inequality is because, by taking the intersection of A with B, we exclude any state, $j \in A$, such that $(\boldsymbol{\mu})_j - (\boldsymbol{\nu})_j < 0$. The second inequality follows because, by considering all of B rather than just its intersection with A, we are adding states, $k \in B \setminus A$, such that $(\boldsymbol{\mu})_k - (\boldsymbol{\nu})_k \geq 0$. Using a symmetric argument, we have

$$\boldsymbol{\nu}(A) - \boldsymbol{\mu}(A) \leq \boldsymbol{\nu}(B^C) - \boldsymbol{\mu}(B^C).$$

Observe, also, that, as $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ are probability distributions,

$$\boldsymbol{\mu}(B) - \boldsymbol{\nu}(B) = \boldsymbol{\mu}(B^C) - \boldsymbol{\nu}(B^C)$$

Thus, for all $A \subset \mathcal{X}$

$$\begin{aligned}|\boldsymbol{\mu}(A) - \boldsymbol{\nu}(A)| &\leq \boldsymbol{\mu}(B) - \boldsymbol{\nu}(B) = \boldsymbol{\mu}(B^C) - \boldsymbol{\nu}(B^C) \\ &= \frac{1}{2}\left[\boldsymbol{\mu}(B) - \boldsymbol{\nu}(B) + \boldsymbol{\mu}(B^C) - \boldsymbol{\nu}(B^C)\right]\end{aligned}$$

Furthermore, for $A = B$ (or $A = B^C$) we have

$$\begin{aligned}|\boldsymbol{\mu}(A) - \boldsymbol{\nu}(A)| &= \frac{1}{2}\left[\boldsymbol{\mu}(B) - \boldsymbol{\nu}(B) + \boldsymbol{\mu}(B^C) - \boldsymbol{\nu}(B^C)\right] \\ &= \frac{1}{2}\sum_{i \in B}[(\boldsymbol{\mu})_i - (\boldsymbol{\nu})_i] + \sum_{i \in B^C}[(\boldsymbol{\nu})_i - (\boldsymbol{\mu})_i] \\ &= \frac{1}{2}\sum_{i \in \mathcal{X}}|(\boldsymbol{\mu})_i - (\boldsymbol{\nu})_i|\end{aligned}$$

□

As a direct consequence of the proof of Lemma 2.4.2, we have the following lemma.

Lemma 2.4.3. For two probability distributions, $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$, defined on $\mathcal{X}$, we have

$$\|\boldsymbol{\mu} - \boldsymbol{\nu}\|_{\mathsf{TV}} = \sum_{\{i \in \mathcal{X}:(\boldsymbol{\mu})_i \geq (\boldsymbol{\nu})_i\}}[(\boldsymbol{\mu})_i - (\boldsymbol{\nu})_i].$$

Now that we have a way to measure the distance between two distributions, we can be a bit more specific about what type of convergence we want to show. That is, we wish to show that

$$\lim_{n \to \infty}\|\boldsymbol{\mu}P^n - \boldsymbol{\pi}\|_{\mathsf{TV}} = 0,$$

for all distributions, $\boldsymbol{\mu}$, on $\mathcal{X}$.

2.4.2 Periodic and aperiodic states

We know that a Markov chain has a unique stationary distribution if it is irreducible and positive recurrent. In order to show that the distribution of an irreducible and positive recurrent Markov chain, started from an arbitrary distribution, converges to its stationary distribution, we will need one more assumption. To see this, consider the following.

Example 2.4.4 (A positive recurrent and irreducible chain that does not converge for all $\boldsymbol{\mu}$)**.** In Example 1.3.4, we looked at the distribution of a random walk on a square for a number of initial distributions. In particular, we saw that if $X_0 \sim \boldsymbol{\delta}_0$ (i.e., if $\mathbb{P}(X_0 = 0) = 1$), that

$$X_n \sim \begin{cases}(1/2, 0, 1/2, 0), & \text{if } n \text{ even,} \\ (0, 1/2, 0, 1/2), & \text{if } n \text{ odd.}\end{cases}$$

Conversely, for $X_0 \sim \boldsymbol{\delta}_1$, we had

$$X_n \sim \begin{cases} (0, 1/2, 0, 1/2, 0), & \text{if } n \text{ even}, \\ (1/2, 0, 1/2, 0), & \text{if } n \text{ odd}. \end{cases}$$

Now, the unique stationary distribution of this chain is

$$\boldsymbol{\pi} = (1/4, 1/4, 1/4, 1/4).$$

For an arbitrary $n \geq 1$, however, it is easy to see that

$$\|\boldsymbol{\delta}_0 P^n - \boldsymbol{\pi}\| = \frac{1}{2} \sum_{i \in \mathcal{X}} |(\boldsymbol{\delta}_0 P^n)_i - (\boldsymbol{\pi})_i| = 1/2.$$

The value 1/2 does not depend on n, so the distance between the distribution of X_n (when $X_0 = 0$) and the stationary distribution does not get any smaller, no matter how long the chain runs for. In other words, for this chain and initial distribution, we do not have convergence to the stationary distribution. Thus, in order to have a situation where convergence to stationarity happens for any initial distribution, we will need to exclude Markov chains such as the above.

Definition 2.4.5 (Period)**.** The *period*, $d(i)$, of a state $i \in \mathcal{X}$ is given by

$$d(i) = \mathsf{gcd}\{n \geq 0 : (P^n)_{i,i} > 0\},$$

where gcd is the *greatest common divisor*.

Thus, for example, if it is only possible to reach a state in multiples of two steps, we say the state has period 2.

Example 2.4.6. Every state of the chain in Example 2.4.4 has period 2.

Definition 2.4.7 (Aperiodic)**.** A state, $i \in \mathcal{X}$, is said to be *aperiodic* if $d(i) = 1$.

It would be cumbersome if we would have to check the period of every single state in a Markov chain. Fortunately, however, this is not the case. This is because periodicity is a class property (that is, all states in a communicating class have the same period), as the following theorem makes clear.

Theorem 2.4.8. *If $i, j \in \mathcal{X}$ communicate (i.e., $i \leftrightarrow j$), then they have the same period (i.e., $d(i) = d(j)$).*

Proof. As i and j communicate, we know there exist $M > 0$ and $N > 0$ such that $(P^M)_{i,j} > 0$ and $(P^N)_{j,i} > 0$. Thus, we know,

$$(P^{M+N})_{i,i} \geq (P^M)_{i,j}(P^N)_{j,i} > 0,$$

We have the inequality because the path that goes from i to j in M steps, then j to i in N steps, is just one possibility of going from i to i in $N + M$ steps (other paths might also have non-zero probability). Note that the fact that $(P^{M+N})_{i,i} > 0$ implies that $d(i)$ divides $M + N$. Now, for all $k \geq 0$, we have

$$(P^{M+k+N})_{i,i} \geq (P^M)_{i,j}(P^k)_{j,j}(P^N)_{j,i},$$

as the path from i to j in M steps, from j back to j in k steps, and from j to i in N steps is just one way of going from i to i in $M + k + N$ steps (note, however, that it might have 0 probability). For any k such that $(P^k)_{j,j} > 0$, we have that $(P^{M+k+N})_{i,i} > 0$ as $(P^M)_{i,j} > 0$ and $(P^N)_{j,i} > 0$. Thus, $d(i)$ must divide $M + k + N$, which means $d(i)$ must divide k. As this holds for all k such that $(P^k)_{j,j} > 0$, $d(i)$ divides $d(j)$. Repeating the whole argument in the opposite direction, we have that $d(j)$ also divides $d(i)$. Thus $d(i) = d(j)$. □

Definition 2.4.9. Let $P = (P_{i,j})$ a matrix.

(i) P is called non-negative if all entries of P are non-negative.

(ii) A non-negative matrix P is called quasi-positive if there is a natural number $n_0 \geq 1$ such that for $n \geq n_0$ all entries of P^n are positive.

A result in number theory is that if a set of natural numbers is closed under addition and has greatest common divisor 1 then it contains all but a finite number of natural numbers.

Lemma 2.4.10. Let $k = 1, 2, \dots$ be an arbitrary but fixed natural number. Then there is a natural number $n_0 \geq n$ such that

$$\{n_0, n_0 + 1, n_0 + 2, \dots\} \subset \{n_1 k + n_2(k + 1) : n_1, n_2 \in \mathbb{N}\}.$$

Proof. Let $n \geq k^2$. Then there exist $m, d \geq 0$ such that

$$n - k^2 = mk + d \quad d < k$$

(euclidean division). Consequently, we can rewrite n by

$$n = k^2 + mk + d = k(k - d + m) + (k + 1)d$$

which results in the fact that

$$n \in \{n_1 k + n_2(k + 1) : n_1, n_2 \in \mathbb{N}\}$$

where $n_0 = k^2$ is the desired number. □

This leads to the following lemma.

Lemma 2.4.11. If P is an irreducible and aperiodic transition matrix if and only if for arbitrary $i, j \in \mathcal{X}$

$$(P^n)_{i,j} > 0,$$

for all sufficiently large n.

Proof. $\Rightarrow$ is easy beacuase of the definitions of aperiodicity and irreducibility.
$\Leftarrow$ P is aperiodic and irreducible i.e. for $i \in \mathcal{X}$ we denote

$$J(i) = \{n \geq 1 | P^n_{i,i} > 0\}$$

with gcd $d(i) = 1$. With Corollary 1.3.2 we have

$$P^{m+n}_{i,i} \geq P^m_{i,i} P^n_{i,i}$$

and consequently if $n, m \in J(i)$ it follows that $m + n \in J(i)$.
We now assume that $J(i)$ contains two successive numbers $k, k+1$ (we will prove this assumption in the end). Then it follows that

$$n_1 k + n_2(k+1) \in J(i) \forall n_1, n_2 \in \mathbb{N}$$

and by 2.4.10 there exists $n_0 = n_0(i) \in \mathbb{N}$ such that

$$\{n_0, n_0 + 1, n_0 + 2, \ldots\} \in J(i).$$

Because of the irreducibility of P we know that for arbitrary $j, l \in \mathcal{X}$ there exist $m, r \in \mathbb{N}$ such that

$$\begin{aligned} P_{j,l}^{m+n_0(i)+r} &\geq P_{j,i}^m P_{i,i}^{n_0(i)} P_{i,l}^r > 0 \\ \Rightarrow P_{j,l}^{m+n_0(i)+1+r} &\geq P_{j,i}^m P_{i,i}^{n_0(i)+1} P_{i,l}^r > 0 \end{aligned}$$

With $n(jl) = m + n_0(i) + 1 + r$ it follows

$$\{n(jl), n(jl)+1, \ldots\} \in J(jl) = \{n \geq 1 | P_{j,l}^n > 0\}.$$

Since this was shown for arbitrary j, l and i we get that P is quasi positive.
Now we prove the assumption. Therefore assume that $J(i)$ does not contain two successive numbers and all elements of $J(i)$ have a minimal distance $d \geq 2$, i.e.:

$$n, n+d \in J(i) \quad gcd(i) = 1.$$

Then d does not divide n otherwise it would exist $m \in \mathbb{N} \cup \{0\}$ with $n = md$ which would result in $gcd(i) = d > 1$ and which is a contradiction to the fact that P is aperiodic. Consequently there exist $m \in \mathbb{N} \cup \{0\}$ amd $k = 1, \ldots, d-1$ such that

$$n = md + k.$$

Moreover, we get by Corollary 1.3.2 that $a(n+d), (b+1)n \in J(i)$ for arbitrary $a, b \in \mathbb{N}$.
We now prove: There exist $a, b \in \mathbb{N}$ such that the difference between the resulting elements is less than d which is a contradiction to the assumption that the minimal distance between two elements of $J(i)$ is greater or equal to d.

$$\begin{aligned} &a(n+d) - (b+1)n \\ &\qquad = a(n+d) - n - bn \\ &\qquad = (a-b)n + ad - md - k \\ &\qquad = (a-b)n + (a-m)d - k \\ &\qquad = d - k < d \end{aligned}$$

where we choose $a = b = m + 1$ in the last equality. □

The class of Markov chains which are irreducible, positive recurrent and aperiodic are the nicest to work with. We call them *ergodic*.

Definition 2.4.12. A transition matrix, P, is called *ergodic* if it is aperiodic, positive recurrent and irreducible.

2.4.3 Coupling

In general it is a bit tricky to talk about the distance between two arbitrary distributions (given we know almost nothing about them). However, there are a number of very useful tools that allow us to bound (and in some cases calculate exactly) total variation distance. One of the most beautiful is the use of coupling method.

Definition 2.4.13 (Coupling)**.** A *coupling* of two distributions, $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$, on $\mathcal{X}$ is a pair of random variables, (X, Y), defined on a common probability space such that the marginal distribution of X is $\boldsymbol{\mu}$ (i.e., $\mathbb{P}(X = i) = (\boldsymbol{\mu})_i$) and the marginal distribution of Y is $\boldsymbol{\nu}$ (i.e., $\mathbb{P}(Y = j) = (\boldsymbol{\nu})_j$).

In other words, we can think of the coupling as a joint distribution on $\mathcal{X} \times \mathcal{X}$ with marginal distributions $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$.

Example 2.4.14 (Coupling two Bernoulli distributions)**.** Consider $\mathcal{X} = \{0, 1\}$ with $\boldsymbol{\mu} = (1/2, 1/2)$ and $\boldsymbol{\nu} = (1/2, 1/2)$. We can couple $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ in a number of ways:

(i) Take X and Y to be independent $\mathsf{Ber}(1/2)$ random variables. Then

$$\mathbb{P}(X = i, Y = j) = 1/4 \quad \text{for all } i, j \in \{0, 1\}.$$

We have $\mathbb{P}(X = i) = \sum_{j\in\{0,1\}} 1/4 = 1/2 = (\boldsymbol{\mu})_i$ and $\mathbb{P}(Y = j) = (\boldsymbol{\nu})_j$.

(ii) Take X to be $\mathsf{Ber}(1/2)$ and set $Y = X$. Then, clearly, $\mathbb{P}(X = i) = (\boldsymbol{\mu})_i$ and $\mathbb{P}(Y = j) = (\boldsymbol{\nu})_j$. But now

$$\mathbb{P}(X = i, Y = j) = \begin{cases} 1/2 & \text{if } i = j = 1, \\ 1/2 & \text{if } i = j = 0, \\ 0 & \text{otherwise.} \end{cases}$$

(iii) Take X to be $\mathsf{Ber}(1/2)$ and set $Y = 1 - X$. Then, clearly, $\mathbb{P}(X = i) = (\boldsymbol{\mu})_i$ and $\mathbb{P}(Y = j) = (\boldsymbol{\nu})_j$. But now

$$\mathbb{P}(X = i, Y = j) = \begin{cases} 1/2 & \text{if } i = 1, j = 0, \\ 1/2 & \text{if } i = 0, j = 1, \\ 0 & \text{otherwise.} \end{cases}$$

Couplings will be very useful to us because they give us a way to bound total variation distance. This is a result of the following theorem.

Theorem 2.4.15. Let $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ be two probability measures on $\mathcal{X}$. Then,

$$\|\boldsymbol{\mu} - \boldsymbol{\nu}\|_{\mathsf{TV}} = \inf\{\mathbb{P}(X \neq Y) : (X, Y) \text{ is a coupling of } \boldsymbol{\mu} \text{ and } \boldsymbol{\nu}\}.$$

Proof. For any coupling, (X, Y), of $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$, and any event $A \subset \mathcal{X}$, we have

$$\begin{aligned}\boldsymbol{\mu}(A) - \boldsymbol{\nu}(A) &= \mathbb{P}(X \in A) - \mathbb{P}(Y \in A)\\ &= \mathbb{P}(X \in A, Y \notin A) - \mathbb{P}(X \notin A, Y \in A)\\ &\leq \mathbb{P}(X \in A, Y \notin A)\\ &\leq \mathbb{P}(X \neq Y).\end{aligned}$$

So

$$\|\boldsymbol{\mu} - \boldsymbol{\nu}\|_{\mathsf{TV}} \leq \inf\{\mathbb{P}(X \neq Y) : (X, Y) \text{ is a coupling of } \boldsymbol{\mu} \text{ and } \boldsymbol{\nu}\}.$$

In order to show the inequality, we explicitly construct a coupling such that $\mathbb{P}(X \neq Y) = \|\boldsymbol{\mu} - \boldsymbol{\nu}\|_{\mathsf{TV}}$. Now, define

$$p = \sum_{i \in \mathcal{X}} \min\{(\boldsymbol{\mu})_i, (\boldsymbol{\nu})_i\}.$$

Observe that

$$\begin{aligned} p &= \sum_{\{i \in \mathcal{X} : (\boldsymbol{\mu})_i \leq (\boldsymbol{\nu})_i\}} (\boldsymbol{\mu})_i + \sum_{\{i \in \mathcal{X} : (\boldsymbol{\mu})_i > (\boldsymbol{\nu})_i\}} (\boldsymbol{\nu})_i\\ &= \sum_{\{i \in \mathcal{X} : (\boldsymbol{\mu})_i \leq (\boldsymbol{\nu})_i\}} (\boldsymbol{\mu})_i + \sum_{\{i \in \mathcal{X} : (\boldsymbol{\mu})_i > (\boldsymbol{\nu})_i\}} (\boldsymbol{\nu})_i + \sum_{\{i \in \mathcal{X} : (\boldsymbol{\mu})_i > (\boldsymbol{\nu})_i\}} (\boldsymbol{\mu})_i - \sum_{\{i \in \mathcal{X} : (\boldsymbol{\mu})_i > (\boldsymbol{\nu})_i\}} (\boldsymbol{\mu})_i\\ &= 1 + \sum_{\{i \in \mathcal{X} : (\boldsymbol{\mu})_i > (\boldsymbol{\nu})_i\}} [(\boldsymbol{\mu})_i - (\boldsymbol{\nu})_i] = 1 - \|\boldsymbol{\mu} - \boldsymbol{\nu}\|_{\mathsf{TV}}.\end{aligned}$$

We then construct the coupling as follows:

(i) Draw $C \sim \mathsf{Ber}(p)$.

(ii) If $C = 1$, draw $Z \sim \boldsymbol{\lambda}$, where

$$(\boldsymbol{\lambda})_i = \frac{\min\{(\boldsymbol{\mu})_i, (\boldsymbol{\nu})_i\}}{p} \qquad \text{for all } i \in \mathcal{X}.$$

Set $X = Y = Z$. (Note that $\boldsymbol{\lambda}$ is a distribution because it is normalized by p).

(iii) If $C = 0$ draw X from $\boldsymbol{\lambda}^X$, where

$$(\boldsymbol{\lambda}^X)_i = \begin{cases} \frac{(\boldsymbol{\mu})_i - (\boldsymbol{\nu})_i}{\|\boldsymbol{\mu} - \boldsymbol{\nu}\|_{\mathsf{TV}}} & \text{if } (\boldsymbol{\mu})_i > (\boldsymbol{\nu})_i,\\ 0 & \text{otherwise,}\end{cases}$$

and Y from $\boldsymbol{\lambda}^Y$, where

$$(\boldsymbol{\lambda}^Y)_i = \begin{cases} \frac{(\boldsymbol{\nu})_i - (\boldsymbol{\mu})_i}{\|\boldsymbol{\nu} - \boldsymbol{\mu}\|_{\mathsf{TV}}} & \text{if } (\boldsymbol{\nu})_i > (\boldsymbol{\mu})_i,\\ 0 & \text{otherwise.}\end{cases}$$

It is easy to check that if X and Y are generated in this way, they have the correct marginals. For X, we have

$$\begin{aligned}\mathbb{P}(X=i) &= p\frac{\min\{(\boldsymbol{\mu})_i,(\boldsymbol{\nu})_i\}}{p}+(1-p)\frac{(\boldsymbol{\mu})_i-(\boldsymbol{\nu})_i}{\|\boldsymbol{\mu}-\boldsymbol{\nu}\|_{\mathsf{TV}}}\mathbb{I}\left((\boldsymbol{\mu})_i>(\boldsymbol{\nu})_i\right)\\ &= \min\{(\boldsymbol{\mu})_i,(\boldsymbol{\nu})_i\}+[(\boldsymbol{\mu})_i>(\boldsymbol{\nu})_i]\,\mathbb{I}\left((\boldsymbol{\mu})_i>(\boldsymbol{\nu})_i\right)=(\boldsymbol{\mu})_i.\end{aligned}$$

A similar proof shows $\mathbb{P}(Y=j)=(\boldsymbol{\nu})_j$. It is clear that if $C=1$ then $\mathbb{P}(X=Y)=1$. If $C=0$, $\mathbb{P}(X=Y)=0$ (because the distributions $\boldsymbol{\lambda}^X$ and $\boldsymbol{\lambda}^Y$ are positive on disjoint sets). Thus,

$$\mathbb{P}(X\neq Y)=\mathbb{P}(C=0)=1-p=\|\boldsymbol{\mu}-\boldsymbol{\nu}\|_{\mathsf{TV}}.$$

□

Example 2.4.16 (A coupling that gives the total variation distance)**.** Let $\mathcal{X}=\{1,2\}$ and consider the distributions $\boldsymbol{\mu}=(1/4,3/4)$ and $\boldsymbol{\nu}=(1/2,1/2)$. It is easy to check that $\|\boldsymbol{\mu}-\boldsymbol{\nu}\|_{\mathsf{TV}}=1/4$. Thus $p=3/4$. We can construct the coupling as follows.

(i) Draw $C\sim\mathsf{Ber}(3/4)$.

(ii) If $C=1$, set $X=Y=Z$, where

$$\mathbb{P}(Z=k)=\begin{cases}1/3 & \text{if } k=1,\\ 2/3 & \text{if } k=2.\end{cases}$$

(iii) If $C=0$, set $X=2$ and $Y=1$.

If it easy to check the marginals are correct. For example

$$\mathbb{P}(X=i)=\begin{cases}p\cdot 1/3=3/4\cdot 1/3=1/4 & \text{if } i=1,\\ p\cdot 2/3+(1-p)=3/4\cdot 2/3+1/4=3/4 & \text{if } i=2.\end{cases}$$

It is also clear (by construction) that $\mathbb{P}(X\neq Y)=1-3/4=\|\boldsymbol{\mu}-\boldsymbol{\nu}\|_{\mathsf{TV}}$.

Although this version of total variation distance might seem very abstract it is actually extremely useful. This is because we usually just need to get an upper bound on total variation distance rather than to calculate it exactly. This means we just need to find a coupling that is 'good enough' (not the one that achieves the infimum).

Example 2.4.17. Consider the two distributions in Example 2.4.14. These two distributions are identical, so the total variation distance is clearly 0. We have three possible bounds, which come from the choice of the different bounds. Using (i), we get an upper bound of $\|\boldsymbol{\mu}-\boldsymbol{\nu}\|_{\mathsf{TV}}\leq 1/2$. Using (ii), we get an upper bound of $\|\boldsymbol{\mu}-\boldsymbol{\nu}\|_{\mathsf{TV}}\leq 0$. Using (iii), we get an upper bound of $\|\boldsymbol{\mu}-\boldsymbol{\nu}\|_{\mathsf{TV}}\leq 1$. Clearly, the bound from (ii) is the best possible (as the total variation distance cannot be negative).

2.4.4 Coupling Markov chains

We can extend the very simple idea of coupling to stochastic processes in general. In particular, we can couple Markov chains. In doing this, we will usually think of two Markov chains with the same transition matrix, P, but different initial distributions. More precisely, we will consider the Markov chain $\{X_n^{\boldsymbol{\mu}}\}_{n\in\mathbb{N}}$, which is Markov $(\boldsymbol{\mu}, P)$, and the Markov chain $\{X_n^{\boldsymbol{\nu}}\}_{n\in\mathbb{N}}$, which is Markov $(\boldsymbol{\nu}, P)$.

In order to couple these chains, we define a process, $\{(X_n^{\boldsymbol{\mu}}, X_n^{\boldsymbol{\nu}})\}_{n\in\mathbb{N}}$, taking values in the space $\mathcal{X} \times \mathcal{X}$ that satisfies the following: if we look at the first coordinate of this process, it behaves like $\{X_n^{\boldsymbol{\mu}}\}_{n\in\mathbb{N}}$; if we look at the second coordinate, it behaves like $\{X_n^{\boldsymbol{\nu}}\}_{n\in\mathbb{N}}$. The next two couplings are canonical examples.

Example 2.4.18 (Independent Coupling)**.** In this coupling, $X_n^{\boldsymbol{\nu}})$ and $\{X_n^{\boldsymbol{\mu}}\}_{n\in\mathbb{N}}$ are allowed to run independently of one another on the same probability space. To do this, we define a Markov chain, $\{Y_n\}_{n\in\mathbb{N}}$, on $\mathcal{X} \times \mathcal{X}$. Its initial distribution, $\boldsymbol{\gamma}$, is given by $(\boldsymbol{\gamma})_{i,j} = (\boldsymbol{\mu})_i(\boldsymbol{\nu})_j$ for all $(i, j) \in \mathcal{X} \times \mathcal{X}$. Its transition matrix is given by Q, where

$$(Q)_{(i,j),(k,l)} = (P)_{(i,k)}(P)_{(j,l)} \quad \text{for all } i, j, k, l \in \mathcal{X}.$$

Example 2.4.19 (The Random Mapping Coupling)**.** Another way to couple two chains is to use their random mapping representations. Recall from Section **??** that a Markov chain with transition matrix P and initial distribution $\boldsymbol{\mu}$ can be constructed by first drawing $X_0 \sim \boldsymbol{\mu}$ and then defining the subsequent values by

$$X_{n+1} = f(X_n, Z_n) \quad \text{for all } n \geq 1,$$

where $\{Z_n\}_{n\in\mathbb{N}}$ is a sequence of iid random variables and $f : \mathcal{X} \times \Lambda \to \mathcal{X}$ (remember these need to be chosen so that $\mathbb{P}(f(i, Z) = j) = (P)_{i,j}$ for all $i, j \in \mathcal{X}$). We construct $\{X_n^{\boldsymbol{\mu}}\}_{n\in\mathbb{N}}$ by drawing $X^{\boldsymbol{\mu}} \sim \boldsymbol{\mu}$ and setting

$$X_{n+1}^{\boldsymbol{\mu}} = f(X_n^{\boldsymbol{\mu}}, Z_n) \quad \text{for all } n \geq 1,$$

Likewise, we draw $X^{\boldsymbol{\nu}} \sim \boldsymbol{\nu}$ and set

$$X_{n+1}^{\boldsymbol{\nu}} = f(X_n^{\boldsymbol{\nu}}, Z_n) \quad \text{for all } n \geq 1.$$

Note that we use the same function, f, and same sequence of iid random variables, $\{Z_n\}_{n\in\mathbb{N}}$, in both constructions. An important feature of this coupling is that if the two chains meet, they will then always move together afterward.

In fact, taking advantage of the strong Markov property, we can always modify a coupling so that once two chains meet they always run together: we run the chains using the original coupling until they meet, then use the random mapping coupling.

Definition 2.4.20. Given a coupling of $\{X_n^{\boldsymbol{\mu}}\}_{n\in\mathbb{N}}$ and $X^{\boldsymbol{\nu}} \sim \boldsymbol{\nu}$ such that $X_n^{\boldsymbol{\mu}} = X_n^{\boldsymbol{\nu}}$ for all $n \geq m$, where m is the first time the chains meet, we define the *coupling time* by

$$\tau_{\mathsf{couple}} = \inf\{n : X_n^{\boldsymbol{\mu}} = X_n^{\boldsymbol{\nu}}\},$$

where, as usual, we take $\inf\{\emptyset\} = \infty$.

Using the coupling time, we can bound the total variation distance between the distribution of $X_n^{\boldsymbol{\mu}}$ and $X_n^{\boldsymbol{\nu}}$ by a function of n.

Theorem 2.4.21. Given a coupling of $\{X_n^{\boldsymbol{\mu}}\}_{n\in\mathbb{N}}$ and $\{X_n^{\boldsymbol{\nu}}\}_{n\in\mathbb{N}}$ such that $X_n^{\boldsymbol{\mu}} = X_n^{\boldsymbol{\nu}}$ for all $n \geq m$, where m is the first time the chains meet, it holds that

$$\|\boldsymbol{\mu}P^n - \boldsymbol{\nu}P^n\|_{\mathsf{TV}} \leq \mathbb{P}(\tau_{\mathsf{couple}} > n).$$

Proof. We know that $(\boldsymbol{\nu}P^n)_i = \mathbb{P}(X_n^{\boldsymbol{\mu}} = i)$ and $(\boldsymbol{\nu}P^n)_j = \mathbb{P}(X_n^{\boldsymbol{\nu}} = j)$, so $(X_n^{\boldsymbol{\mu}}, X_n^{\boldsymbol{\nu}})$ is a coupling of $\boldsymbol{\mu}P^n$ and $\boldsymbol{\nu}P^n$. Thus, by Theorem 2.4.15, we have

$$\|\boldsymbol{\mu}P^n - \boldsymbol{\nu}P^n\|_{\mathsf{TV}} \leq \mathbb{P}(X_n^{\boldsymbol{\mu}} \neq X_n^{\boldsymbol{\nu}}).$$

The proof is thus complete as, by construction, $\mathbb{P}(X_n^{\boldsymbol{\mu}} \neq X_n^{\boldsymbol{\nu}}) = \mathbb{P}(\tau_{\mathsf{couple}} > n)$. □

Now that we have these tools, there is a clear method by which we can show the convergence theorem (which is our final goal):

(i) Construct a coupling of $\{X_n^{\boldsymbol{\mu}}\}_{n\in\mathbb{N}}$ and $\{X_n^{\boldsymbol{\nu}}\}_{n\in\mathbb{N}}$.

(ii) Show $\mathbb{P}(\tau_{\mathsf{couple}} < \infty) = 1$ (in other words, $\mathbb{P}(\tau_{\mathsf{couple}} > n) \to 0$.

Theorem 2.4.22 (Convergence to Equilibrium)**.** Let P be an ergodic transition matrix on $\mathcal{X}$. For all probability distributions, $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$, on $\mathcal{X}$

$$\lim_{n\to\infty} \|\boldsymbol{\mu}P^n - \boldsymbol{\nu}P^n\|_{\mathsf{TV}} = 0.$$

In particular, choosing $\boldsymbol{\nu} = \boldsymbol{\pi}$, where $\boldsymbol{\pi}$ is the stationary distribution, we have

$$\lim_{n\to\infty} \|\boldsymbol{\mu}P^n - \boldsymbol{\pi}\|_{\mathsf{TV}} = 0.$$

Proof. We couple $\{X_n^{\boldsymbol{\mu}}\}_{n\in\mathbb{N}}$ and $\{X_n^{\boldsymbol{\nu}}\}_{n\in\mathbb{N}}$ by using the independent coupling until they meet, then using the random mapping coupling. Thus, in order to show the proof, we simply need to show that $\mathbb{P}(\tau_{\mathsf{couple}} < \infty) = 1$.

Recall that we can think of the independent coupling as a Markov chain, $\{Y_n\}_{n\in\mathbb{N}}$, on the state space $\mathcal{X} \times \mathcal{X}$, with transition matrix

$$(Q)_{(i,j),(k,l)} = (P)_{(i,k)}(P)_{(j,l)} \quad \text{for all } i,j,k,l \in \mathcal{X}$$

and initial distribution given by $(\boldsymbol{\gamma})_{i,j} = (\boldsymbol{\mu})_i(\boldsymbol{\nu})_j$ for all $(i,j) \in \mathcal{X} \times \mathcal{X}$. The two chains, $\{X_n^{\boldsymbol{\mu}}\}_{n\in\mathbb{N}}$ and $\{X_n^{\boldsymbol{\nu}}\}_{n\in\mathbb{N}}$, will meet when $\{Y_n\}_{n\in\mathbb{N}}$ enters the diagonal set $D = \{(i,i) : i \in \mathcal{X}\}$. Thus, we simply need to show the first passage time into D is almost surely finite.

Now, because P is aperiodic, we can find an n such that $(P^n)_{i,k} > 0$ and $(P^n)_{j,l} > 0$ and, thus, $(Q^n)_{(i,j),(k,l)} > 0$ for any $i,j,k,l \in \mathcal{X}$. This means that Q is irreducible. In addition, Q has a stationary distribution given by $(\widetilde{\boldsymbol{\pi}})_{(i,j)} = (\boldsymbol{\pi})_i(\boldsymbol{\pi})_j$. By Theorem 1.9.7, this implies that Q is positive recurrent (and, thus, recurrent). By Theorem 1.6.6, this implies that $\mathbb{P}(\tau_{i,j}^F < \infty) = 1$ for all $(i,j) \in \mathcal{X} \times \mathcal{X}$ (and, in particular, for all $(i,i) \in D$. Thus $\mathbb{P}(\tau_D^F < \infty) = 1$, which implies $\mathbb{P}(\tau_{\mathsf{couple}} < \infty) = 1$. Thus

$$\lim_{n\to\infty} \|\boldsymbol{\mu}P^n - \boldsymbol{\nu}P^n\|_{\mathsf{TV}} \leq \mathbb{P}(\tau_{\mathsf{couple}} > n) \to 0.$$

□

Example 2.4.23 (Why the proof does not hold for periodic chains)**.** In understanding why the proof works, it is helpful to consider why it does not work when we drop the assumption of aperiodicity. Consider, again, the random walk on the corners of the square introduced in Example 1.1.3. We saw that this chain is periodic with period 2. Now think about what happens if $\{X_n^{\boldsymbol{\mu}}\}_{n\in\mathbb{N}}$ states in state 0 and $\{X_n^{\boldsymbol{\nu}}\}_{n\in\mathbb{N}}$ starts in state 1. They never meet!!! Thus, $\mathbb{P}(\tau_{\mathsf{couple}} < \infty) < 1$, so it is not true that $\mathbb{P}(\tau_{\mathsf{couple}} > n) \to 0$.

2.5 The Ergodic Theorem

In the exercises, you have used sample averages to estimate expected values on a number of occasions. When the random variables being considered are iid, this is justified by the Strong Law of Large Numbers (Theorem **??**). However, when dealing with Markov chains, the strong law of large numbers does not apply. This is because the random variables are no longer independent or identically distributed. Fortunately, there is a version of the SLLN for Markov chains. This is called the *ergodic theorem.* In order to introduce this theorem, we need the following definition.

Definition 2.5.1. We denote by $V_i(n)$ the number of visits to $i \in \mathcal{X}$ before time n. That is,

$$V_i(n) = \sum_{k=0}^{n-1} \mathbb{I}(X_k = i).$$

We denote by $\tau_i^{(n)}$ the n-th passage time to i. It is $\tau_i^{(0)} = 0$ and $\tau_i^F = \tau_i^{(1)}$ and

$$\tau_i^{(n+1)} = \inf\{n > \tau_i^{(n)} : X_n = i\}.$$

We denote by $S_i^{(n)}$ the length of the n-th excursion to i:

$$S_i^{(n)} = \begin{cases} \tau_i^{(n)} - \tau_i^{(n-1)} & \tau^{(n-1)} < \infty \\ 0 & \text{otherwise} \end{cases}.$$

Lemma 2.5.2. For $n = 2, 3, \dots$ conditionally on $\tau_i^{(n-1)} < \infty$, $S_i^{(n)}$ is independent of $\{X_m : m \le \tau_i^{(n-1)}\}$ and it holds:

$$\mathbb{P}(S_i^{(n)} = r | \tau_i^{(n-1)} < \infty) = \mathbb{P}(\tau_i^F = r | X_0 = i)$$

and therefore

$$\mathbb{E}[S_i^{(n)} | X_0 = i] = \mathbb{E}[\tau_i^F | X_0 = i] = m_i.$$

Proof. Apply the Strong Markov property for the stopping time $\tau = \tau_i^{(n-1)}$.
It is $X_\tau = i$ for $\tau < \infty$ and consequently is $(X_{\tau+m})_{m\in\mathbb{N}}$ $Mar(\delta_i, P)$ and independent of $X_0, \dots, X_\tau$.
Moreover, $S_i^{(n)}$ can be rewritten by

$$S_i^{(n)} = \inf\{m \ge 1 | X_{\tau+m} = i\}$$

so that $S_i^{(n)}$ is the first passage time of $(X_{\tau+m})_{m\in\mathbb{N}}$ to state i. The fact follows by the fact that the considered Markvo chains are homogeneous. □

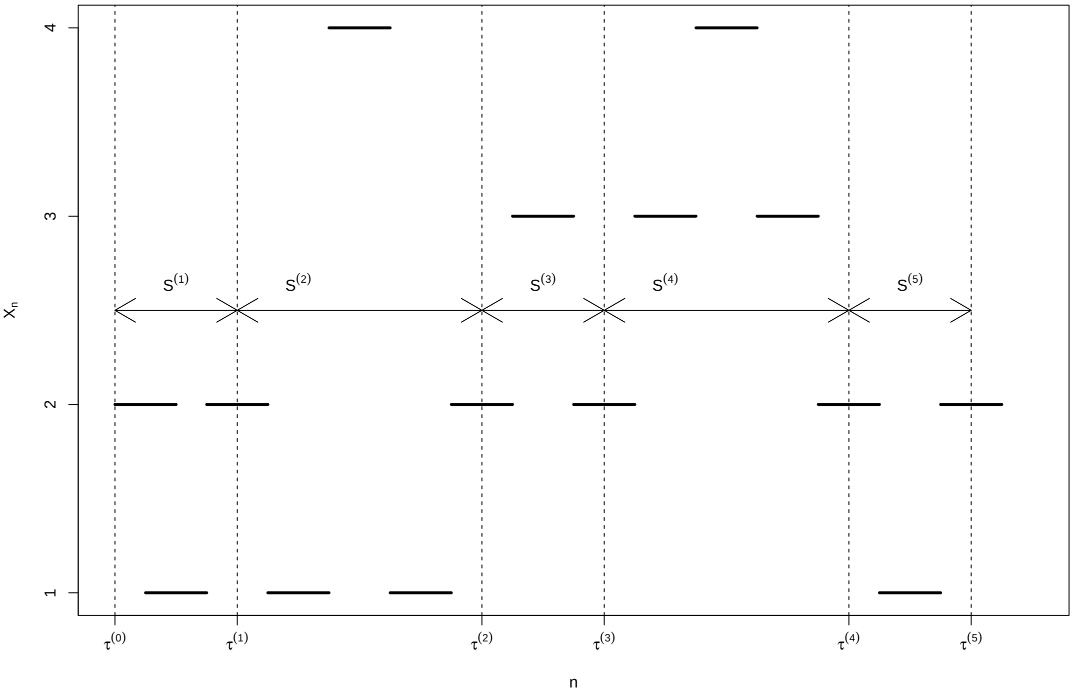

Figure 2.5.1: Illustration of terms defined in Definition 2.5.1 for $n = 15$ $i = 2$ and $V_2(n) = 6$.

Theorem 2.5.3 (Ergodic Theorem)**.** Let P be irreducible and μ be an arbitrary distribution. Then, if $\{X_n\}_{n\in\mathbb{N}}$ is Markov$(\boldsymbol{\mu}, P)$,

$$\mathbb{P}\left(\frac{V_i(n)}{n} \to \frac{1}{m_i} \text{ as } n \to \infty\right) = 1,$$

where $i \in \mathcal{X}$ and $m_i = \mathbb{E}_i\tau_i^F$. If P is also positive recurrent with stationary distribution $\vec{\pi}$, then, for any bounded function $f : \mathcal{X} \to \mathbb{R}$,

$$\mathbb{P}\left(\frac{1}{n}\sum_{k=0}^{n-1} f(X_k) \to \bar{f} \text{ as } n \to \infty\right) = 1,$$

where $\bar{f} = \sum_{i\in\mathcal{X}} f(i)(\boldsymbol{\pi})_i$.

$V_i(n)/n$ is the proportion of time spent in i before time n.

Proof.

Part 1. P is either transient or recurrent. If P is transient, then the number of visits to i is (by the definition of transience) almost surely finite. So,

$$\frac{V_i(n)}{n} \to 0 = \frac{1}{m_i}.$$

Suppose, then, that P is recurrent. Then, by the strong Markov property, $\{X_{\tau_i^F+n}\}_{n\in\mathbb{N}}$ is Markov$(\boldsymbol{\delta}_i, P)$ and independent of $X_0, \ldots, X_{\tau_i^F}$. Furthermore, the asymptotic proportion of time spent in i is the same for $\{X_n\}_{n\in\mathbb{N}}$ and $\{X_{\tau_i^F+n}\}_{n\in\mathbb{N}}$, so we can assume (without altering our statements about asymptotic proportions) that $\boldsymbol{\mu} = \boldsymbol{\delta}_i$.

Now, let $\tau_i^{(n)}$ be the time of the nth passage into i. That is, we define $\tau_i^{(0)} = 0$ and $\tau^{(n+1)} = \inf\{n > \tau_i^{(n)} : X_n = i\}$ for all $n \geq 0$. Then define $S_i(n)$ to be the length of the nth excursion to i. That is,

$$S_i^{(n)} = \begin{cases} \tau_i^{(n)} - \tau_i^{(n-1)} & \text{if } \tau_i^{n-1} < \infty, \\ 0 & \text{otherwise.} \end{cases}$$

We then have the following inequalities (draw a picture to help understand them)

$$S_i^{(1)} + \cdots + S_i^{(V_i(n)-1)} \leq n - 1,$$

and

$$S_i^{(1)} + \cdots + S_i^{(V_i(n))} \geq n.$$

So,

$$\frac{S_i^{(1)} + \cdots + S_i^{(V_i(n)-1)}}{V_i(n)} \leq \frac{n}{V_i(n)} \leq \frac{S_i^{(1)} + \cdots + S_i^{(V_i(n))}}{V_i(n)}. \tag{2.2}$$

Now, by the strong Markov property, the $\{S_i(n)\}_{n\geq 1}$ is clearly an iid sequence. Thus, by the SLLN,

$$\mathbb{P}\left(\frac{S_i^{(1)} + \cdots + S_i^{(n)}}{n} \to m_i \text{ as } n \to \infty\right) = 1,$$

as $\mathbb{E}S_i^{(1)} = m_i$. As P is recurrent,

$$\mathbb{P}(V_i(n) \to \infty \text{ as } n \to \infty) = 1.$$

So,

$$\frac{S_i^{(1)} + \cdots + S_i^{(V_i(n))}}{V_i(n)} \to m_i$$

almost surely, and

$$\frac{S_i^{(1)} + \cdots + S_i^{(V_i(n)-1)}}{V_i(n)} = \frac{S_i^{(1)} + \cdots + S_i^{(V_i(n))}}{V_i(n)} - \frac{S_i^{V_i(n)}}{V_i(n)} \to m_i$$

almost surely. Thus, using the squeeze theorem in (2.2), we have

$$\frac{n}{V_i(n)} \to m_i$$

almost surely, which implies

$$\frac{V_i(n)}{n} \to \frac{1}{m_i}$$

almost surely.

Part 2. As P is assumed to be irreducible and positive recurrent, we know P has a unique stationary distribution $\boldsymbol{\pi}$. To simplify things, we assume without loss of generality that $|f| < 1$. Now, for any $J \subseteq \mathcal{X}$, we have

$$\begin{aligned}\left|\frac{1}{n}\sum_{k=0}^{n-1} f(X_k) - \bar{f}\right| &= \left|\sum_{i\in\mathcal{X}}\left(\frac{V_i(n)}{n} - (\boldsymbol{\pi})_i\right) f(i)\right| \\ &\leq \sum_{i\in\mathcal{X}}\left|\frac{V_i(n)}{n} - (\boldsymbol{\pi})_i\right| \\ &\leq \sum_{i\in J}\left|\frac{V_i(n)}{n} - (\boldsymbol{\pi})_i\right| + \sum_{i\in\mathcal{X}\setminus J}\left|\frac{V_i(n)}{n} - (\boldsymbol{\pi})_i\right| \\ &\leq \sum_{i\in J}\left|\frac{V_i(n)}{n} - (\boldsymbol{\pi})_i\right| + \sum_{i\in\mathcal{X}\setminus J}\left(\frac{V_i(n)}{n} + (\boldsymbol{\pi})_i\right) \\ &\leq 2\sum_{i\in J}\left|\frac{V_i(n)}{n} - (\boldsymbol{\pi})_i\right| + 2\sum_{i\in\mathcal{X}\setminus J}(\boldsymbol{\pi})_i,\end{aligned}$$

Given $\epsilon > 0$, we can choose J finite so that

$$\sum_{i\in\mathcal{X}\setminus J}(\boldsymbol{\pi})_i < \frac{\epsilon}{4}.$$

We know that $V_i(n)/n \to 1/m_i = (\boldsymbol{\pi})_i$ almost surely for any $i \in \mathcal{X}$. Thus, for a given $i \in J$, we can almost surely find an N_i (which is random) so that, for all $n \geq N_i$,

$$\left|\frac{V_i(n)}{n} - (\boldsymbol{\pi})_i\right| < \frac{\epsilon}{4|J|}.$$

Choosing $N = \max_{i\in J} N_i$, we have

$$\sum_{i\in J}\left|\frac{V_i(n)}{n} - (\boldsymbol{\pi})_i\right| < \frac{\epsilon}{4}.$$

for all $n \geq N$. Thus, for any $\epsilon > 0$, there almost surely exists N and J such that

$$\left|\frac{1}{n}\sum_{k=0}^{n-1} f(X_k) - \bar{f}\right| < \epsilon.$$

□

The ergodic theorem is very important for us. In particular, it shows that if we have a Markov chain $\{X_n\}_{n\in\mathbb{N}}$ with stationary distribution $\boldsymbol{\pi}$, then

$$\frac{1}{n}\sum_{k=0}^{n-1} f(X_k)$$

is a *consistent* estimator of $\mathbb{E}f(X)$, where $X \sim \boldsymbol{\pi}$. However, it is very important to note that the Ergodic theorem is an asymptotic theorem — as the sample size increases to infinity, everything will work well, but for finite sampling sizes estimators might still not be very accurate.

Example 2.5.4 (Estimators behaving badly)**.** Consider the Markov chain illustrated in Figure 4.0.1.

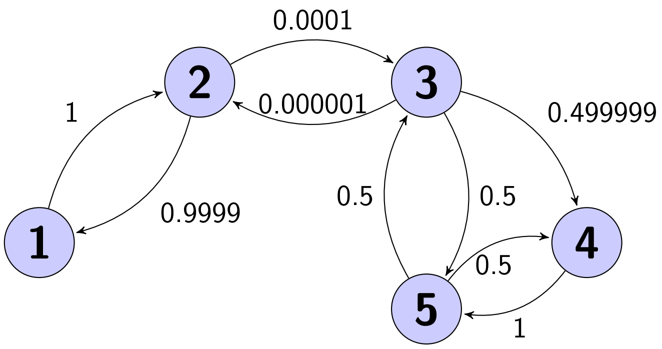

Figure 2.5.2: The transition graph of the Markov chain.

Let $f(\cdot) = \mathbb{I}(\cdot \in \{3,4,5\})$ and suppose that we want to estimate

$$\bar{f} = \mathbb{E}f(X) = \mathbb{P}(X \in \{3,4,5\}),$$

where $X \sim \boldsymbol{\pi}$ with $\boldsymbol{\pi}$ the stationary distribution of the chain. Using the ergodic theorem, we know that

$$\widehat{f} = \frac{1}{n}\sum_{k=0}^{n-1} \mathbb{I}(X_k \in \{3,4,5\})$$

is a consistent estimator of $\bar{f}$. Thus, we can estimate $\bar{f}$ using the following code.

Listing 2.6: Estimating $\bar{f}$

```
n= 10^6
rho=0.1 # for burn-in
X=rep(0, n)
f= rep(0, n)

P=rbind(c(0,1,0,0,0),
 c(0.9999,0,0.0001,0,0),
 c(0,0.000001,0,0.499999,0.5),
 c(0,0,0,0,1),
 c(0,0,0.5,0.5,0))
X0= 3 # starting point of markov chain
X=c() # store values of markov chain
f= c()# store the values of f evaluated in components of X
Z=runif(1, min=0, max=1)
X[1]=min(which(Z<=cumsum(P[X0,]) ))
f[1]= (X[1]==3) + (X[1]==4) + (X[1]==5)
for(i in 1:(n-1))
{
  Z=runif(1, min=0, max=1)
  X[i+1]=min(which(Z<=cumsum(P[X[i],]) ))
  f[i+1]= (X[i+1]==3) + (X[i+1]==4) + (X[i+1]==5)
}
mean(f)
mean(f[(rho*n):n]) # burn in estimator
i
```

The stationary distribution of the Markov chain is given by

$$\boldsymbol{\pi} \approx (0.0028, 0.0028, 0.2841, 0.1420, 0.5682),$$

so we know $\bar{f} \approx 0.9943$. However, starting the chain with $X_0 = 1$, I got an estimate of $\widehat{f} = 0$ using a sample of size $n = 10^3$. For $n = 10^5$, I got an estimate of $\widehat{f} = 0.72$ and, for $n = 10^6$, I got $\widehat{f} = 0.9758$. In contrast, for $X_0 = 3$, an estimate of $\widehat{f} = 1$ was obtained using a sample size of $n = 10^6$. Note that none of these estimators are even remotely accurate.

The reason why the estimator does not work well is that the Markov chain simply does not move around enough. If it starts in $\{1, 2\}$, it takes a long time on average before it reaches $\{3, 4, 5\}$. If it starts in $\{3, 4, 5\}$, it takes an even longer time on average before it moves to $\{1, 2\}$. As a result, the chain takes too long to sample enough values from both parts of the state space.

Situations where a Markov chain spends too much time sampling from one high probability region of a state space (at the expense of other high probability regions) often occur in MCMC sampling. We will discuss a number of methods for avoiding such situations in Section 4.

Chapter 3

Markov and Gibb's Random Fields

3.1 Markov Random Fields

Markov chains (and stochastic processes in general) are, in some sense, just a collection of ordered random variables. In this chapter, we think about extending the idea of random variables arranged on a line to random variables arranged spatially. Such arrangements of random variables are called *random fields*. To remove a lot of technical difficulties, we will only consider special cases of random fields: those with a finite number of random variables, which each take only countable values.

Definition 3.1.1 ((Discrete) Random Field)**.** Let S be a finite set and Λ be a countable set (called the *phase space*). A *random field on* S *with phases in* Λ is a collection of random variables $X = \{X_s\}_{s \in S}$, indexed by S and taking values in Λ.

Alternatively (and often usefully) we can think of such a random field as a random variable taking values in the *configuration space* Λ^S.

Example 3.1.2 (Temperature in Germany)**.** Suppose we want to model the maximum temperature in a number of southern German cities on a given day. We measure only the number of degrees (Celsius), so the temperature in a given city is a discrete random variable taking values in $\Lambda = \{-50, -49, \ldots, 60\}$ (hopefully, we do not see temperature outside this range!). The set S is here given by, say,

$$\begin{aligned} S &= \{\text{Stuttgart, Ulm, Freiburg, Memmingen, Augsburg, Munich, Nuremberg}\} \\ &= \{1, 2, \ldots, 7\}, \end{aligned}$$

where we label everything with numbers to simplify notation. The temperatures are then given by seven random variables, $X_1, \ldots, X_7$. Now, clearly, the temperature in Ulm, X_2, is related to the temperatures of nearby cities. However, this relationship cannot be modeled by arranging the random variables on a line. Instead, we need spatial structure and thus model things using a random field.

It is often convenient to consider only a subset, $A \subset S$, of the random variables in the random field.

Definition 3.1.3 (Restriction of X to A)**.** For $A \subset S$, we define

$$X_A = \{X_s\}_{s \in A}$$

to be the *restriction of* X *to* A.

As was the case with stochastic processes, we need some form of structure in order to carry out mathematical analysis of random fields. In particular, we will consider random fields with a spatial analogue of the Markov property. In order to define this property, we need to clarify the notion of conditional independence.

Definition 3.1.4 (Conditional Independence)**.** Given events A, B and C, we say that A and B are *conditionally independent* given C if

$$\mathbb{P}(A \cap B \,|\, C) = \mathbb{P}(A \,|\, C)\mathbb{P}(B \,|\, C),$$

or, equivalently,

$$\mathbb{P}(A \,|\, B, C) = \mathbb{P}(A \,|\, C).$$

Example 3.1.5. Consider a Markov chain $\{X_n\}_{n \in \mathbb{N}}$. Then, by the Markov property, X_{N+1} is conditionally independent of $\{X_n\}_{n=0}^{N-1}$ given X_N.

In the spatial context, we will define an analogue of the Markov property by considering random fields where the component random variables are conditionally independent of the Markov random field given the values of nearby random variables. In order to be clear what we mean by nearby random variables, we define a *neighborhood system* on S.

Definition 3.1.6. A neighborhood system on S is a family $\mathcal{N} = \{\mathcal{N}_s\}_{s \in S}$ of subsets of S such that

(i) $s \notin \mathcal{N}_s$ for all $s \in S$,

(ii) $t \in \mathcal{N}_s \Rightarrow s \in \mathcal{N}_t$ for all $s, t \in S$.

We can represent the tuple $(S, \mathcal{N})$ using an undirected graph, $G = (V, E)$, where $V = S$ and edges are placed between all pairs $s, t \in S$ such that $t \in \mathcal{N}_s$.

Example 3.1.7 (Example 3.1.2 cont.)**.** When considering temperatures in southern German cities, it makes sense to define a neighborhood system based on geography. We do this by drawing a graph (here this is easier than trying to specify the neighborhood system directly). Figure 3.1.1 shows one such neighborhood system for the cities in southern Germany.

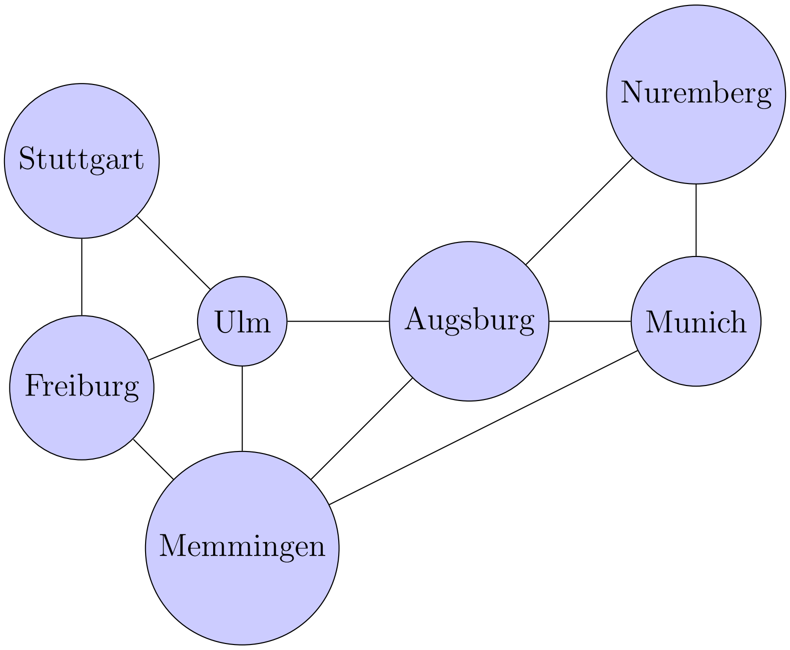

Figure 3.1.1: Graph showing a neighborhood system for a random field describing the maximum temperature in southern German cities.

Often, instead of thinking in the abstract terms of a neighborhood system, we will simply think of the equivalent graph. Using the neighborhood system, we can be precise about what we mean by a spatial Markov property.

Definition 3.1.8 (Markov Random Field)**.** A random field, X, is a *Markov random field* (MRF) with respect to the neighborhood system $\mathcal{N}$ if X_s and $X_{S\setminus(\mathcal{N}_s \cup \{s\})}$ are conditionally independent given $X_{\mathcal{N}_s}$. That is, for all $s \in S$,

$$\mathbb{P}(X_s = x_s \,|\, X_{S\setminus\{s\}} = x_{S\setminus\{s\}}) = \mathbb{P}(X_s = x_s \,|\, X_{\mathcal{N}_s} = x_{\mathcal{N}_s}).$$

Example 3.1.9 (Example 3.1.2 cont.)**.** Let us treat the temperatures in the German cities as an MRF. Then, for example, we see that the temperature in Ulm is conditionally independent of the temperature in Munich given the temperatures in Stuttgart, Freiburg, Memmingen and Augsburg. Suppose we did not know the temperature in Ulm, but we did know the temperatures in the other six cities. If the temperatures form an MRF, then all the information about the temperature in Ulm is encapsulated in the temperatures in Stuttgart, Freiburg, Memmingen and Augsburg. The temperatures in Munich and Nuremberg do not give us any additional information. In other words, if we know the temperature in a 'neighborhood' of Ulm, then we do not need to know the temperatures further away.

When talking about random fields, we consider a number of different distributions.

Definition 3.1.10. Consider a random field $X = \{X_s\}_{s\in S}$ and enumerate S so that $S = \{1, \dots, K\}$. We then consider the distribution of the whole field (the *joint distribution*)

$$\pi(x) = \mathbb{P}(X_1 = x_1, \dots, X_K = x_K),$$

this is (equivalently) the distribution of the random variable X taking values in Λ^S. Note the notation is a little strange here: the x is actually a vector of the form $x = (x_1, \dots, x_K)$. We also consider the *marginal distributions* of the components of X. For these, we write

$$\pi_s(x_s) = \mathbb{P}(X_s = x_s).$$

Lastly, we consider the *conditional distributions* of the form

$$\pi_s^L(x_s \,|\, x_{S\setminus\{s\}}) = \mathbb{P}(X_s = x_s \,|\, X_1 = x_1, \dots, X_{s-1} = x_{s-1}, X_{s+1} = x_{s+1}, \dots, X_K = x_K),$$

In the case of the MRFs, the conditional distributions have a particular importance. This is because, for an MRF

$$\begin{aligned}\pi_s^L(x_s \,|\, x_{S\setminus\{s\}}) &= \mathbb{P}(X_s = x_s \,|\, X_1 = x_1, \dots, X_{s-1} = x_{s-1}, X_{s+1} = x_{s+1}, \dots, X_K = x_K) \\ &= \mathbb{P}(X_s = x_s \,|\, X_{\mathcal{N}_s} = x_{\mathcal{N}_s}).\end{aligned}$$

Definition 3.1.11. Given an MRF on S, $\pi_s^L(x_s \,|\, X_{S\setminus s})$ is called the *local characteristic* of the MRF at site s. The family of local characteristics, $\{\pi_s^L\}_{s\in S}$ is called the *local specification* of the MRF.

As we will see later, the local specification will give us a powerful method for simulating MRFs. This is based on the fact that, under a technical condition, the local specification of an MRF completely determines its distribution, π. To show this, we essentially just need to show that if we know $\mathbb{P}(X = x \,|\, Y = y)$ and $\mathbb{P}(Y = y \,|\, X = x)$, we can work out $\mathbb{P}(X = x, Y = y)$.

Theorem 3.1.12. The distribution of an MRF, X, is completely determined by its local specification so long as $\pi(x) > 0$ for all $x \in \Lambda^S$.

Proof. We see this via the identity

$$\pi(x) = \prod_{i=1}^{K} \frac{\pi^L(x_i \,|\, x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_K)}{\pi^L(y_i \,|\, x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_K)} \pi(y) \propto \prod_{i=1}^{K} \frac{\pi^L(x_i \,|\, x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_K)}{\pi^L(y_i \,|\, x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_K)},$$

for all $x, y \in \Lambda^S$. This identify seems a bit strange, but what it says is that if we know $\{\pi_s^L(x_s \,|\, x_{S\setminus s})\}_{s\in S}$ then we can calculate $\pi(x)$ for all $x \in \Lambda^S$ by first fixing some point $y \in \Lambda^S$ (it could be anything), then calculating these products of ratios of conditional distributions. Afterward we can normalize everything to get probability distributions.

To show the identity, observe that

$$\begin{aligned}\pi(x) &= \mathbb{P}(X_1 = x_1, \dots, X_K = x_K) \\ &= \mathbb{P}(X_K = x_K \,|\, X_1 = x_1, \dots, X_{K-1} = x_{K-1})\mathbb{P}(X_1 = x_1, \dots, X_{K-1} = x_{K-1}) \\ &= \mathbb{P}(X_K = x_K \,|\, X_1 = x_1, \dots, X_{K-1} = x_{K-1}) \frac{\mathbb{P}(X_1 = x_1, \dots, X_{K-1} = x_{K-1}, X_K = y_k)}{\mathbb{P}(X_K = y_k \,|\, X_1 = x_1, \dots, X_{K-1} = x_{K-1})} \\ &= \frac{\pi^L(x_K \,|\, x_1, \dots, x_{K-1})}{\pi^L(y_K \,|\, x_1, \dots, x_{K-1})} \pi(x_1, \dots, x_{K-1}, y_K).\end{aligned}$$

Repeating the procedure for $\pi(x_1, \dots, x_{K-1}, y_K)$ and pulling out x_{K-1} we get

$$\pi(x) = \frac{\pi^L(x_{K-1} \,|\, x_1, \dots, x_{K-2}, y_K)}{\pi^L(y_{K-1} \,|\, x_1, \dots, x_{K-2}, y_K)} \frac{\pi^L(x_K \,|\, x_1, \dots, x_{K-1})}{\pi^L(y_K \,|\, x_1, \dots, x_{K-1})} \pi(x_1, \dots, y_{K-1}, y_K).$$

Continuing on, we get the identity. □

3.2 Gibb's Random Fields

Many random fields that we wish to consider will have distributions, π, of the form

$$\pi_T(x) = \frac{1}{Z_T} \exp\left\{ -\frac{1}{T}\mathcal{E}(x) \right\}, \tag{3.1}$$

where $T > 0$ is called the *temperature*, $\mathcal{E}(x)$ is the *energy* of configuration $x \in \Lambda^S$ and

$$Z_T = \sum_{x \in \Lambda^S} \exp\left\{ -\frac{1}{T}\mathcal{E}(x) \right\},$$

is called the *partition function.* These distributions come from statistical physics, where they are often called *Boltzmann distributions* after the great physicist who studied them. Actually, the distribution of any random field can be written in the form (3.1) (with $T = 1$), but we will just consider cases where this is natural. Note that the terminology and notation in area varies a fair bit from author to author (in particular, mathematicians are not quite true to the original physical meanings of many of these terms). However, once you understand things, it is not too hard to adapt notation or notice and understand differences in various treatments.

It is worth thinking a little bit about how $\mathcal{E}(x)$ and T shape the value of $\pi(x)$. First, observe that there is a minus sign in front of $\mathcal{E}(x)$, so increasing values of $\mathcal{E}(x)$ will decrease the value of π. This is in accordance with the basic physical principle that low energy configurations are more likely. Second, note that as T increases, the value of $\mathcal{E}(x)$ becomes less important. Indeed, as $T \to \infty$, $\mathcal{E}(x)/T \to 0$, so π will converge to a uniform distribution on Λ^S.

3.2.1 The Form of the Energy Function

In order to say anything meaningful, we need to consider restrictions on the energy function $\mathcal{E}(x)$. In particular, we will consider energy functions that are calculated by considering *cliques* of the graph defined by $(S, \mathcal{N})$.

Definition 3.2.1. A clique is a complete subgraph of a graph.

The energy functions we consider work by assigning values to each clique of $(S, \mathcal{N})$. The functions which assign these values are called a *Gibb's potential.*

Definition 3.2.2 (Gibb's Potential)**.** A Gibb's potential on Λ^S relative to the neighborhood system $\mathcal{N}$ is a collection $\{V_C\}_{C \subset S}$ of functions $V_C : \Lambda^S \to \mathbb{R} \cup \{+\infty\}$ such that

(i) $V_C = 0$ if C is not a clique.

(ii) For all $x, x' \in \Lambda^S$ and all $C \subset S$

$$x_C = x'_C \Rightarrow V_C(x) = V_C(x').$$

Definition 3.2.3. The energy function $\mathcal{E} : \Lambda^S \to \mathbb{R} \cup \{+\infty\}$ is said to derive from the potential $\{V_C\}_{C \subset S}$ if

$$\mathcal{E}(x) = \sum_{C \subset S} V_C(x).$$

3.2.2 The Ising Model

One of the most famous models in statistical physics (and theoretical probability) is the Ising model. This is a very simple model of magnetization. Here, we take $S = \mathbb{Z}_m^2$, where $\mathbb{Z}_m^2 = \{(i,j) \in \mathbb{Z}^2 : i,j \in [1,m]\}$. The neighborhood system is given by

$$\mathcal{N}_s = \{(i,j) : \mathbb{Z}_m^2 : |i - s_1| + |j - s_2| = 1\} \text{ for all } s = (s_1, s_2) \in \mathbb{Z}_m^2$$

The graph of $(\mathcal{N}, S)$ is a *square lattice*. Figure 3.2.1 shows the neighborhood system for $m = 3$.

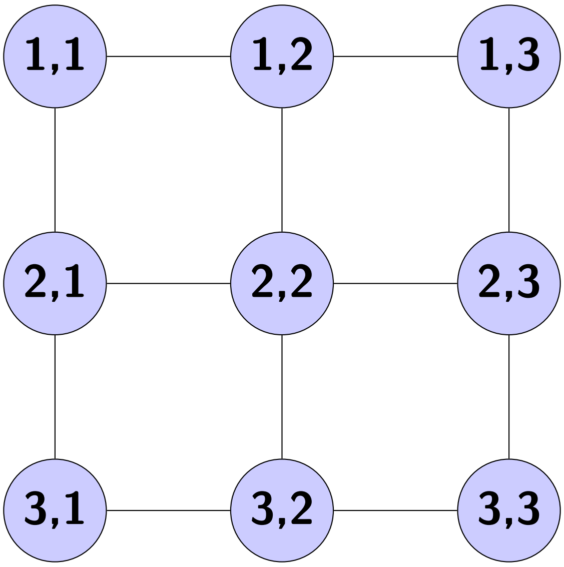

Figure 3.2.1: Graph showing the neighborhood system of the Ising model on $\mathbb{Z}_3^2$.

The $\{X_s\}_{s\in S}$ each take values in $\Lambda = \{-1, 1\}$. The cliques of this graph are the individual nodes and adjacent pairs. The Gibb's potential is given by

$$V_{\{s\}} = -H \cdot x_s \quad \text{for all } s \in S,$$

where $H \in \mathbb{R}$ represents an external magnetic field, and

$$V_{\{(s,t)\}} = -J \cdot x_s x_t \quad \text{for all } (s,t) \in E,$$

where $J \in \mathbb{R}$ controls the strength of interaction between neighbors and E is the edges in the graph induced by the neighborhood system. If $J > 0$ the model is said to be *ferromagnetic* and if $J < 0$ the model is said to be *anti-ferromagnetic*. The energy function is thus given by

$$\mathcal{E}(x) = -J \sum_{(s,t)} x_s x_t - H \sum_{s \in S} x_s.$$

We will, in general, consider very simple cases where $H = 0$ and $J = 1$ (or, sometimes $J = -1$). Then we have a distribution of the form

$$\pi(x) = \frac{1}{Z_T} \exp\left\{ \frac{1}{T} \sum_{(s,t)} x_s x_t \right\}$$

In this model, if $x_s = x_t = 1$ or $x_s = x_t = -1$ then $x_s x_t = 1$. Otherwise, $x_s x_t = -1$. Thus, the configurations with the lower energy / highest probability are those where adjacent variables have the same value (spin). In the case where $J = -1$, the opposite is true: configurations where adjacent variables have opposite spins are more probable.

How do we simulate from the Ising model? The only method we know at the moment is the Metropolis algorithm. One way to implement this is to choose, at each step, a vertex of the current configuration and 'flip it' (replace 1 with -1 and replace -1 with 1). This gives a proposal configuration y. The acceptance probability for this proposal is given by

$$\alpha(x,y) = \frac{\exp\left\{-\frac{1}{T}\mathcal{E}(y)\right\}}{\exp\left\{-\frac{1}{T}\mathcal{E}(x)\right\}} = \exp\left\{-\frac{1}{T}\mathcal{E}(y) + \frac{1}{T}\mathcal{E}(x)\right\}.$$

Using the latter form of the energy function avoids problems associated with calculating exponentials of large numbers. To implement this, we first make a function in Matlab that calculates the energy (according to the Ising model) of a configuration. This function is a little long (because edges and corners need to be considered separately), but is not too complicated.

Listing 3.1: R code for calculating the energy function in the Ising model

```
ising_energy=function(J, H, X)
{
  m= ncol(X)

  single_energies=0
  pair_energies= 0
  for (i in 1:m)
  {
    for(j in 1:m)
    {
      single_energies=single_energies - H*X[i,j]
    }
  }
  for(i in 1:m)
  {
    for(j in 1:m)
    {
      if(i<m & j < m)
      {
        # pair to the right vertex
        pair_energies= pair_energies - J*X[i, j]*X[i, j+1]
        # pair to the below vertex
        pair_energies= pair_energies - J*X[i,j]*X[i+1, j]
      }
      else if (i==m & j < m)
      {
        # pair to the right vertex
        pair_energies= pair_energies - J*X[i, j]*X[i, j+1]
      }

      else if (i<m & j==m)
      {
        # pair to the below vertex
        pair_energies= pair_energies - J*X[i,j]*X[i+1, j]
      }
      else
      {
        pair_energies=pair_energies
      }
    }
  }
  energy= single_energies + pair_energies
  return(energy)
}
X0=matrix(c(-1, -1 , 1 ,-1 ,-1 , 1, 1 , 1 ,-1), ncol=3)
X01=matrix(c(-1, 1, 1, -1), ncol=2)
H0= 0.5
J0= 1
ising_energy(J=J0, H=H0, X=X0)
ising_energy(J=J0, H=H0, X=X01)
```

Using this function, the Metropolis algorithm itself is quite straightforward.

Listing 3.2: R code for simulating the Ising model via the Metropolis algorithm

```
N= 10^3
J0=1
H0=0
T0=2
m=50

X=matrix(rbinom(m^2, size=1, prob=0.5)*2-1, ncol=m, nrow=m)
system.time(
for(k in 1:N)
{
  # choose randomly a coordinate (i,j)
  i= sample(1:m, size=1, replace=TRUE)
  j= sample(1:m, size=1, replace=TRUE)

  Y=X
  Y[i, j]= -1 *Y[i,j] # flip this energy

  alpha= exp(1/T0*(ising_energy(J=J0, H=H0, X=X)- ising_energy(J=J0, H=H0, X=Y)))
  Z= runif(1, 0, 1)

  if(Z< alpha){X=Y}
}
)
library(gplots)
library(RColorBrewer)
```

```
my_palette <- colorRampPalette(c("red", "yellow"))(n = 2)
col_breaks = c(-1, 0, 1)
heatmap.2(X,
         #cellnote = X, # same data set for cell labels
         Rowv=FALSE,
         main = "Ising", # heat map title
         #notecol="black",    # change font color of cell labels to black
         density.info="none", # turns off density plot inside color legend
         trace="none",       # turns off trace lines inside the heat map # "none"
         tracecol="black",
         margins =c(12,9),   # widens margins around plot
         col=my_palette,     # use on color palette defined earlier
         breaks=col_breaks,  # enable color transition at specified limits
         symbreaks=TRUE,
         dendrogram="none",   # only draw a row dendrogram
         Colv="NA")
```

3.2.3 The Relation Between Markov and Gibb's Random Fields

As it turns out, so long as we impose some technical conditions, Markov random fields and Gibb's random fields are actually the same things. This theorem is a celebrated result known as the Hammersley-Clifford theorem. We break this theorem up into a number of separate results. The first shows that Gibb's fields are Markov random fields.

Theorem 3.2.4. If X is a random field with distribution, $\pi(x)$, given by

$$\pi(x) = \frac{1}{Z_T} \exp\left\{-\frac{1}{T}\mathcal{E}(x)\right\},$$

where $\mathcal{E}(\cdot)$ derives from a Gibb's potential, $\{V_C\}_{C\subset S}$, relative to the neighborhood system $\mathcal{N}$, then X is Markovian with respect to the same neighborhood system. Moreover, its local specification is given by

$$\pi_s^L(x_s \,|\, x_{S\setminus\{s\}}) = \frac{\exp\left\{-\frac{1}{T}\sum_{\{C\subset S: s\in C\}} V_C(x)\right\}}{\sum_{\lambda\in\Lambda}\exp\left\{\frac{1}{T}\sum_{\{C\subset S: s\in C\}} V_C(\lambda, x_{S\setminus\{s\}})\right\}}, \tag{3.2}$$

where the notation $\lambda, x_{S\setminus s}$ means that we change the sth element of x to λ.

Proof. Recall that $\pi_s^L(x_s \,|\, x_{S\setminus\{s\}}) = \mathbb{P}(X_s = x_s \,|\, X_{S\setminus\{s\}} = x_{S\setminus\{s\}})$. Now if (3.2) holds, then we have

$$\mathbb{P}(X_s = x_s \,|\, X_{S\setminus\{s\}} = x_{S\setminus\{s\}}) = \frac{\exp\left\{-\frac{1}{T}\sum_{\{C\subset S: s\in C\}} V_C(x)\right\}}{\sum_{\lambda\in\Lambda}\exp\left\{\frac{1}{T}\sum_{\{C\subset S: s\in C\}} V_C(\lambda, x_{S\setminus\{s\}})\right\}}.$$

This only depends on the cliques, C, such that $s \in C$. Now, if $t \in C$ and $s \in C$ then, either $t = s$ or $t \in \mathcal{N}_s$. In other words, the only elements of $\{x_s\}_{s \in S}$ that are considered are x_s and $x_{\mathcal{N}_s}$. Thus,

$$\mathbb{P}(X_s = x_s \,|\, X_{S\setminus\{s\}} = x_{S\setminus\{s\}}) = \mathbb{P}(X_s = x_s \,|\, X_{\mathcal{N}_s} = x_{\mathcal{N}_s}).$$

As a result, in order to prove the theorem it is sufficient to prove (3.2) holds. Observe that we have

$$\mathbb{P}(X_s = x_s \,|\, X_{S\setminus\{s\}} = x_{S\setminus\{s\}}) = \frac{\mathbb{P}(X_s = x_s, X_{S\setminus\{s\}} = x_{S\setminus\{s\}})}{\mathbb{P}(X_{S\setminus\{s\}} = x_{S\setminus\{s\}})} = \frac{\pi(x)}{\sum_{\lambda \in \Lambda} \pi(\lambda, x_{S\setminus\{s\}})}.$$

Now,

$$\pi(x) = \frac{1}{Z_T} \exp\left\{ -\frac{1}{T} \sum_{\{C \subset S : s \in C\}} V_C(x) - \frac{1}{T} \sum_{\{C \subset S : s \notin C\}} V_C(x) \right\},$$

and

$$\sum_{\lambda \in \Lambda} \pi(\lambda, X_{S\in\{s\}}) = \sum_{\lambda \in \Lambda} \frac{1}{Z_T} \exp\left\{ -\frac{1}{T} \sum_{\{C \subset S : s \in C\}} V_C(\lambda, x_{S\setminus\{s\}}) - \frac{1}{T} \sum_{\{C \subset S : s \notin C\}} V_C(x) \right\},$$

where we use the fact that $V_C(x)$ does not depend on λ if $\lambda \notin C$. We then cancel out the

$$\exp\left\{ -\frac{1}{T} \sum_{\{C \subset S : s \notin C\}} V_C(x) \right\}$$

to get the result. □

As we will see, it is much easier to work with Gibb's fields if we work with the local specification given above. We can further simplify the expression by introducing the *local energy.*

Definition 3.2.5 (Local energy)**.** The *local energy* of configuration x at site $s \in S$ is given by

$$\mathcal{E}_s(x) = \sum_{C \subset S : s \in C} V_C(x).$$

Using the local energy, we can write the local specification as

$$\pi_s^L(x_s \,|\, x_{S\setminus\{s\}}) = \frac{\exp\left\{-\frac{1}{T}\mathcal{E}_s(x)\right\}}{\sum_{\lambda \in \Lambda} \exp\left\{-\frac{1}{T}\mathcal{E}_s(\lambda, x_{S\setminus\{s\}})\right\}}.$$

Example 3.2.6 (The simplified Ising model)**.** Consider the simplified Ising model, where $\mathcal{E}(x) = -\sum_{(s,t)} x_s x_t$. In this case, we have

$$\pi_s^L(x_s \,|\, x_{S\setminus\{s\}}) = \frac{\exp\left\{\frac{1}{T}\sum_{\{t : s \sim t\}} x_s x_t\right\}}{\sum_{\lambda \in \Lambda} \exp\left\{\frac{1}{T}\sum_{\{t : s \sim t\}} \lambda x_t\right\}}.$$

Using the fact that λ only takes the values -1 and 1, we have

$$\pi_s^L(x_s \,|\, x_{S\setminus\{s\}}) = \frac{\exp\left\{\frac{1}{T}\sum_{\{t:s\sim t\}} x_s x_t\right\}}{\exp\left\{-\frac{1}{T}\sum_{\{t:s\sim t\}} x_t\right\} + \exp\left\{\frac{1}{T}\sum_{\{t:s\sim t\}} x_t\right\}}.$$

Thus we can write the probability X_s is 1 as

$$\begin{aligned}\mathbb{P}(X_s = 1 \,|\, X_{S\setminus\{s\}} = x_{S\setminus\{s\}}) &= \frac{\exp\left\{\frac{1}{T}\sum_{\{t:s\sim t\}} x_s x_t\right\}}{\exp\left\{-\frac{1}{T}\sum_{\{t:s\sim t\}} x_t\right\} + \exp\left\{\frac{1}{T}\sum_{\{t:s\sim t\}} x_t\right\}} \\ &= \frac{1 + \tanh\left(\frac{1}{T}\sum_{\{t:s\sim t\}} x_t\right)}{2}.\end{aligned}$$

The next theorem shows that, under a technical condition, an MRF is also a Gibb's field.

Theorem 3.2.7. Let π be the distribution of an MRF with respect to $(S, \mathcal{N})$ (where $\pi(x) > 0$ for all $x \in \Lambda^S$). Then,

$$\pi(x) = \frac{1}{Z}\exp\{-\mathcal{E}(x)\},$$

for some energy function $\mathcal{E}(\cdot)$ derived from a Gibb's potential, $\{V_C\}_{C\subset S}$, associated with $(S, \mathcal{N})$.

Proof. See [1]. □

The above theorem shows that an MRF can always be written as a Gibb's field. However, it gives no guarantee that this representation will be unique. Indeed, unless another condition is imposed, the representation will not be unique. To see this, observe that, for $\alpha \in \mathbb{R}$,

$$\begin{aligned}\pi(x) &= \frac{1}{Z_T}\exp\left\{-\frac{1}{T}\mathcal{E}(x)\right\} \\ &= \frac{1}{Z_T}\exp\left\{-\frac{1}{T}\mathcal{E}(x)\right\}\frac{\exp\left\{-\frac{1}{T}\alpha\right\}}{\exp\left\{-\frac{1}{T}\alpha\right\}} \\ &= \frac{1}{Z_T\exp\left\{-\frac{1}{T}\alpha\right\}}\exp\left\{-\frac{1}{T}\left[\mathcal{E}(x) + \alpha\right]\right\} \\ &= \frac{1}{\widetilde{Z}_T}\exp\left\{-\frac{1}{T}\widetilde{\mathcal{E}}(x)\right\}.\end{aligned}$$

In other words, we can always add a constant to the energy function and renormalize accordingly. However, it turns out that if we normalize the potential we get a unique representation.

Definition 3.2.8 (Normalized Potential)**.** A Gibb's potential is *normalized* with respect to a phase $\lambda^* \in \Lambda$ if $V_C(x) = 0$ when there exists an $s \in C$ such that $x_s = \lambda^*$.

Theorem 3.2.9. There exists one and only one potential normalized with respect to a given $\lambda^* \in \Lambda$ corresponding to a Gibb's distribution.

Proof. See [1] Theorem 2.2 and its proof. □

3.3 The Gibb's Sampler

We can use the local specifications of Markov random fields as the basis for an MCMC sampler. This sampler is called the Gibb's sampler.

Algorithm 3.3.1 (The Gibb's Sampler)**.** Given an initial distribution μ,

(i) Draw $X_0 \sim \mu$. Set $n = 0$.

(ii) Draw s^* uniformly from S.

(iii) Set $Y = X_n$.

(iv) Draw $Y_{s^*} \sim \pi^L_{s^*}(\cdot \,|\, Y_{S\setminus\{s^*\}})$.

(v) Set $X_{n+1} = Y$.

(vi) Set $n = n + 1$ and repeat from step 2.

This version of the Gibb's sampler (where s^* is drawn uniformly from S) is sometimes called the *Glauber dynamics*.

Example 3.3.1 (Simplified Ising model on a square lattice with periodic boundary conditions)**.** Consider the simplified Ising model with $\mathcal{E}(x) = \sum_{(s,t)} x_s x_t$ on $\mathbb{Z}_m^2$ with a neighborhood system given by

$$\mathcal{N}_s = \{(i,j) \in \mathbb{Z}_m^2 : (i - s_1) \bmod m + (j - s_2) \bmod m = 1\}.$$

Note that this neighborhood system produces a square lattice, as before, but the edges are 'glued' together so that the vertices on the far right are connected to those on the far left and the vertices on the top are connected to those on the bottom. A good way to think of this is like pac-man - if a walker on the graph goes too far to the right, it appears on the left. The following code generates realizations of this model using the conditional distribution derived in Example 3.2.6.

Listing 3.3: R code for calculating the energy function in the Ising model

```
N=10^6
T=3
m=500
X=matrix(0, nrow=m, ncol=m)

for(i in 1:N)
{
  i=sample(1:m) ## choose random i
  j=sample(1:m) ## choose random j

  above=X[(i-2)%%m + 1, j]
  below=X[i%%m +1 , j]
  left=X[i, (j-2)%%m +1]
  right=X[i, j%%m + 1]
  prob_X_is_one=( 1 + tanh(1/T * (above+below+left+right)))/2

  U=runif(1, min=0, max=1)
  if(U < prob_X_is_one)
  {
    X[i, j]=1
  }
  else
  {
    X[i,j]=-1
  }
}
my_palette <- colorRampPalette(c("red", "yellow"))(n = 2)
col_breaks = c(-1, 0, 1)
heatmap.2(X,
         #cellnote = X, # same data set for cell labels
         Rowv=FALSE,
         main = "Ising", # heat map title
         #notecol="black",    # change font color of cell labels to black
         density.info="none", # turns off density plot inside color legend
         trace="none",       # turns off trace lines inside the heat map # "none"
         tracecol="black",
         margins =c(12,9),   # widens margins around plot
         col=my_palette,     # use on color palette defined earlier
         breaks=col_breaks,  # enable color transition at specified limits
         symbreaks=TRUE,
         dendrogram="none",   # only draw a row dendrogram
         Colv="NA")
```

Figures 3.3.1 and 3.3.2 show how the temperature parameter changes the output of the Ising model.

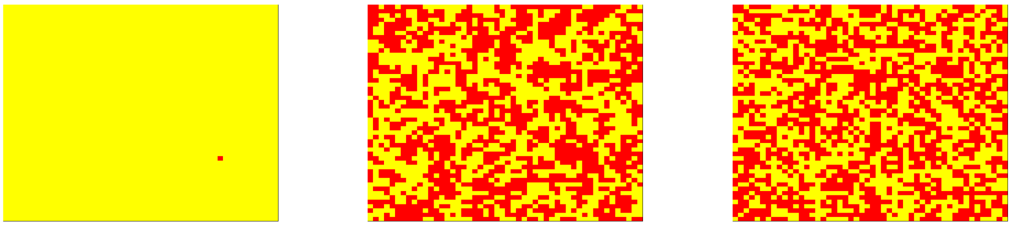

Figure 3.3.1: The Ising model on the 50×50 square lattice simulated from an initial configuration of all ones for 10^6 steps. Left to right: $T = 1$, $T = 5$, $T = 15$.

In order to show that the Gibb's sampler works as desired, we first need to show that it has π as a stationary distribution. This is true if the detailed balance equations are satisfied.

Theorem 3.3.2. The Markov chain, $\{X_n\}_{n\in\mathbb{N}}$, generated by the Gibb's sampler is in detailed balance with π.

Proof. To show this, we need to show that

$$P_{x,y}\pi(x) = P_{y,x}\pi(y),$$

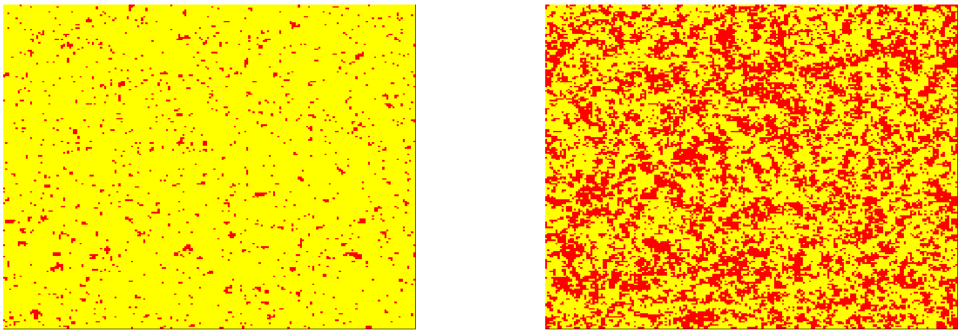

Figure 3.3.2: The Ising model on the 200 × 200 square lattice simulated from an initial configuration of all ones for 10^6 steps. Left: $T = 2$; right: $T = 3$.

for all $x, y \in \Lambda^S$, where $P_{x,y} = \mathbb{P}(X_{n+1} = y \,|\, X_n = x)$. Now, if $\#\{s : y_s \neq x_s\} > 1$, we have $P_{x,y} = P_{y,x} = 0$. So, we can assume there exists a single $s \in S$, labeled s^*, such that $x_{s^*} \neq y_{s^*}$. Now, using the fact that $x_{S\setminus\{s^*\}} = y_{S\setminus\{s^*\}}$,

$$\begin{aligned}
\pi(x)P_{x,y} &= \pi(x)\frac{1}{|S|}\pi^L_{s^*}(y_{s^*} \,|\, x_{S\setminus\{s^*\}}) \\
&= \frac{\pi(x)}{|S|}\frac{\pi(y_{s^*}, x_{S\setminus\{s^*\}})}{\sum_{\lambda\in\Lambda}\pi(\lambda, x_{S\setminus\{s^*\}})} \\
&= \frac{\pi(x)}{|S|}\frac{\pi(y)}{\sum_{\lambda\in\Lambda}\pi(\lambda, x_{S\setminus\{s^*\}})} \\
&= \frac{\pi(y)}{|S|}\frac{\pi(x_{s^*}, y_{S\setminus\{s^*\}})}{\sum_{\lambda\in\Lambda}\pi(\lambda, y_{S\setminus\{s^*\}})} \\
&= \pi(y)\frac{1}{|S|}\pi^L_{s^*}(x_{s^*} \,|\, y_{S\setminus\{s^*\}}) = \pi(y)P_{y,x},
\end{aligned}$$

for all $x, y \in \Lambda^S$ such that $\#\{s : y_s \neq x_s\} = 1$. Thus, the detailed balance equations are satisfied. □

3.4 Simulated Annealing

One very useful application of MCMC techniques is to solve difficult optimization problems. A classic example of such a problem is the traveling salesman problem.

Example 3.4.1 (The Traveling Salesman Problem)**.** Suppose a salesperson has to visit all M cities in a particular area. He or she wishes to do this in the most efficient way possible. The distances between the cities are known, with $D_{i,j}$ being the distance between the ith and jth city. The optimization problem is then to find an order in which to visit the cities, $R = (R_1, R_2, \ldots, R_M)$, that minimizes the total distance traveled in visiting each city. That is, we wish to find

$$\operatorname*{argmin}_{R\subset\{1,\ldots,M\}^M}\left(\sum_{i=1}^{M-1} D_{R_i,R_{i+1}} + D_{R_M,R_1}\right).$$

This problem is known to be very difficult to solve as M grows large (it is a member of a famous class of difficult problems, known as NP hard problems).

The optimization problems we wish to solve have the following general form. Assume we have a function, $H : \Lambda^S \to \mathbb{R}$, where Λ is countable and S is finite. We wish to find the *global minimizer* of this function. That is, we wish to find $x^* \in \Lambda^S$ such that $H(x^*) \leq H(x)$ for all $x \in \Lambda^S$ (this is also denoted $\operatorname{argmin} H(x)$).

3.4.1 Method of Steepest Descent

When it is possible to calculate the gradient of the function H, we can use gradient descent (also called the method of steepest descent) to solve $\operatorname{argmin} H(x)$ computationally. The basic idea is to start with an initial guess, x_0. This guess is then updated by moving a small amount in the direction where $H(x)$ gets smaller fastest (this is $-\nabla H(x_0)$). This process is continued until the gradient gets sufficiently close to 0 (or some more sophisticated stopping condition is satisfied). This approach can be extended to cases (such as those we consider) where a derivative cannot be calculated. In this case, we can define a neighborhood of the current guess and search all (or some) of the points in the neighborhood to look for the direction in which $H(x)$ decreases fastest.

Unfortunately, this method suffers from a very significant limitation: it gets stuck in local minima. A local minima is a point, x^L, such that, for some $\epsilon > 0$, we have $H(x_L) \leq H(x)$ for all x such that $\|x - x_l\| < \epsilon$, where $\| \cdot \|$ is the appropriate norm. You can think of this problem in terms of an analogy. Steepest descent algorithms start at a particular point on the graph of a function and walk ‘downhill’ in the steepest direction. When they reach the bottom of the hill, they stop. If the bottom of the hill is not the location of the global minimizer, they will never reach the global minimizer.

3.4.2 Using Stochastic Methods to Avoid Traps

Simulated annealing is a method that is, in many ways, similar to the steepest descent method. However, instead of calculating a gradient or searching all possible points in the neighborhood of the current point in order to find where to move, it randomly chooses a possible new point. It then moves to this new point if the new value of H is lower. However, it also moves, with a certain probability, to points with higher values of H. In this way, it is able to escape local minima (by traveling through regions where H is increasing).

Simulated annealing achieves this using a sequence of Boltzmann distributions, $\{\pi_T\}_{T\in[0,\infty)}$, where

$$\pi_T(x) = \frac{1}{Z_T}\exp\left\{-\frac{1}{T}H(x)\right\}$$

for all $x \in \Lambda^S$.

Observe that H, which is here acting as an energy function, takes it lowest values at the global minimizers (there could be more than one) of H. Thus, the global minimizers are the most likely configurations if we draw from π_T (so long as $T < \infty$). We have already seen that, as $T \to \infty$, the distributions approach a uniform distribution on Λ^S. It turns out that, as $T \to 0$, the distributions approach a distribution that puts all of its probability mass on the global minimizers of H.

Lemma 3.4.2. Let Λ^S be finite and $H : \Lambda^S \to \mathbb{R}$ have a unique minimizer, x^*. If

$$\pi_T(x) = \frac{1}{Z_T}\exp\left\{-\frac{1}{T}H(x)\right\},$$

then $\lim_{T\to 0}\pi_T(x^*) = 1$.

Proof. We have

$$\pi_T(x^*) = \frac{\exp\{-\frac{1}{T}H(x^*)\}}{\sum_{x\in\Lambda^S}\exp\left\{-\frac{1}{T}H(x)\right\}} = \frac{\exp\{-\frac{1}{T}a\}}{\sum_{x\in\Lambda^S}\exp\left\{-\frac{1}{T}H(x)\right\}},$$

where $a = H(x^*) = \min_{x\in\Lambda^S} H(x)$. We can then write

$$\begin{aligned}\pi_T(x^*) &= \frac{\exp\{-\frac{1}{T}a\}}{\exp\left\{-\frac{1}{T}a\right\} + \sum_{x\in\Lambda^S\setminus\{x^*\}}\exp\left\{-\frac{1}{T}H(x)\right\}},\\ &\geq \frac{\exp\{-\frac{1}{T}a\}}{\exp\left\{-\frac{1}{T}a\right\} + (|\Lambda^S|-1)\exp\left\{-\frac{1}{T}b\right\}},\end{aligned}$$

where $b = \min_{x\in\Lambda^S\setminus\{x^*\}} H(x)$. Dividing both the numerator and denominator by $\exp\{-a/T\}$, we have

$$\pi_T(x^*) = \frac{1}{1 + (|\Lambda^S|-1)\exp\left\{\frac{1}{T}(a-b)\right\}}.$$

Because $a < b$,

$$\exp\left\{\frac{1}{T}(a-b)\right\} \to 0$$

as $T \to 0$. Thus, $\pi_T(x^*) \to 1$ as $T \to 0$. This proof is easily extended to cases where the global minimizer is not unique (you can try this yourself if you want). □

The above results suggest that if it were possible to sample from a Boltzmann distribution with a sufficiently small value of T, then we would draw a global minimizer of H with high probability. In order to sample from such a distribution, we can use MCMC methods. However, it is usually very difficult to find a good starting distribution. Thus, the idea is to start by trying to sample from a Boltzmann distribution with a high temperature. The MCMC algorithm should quickly approach stationarity at this temperature. We then drop the temperature, slowly so that the chain stays close to (or at) stationarity. Continuing on in this manner, we eventually get samples from a distribution that with very high probability returns a global minimizer of H. This gives the simulated annealing algorithm.

Algorithm 3.4.1 (Simulated Annealing)**.** Given a sequence, $\{T_n\}_{n\in\mathbb{N}}$, such that $T_n \in [0,\infty)$ for all $n \in \mathbb{N}$ and $T_1 \geq T_2 \geq \cdots$,

(i) Draw $X_0 \sim \mu$. Set $n = 0$.

(ii) Generate X_n using one step of an MCMC sampler with stationary distribution π_{T_n}.

(iii) If a stopping condition is met (often this is just $n = N$), stop. Otherwise, set $n = n + 1$ and repeat from step 2.

Note that there are a number of things that are not specified by this algorithm: the sequence $\{T_n\}_{n \in \mathbb{N}}$, called the *cooling sequence*; the initial distribution, μ; and the MCMC sampler itself.

Choosing the right cooling sequence is very important in order to ensure that the simulated annealing algorithm converges. The following result gives a criterion for guaranteed convergence.

Theorem 3.4.3. If Λ^S is finite and x^* is the unique global minimizer of H, then if

$$T_n \geq \frac{1}{\log n} |\Lambda^S| \left(\max_{x \in \Lambda^S} H(x) - \min_{x \in \Lambda^S} H(x) \right),$$

we have that $X_n \to x^*$ in probability.

Proof. See [1]. □

Although this cooling schedule gives guaranteed convergence, it is not possible to use it in practice. This is because $1/\log n$ goes to zero far too slowly. In actual applications, people tend to use cooling sequences that go to zero much faster. One of the most commonly used strategies is *geometric cooling.*

Definition 3.4.4 (Geometric cooling)**.** A cooling sequence uses geometric cooling if T_0 is given and

$$T_{n+1} = \beta T_n,$$

for all $n \geq 0$ and some given $\beta \in (0, 1)$.

Note that, using such a cooling sequence, it is no longer certain that the simulated annealing algorithm will converge to the correct answer.

In general, the initial state of the chain is deterministically chosen (so μ is just a distribution that puts probability 1 on this state). If possible, it is best to choose the starting position as close as possible to the global minimizer (if there is any information at all about where it might be).

Choosing the Markov chain is where much of the 'art' of simulated annealing lies. The basic idea is to try and choose a chain that is able to explore the state space quickly but that does not try to jump too far from its current position in any one step (as, when the temperature is cooling, it is important the chain takes lots of small steps). Such a chain works a lot like a steepest descent algorithm, except that it sometimes climbs up hills.

Example 3.4.5 (Example 3.4.1 cont.)**.** Using simulated annealing, we can now write a program that tries to solve the Traveling Salesman Problem. The function we are trying to find the minimizer of is

$$H(R) = \left(\sum_{i=1}^{M-1} D_{R_i, R_{i+1}} + D_{R_M, R_1} \right).$$

We will use geometric cooling and a Metropolis MCMC sampler. The states of the Markov chain will be routes. The proposals will be generated as follows. At the nth step, the chain will have, as its value, the current route $R_{\text{cur}} = (R_1, \ldots, R_M)$. The proposal is generated by drawing two numbers, i and j, uniformly from $\{1, \ldots, M\}$, subject to the constraint $i \neq j$. Assuming without loss of generality that $i < j$, R_{prop} is given by

$$R_{\text{prop}} = (R_1, \ldots, R_{i-1}, R_j, R_{j-1}, \ldots, R_i, R_{j+1}, \ldots, R_M).$$

The acceptance probability is then given by

$$\alpha(R_{\text{cur}}, R_{\text{prop}}) = \frac{\frac{1}{Z_{T_n}} \exp\left\{-\frac{1}{T} H(R_{\text{prop}})\right\}}{\frac{1}{Z_{T_n}} \exp\left\{-\frac{1}{T} H(R_{\text{cur}})\right\}} = \exp\left\{-\frac{1}{T}\left[H(R_{\text{prop}}) - H(R_{\text{cur}})\right]\right\}$$

In the program, the locations of the cities are generated uniformly in a window $[0,1]^2$.

Listing 3.4: R code for a simulated annealing solver of the Traveling Salesman Problem

```
N=10^5
M=15
T=10
beta=0.99

# Generate the positions of the cities
positions= matrix(runif(2*M, min=0, max=1), ncol=2, nrow=M)
 # plot of the city allocations
plot(positions[,1], positions[,2], xlab="x-coordinate", ylab="y-coordinate")

# Calculate the distances between the cities (note it is symmetric)
D=matrix(0, nrow=M, ncol=M)
for(i in 1:M)
{
  for(j in 1:i)
  {
  # norm of the difference between city i and j
    D[i, j]= sum((positions[i,] - positions[j,])^2)
  }
}
 # the last term is zero in this case,
 # but generally you can fill a symmetric matrix like this
D= D + t(D) - diag(D)

# Set initial route to be from 1 to 2 to 3 to ... to M to 1
route= 1:M
# plot of the city allocations
plot(positions[,1], positions[,2], xlab="x-coordinate", ylab="y-coordinate")
for(i in 1:(M-1))
{
  segments(x0=positions[route[i], 1], y0=positions[route[i], 2],
   x1=positions[route[i+1], 1], y1=positions[route[i+1], 2])
}
segments(x0=positions[route[M], 1], y0=positions[route[M], 2],
 x1=positions[route[1], 1], y1=positions[route[1], 2])



# Define cost function (function H, which has to be minimized)
cost_func= function(route)
{
  cost= 0
  for (i in 1:(M-1))
  {
    cost=cost + D[route[i], route[i+1]]
  }
  return(cost)
}

## Calculate initial costs
initial_cost= cost_func(route)
current_cost= cost_func(route)

# The actual algorithm
for (i in 1:N)
{
  # Generate the proposal
  # either by
  #swap_indices=sample(1:M, 2, replace=FALSE)

  # or by
  swap_indices= c(0, 0)
  while(swap_indices[1]==swap_indices[2])
  {
    swap_indices=sort(ceiling(M*runif(2, min=0, max=1)))
  }
  # cut the segment and turn it "upside down"
  new_route= route
  for(i in 0:(swap_indices[2]-swap_indices[1]))
  {
    new_route[swap_indices[1]+i]=route[swap_indices[2]-i]
  }

  # calculate the new costs
  new_cost=cost_func(new_route)

  # calculate the acceptance probability
  alpha = min(exp(-(new_cost- current_cost)/T),1)

  # Decide to accept or not
  rand= runif(1, min=0, max=1)
  if(rand< alpha)
  {
    route= new_route
    current_cost= new_cost
  }

  # cool the temperature T
  T= beta*T
}

plot(positions[,1], positions[,2], xlab="x-coordinate", ylab="y-coordinate")
# plot of the city allocations
for(i in 1:(M-1))
{
  segments(x0=positions[route[i], 1], y0=positions[route[i], 2],
  x1=positions[route[i+1], 1], y1=positions[route[i+1], 2])
}
segments(x0=positions[route[M], 1], y0=positions[route[M], 2],
 x1=positions[route[1], 1], y1=positions[route[1], 2])


initial_cost
current_cost
```

Two sets of cities, with their solutions are depicted in Figure 3.4.1.

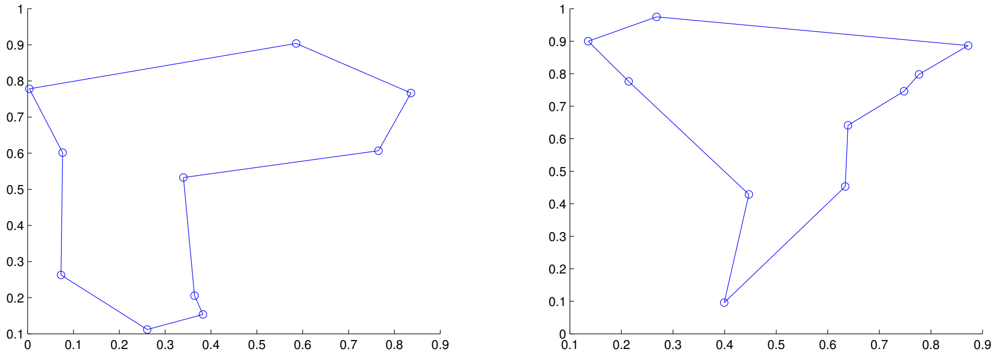

Figure 3.4.1: Two configurations of cities and the accompanying solutions to the traveling salesman problem.

3.4.3 Incorporating constraints

Most real-world optimization problems have some form of constraint. In general, a constrained optimization problem can be written in the form

$$\underset{x \in \Lambda^S}{\operatorname{argmin}} H(x) \text{ such that } x \in \Omega.$$

Here, $\Omega \subset \Lambda^S$ is a set containing the constraints. For example, if $\Lambda^S = \{0, 1, \ldots, M\}^2$, then a possible constraint set could be

$$\Omega = \{x = (x_1, x_2) \in \{0, 1, \ldots, M\}^2 : x_1 > x_2 > 5\}$$

or

$$\Omega = \{x = (x_1, x_2) \in \{0, 1, \ldots, M\}^2 : x_1 + x_2 = 15\}.$$

There are a number of ways to deal with such constraints. One is to rewrite

$$\operatorname*{argmin}_{x \in \Lambda^S} H(x) \text{ such that } x \in \Omega$$

as

$$\operatorname*{argmin}_{x \in \Omega} H(x).$$

This works well if Ω is not a complicated set. However, if Ω is quite complicated — for example, if it is the union of a number of disjoint sets — it may be difficult to find a good MCMC sampler that is irreducible on Ω (or the subset of it where H is non-zero). An alternative, is to use a *penalty function*. That is, we replace $H(x)$ by

$$\widetilde{H}(x) = H(x) + P(x) \cdot \mathbb{I}(x \notin \Omega),$$

where $P(x) > 0$ for all $x \in \Lambda^S \setminus \Omega$. If P is chosen such that $\operatorname{argmin}_{x \in \Omega} H(x) < H(y) + P(y)$ for all $y \notin \Omega$, then a minimizer of $\widetilde{H}$ will be a solution of the constrained optimization problem. Often, P is simply chosen constant. For example, suppose we have the optimization problem

$$\operatorname*{argmin}_{x=(x_1,x_2) \in \{0,1,\ldots,M\}^2} - \left(x_1^2\right)$$

subject to the constraint $x_1 + x_2 = 15$. The unconstrained optimization problem has M distinct solutions, with $x_1^* = M$ and x_2 taking any value in $\{0, 1, \ldots, M\}$. The constrained optimization problem has the solution $x_1^* = 15$ and $x_2 = 0$. If we choose, say, $P(x) = M$ for all $x \in \Lambda^S$, then the solution to

$$\operatorname*{argmin}_{x=(x_1,x_2) \in \{0,1,\ldots,M\}^2} -(x_1^2) + M^2 \cdot \mathbb{I}(x_1 + x_2 \neq 15)$$

is the solution to the constrained problem. It is often a good idea, when choosing P, to make it so that slight deviations from the constraints are not punished too much. In this way, the simulated annealing algorithm is able to travel from local minima to local minima (and, hopefully, eventually to the global minima) through parts of Λ^S that violate the constraints.

Example 3.4.6 (Dense hardcore packings on graphs)**.** Consider a graph $G = (V, E)$, where each vertex is given a color in the set $\{0, 1\}$, subject to the constraint that no two adjacent vertices have color 1. The optimization problem is then to find the maximum number of vertices that can be colored 1 without violating this constraint. For example, consider the graph in Figure 3.4.2.

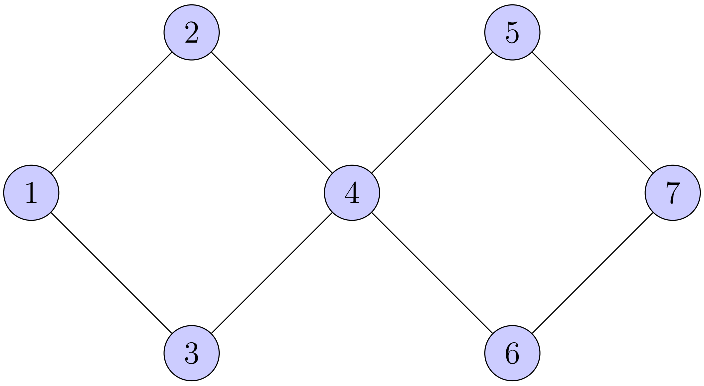

Figure 3.4.2: An example graph.

Note that the optimal solution to this problem is to place 1s on the vertices $2, 3, 5$ and 6. However, as you will see, there are a number of local minima where the simulated annealing algorithm gets stuck — for example, if 1s are placed on the vertices $1, 4$ and 7. One option in approaching the problem is to write the objective function, H, with the constraint built in. For example, we can aim to minimize

$$H(x) = (\text{number of 0s in } x) + 2 \cdot (\text{number of edges where both vertices are 1}),$$

where $x \in \{0,1\}^7$ is a vector assigning colors to each of the vertices. In the code below, we implement simulated annealing with geometric cooling and a Metropolis proposal that chooses a vertex uniformly at random and 'flips' it.

Listing 3.5: R code for a simulated annealing approach to the dense packing problem using a penalty function

```
N= 10^5
beta=0.99
T=10
m=7
E = rbind(c(0, 1, 1, 0,0, 0, 0), c(1, 0, 0, 1, 0, 0, 0),
 c(1, 0, 0, 1, 0, 0, 0), c(0, 1, 1, 0, 1, 1, 0),
          c(0, 0, 0, 1, 0, 0, 1),
          c(0, 0, 0, 1, 0, 0, 1), c(0, 0, 0, 0, 1, 1, 0))
X=rep(0, 7)
current_cost= 7
for(i in 1:N)
{
  index= ceiling(runif(1)*m)
  Y=X
  Y[index]= 1- X[index]
  new_cost=0
  for(i in 1:m)
  {
    if(Y[i]==0)
    {
      new_cost=new_cost+1
  }
  for(j in 1:m)
  {
   if(Y[i]==1 & Y[j]==1&&E[i, j]==1)
   {
    new_cost=new_cost+3
   }
  }
 }
 alpha= exp(-(new_cost - current_cost)/T)
 rand= runif(1, min=0, max=1)
 if(rand< alpha)
 {
  X=Y
  current_cost=new_cost
 }
 T=beta*T
}
X
```

An approach that works better in this case, however, is to incorporate the constraints into the MCMC step. That is, we sample in such a way that adjacent vertices are never both colored 1. We do this using the following Metropolis proposal: we choose a vertex at random, set it to 1 and set all adjacent vertices to 0.

Listing 3.6: R code for a simulated annealing approach to the dense packing problem that only samples in the constrained set

```
### Alternative
N= 10^5
beta=0.99
T=10
m=7
E = rbind(c(0, 1, 1, 0,0, 0, 0), c(1, 0, 0, 1, 0, 0, 0),
c(1, 0, 0, 1, 0, 0, 0), c(0, 1, 1, 0, 1, 1, 0),
     c(0, 0, 0, 1, 0, 0, 1), c(0, 0, 0, 1, 0, 0, 1),
      c(0, 0, 0, 0, 1, 1, 0))
X=rep(0, 7)
current_cost= 7
for(i in 1:N)
{
 index= ceiling(runif(1)*m)
 Y=X
 Y[index]=1
 for(i in 1:m)
 {
  if(E[index, i]==1)
  {
  Y[i]=0
  }
 }
 new_cost=0
 for(i in 1:m)
 {
  new_cost=new_cost + (Y[i]==0)
 }
 alpha= exp(-(new_cost - current_cost)/T)
 rand= runif(1, min=0, max=1)
 if(rand< alpha)
 {
  X=Y
  current_cost=new_cost
 }
 T=beta*T
}
X
```

Chapter 4

Improving the Efficiency of MCMC Samplers

Example 4.0.7 (Estimators behaving badly)**.** Consider the Markov chain illustrated in Figure 4.0.1.

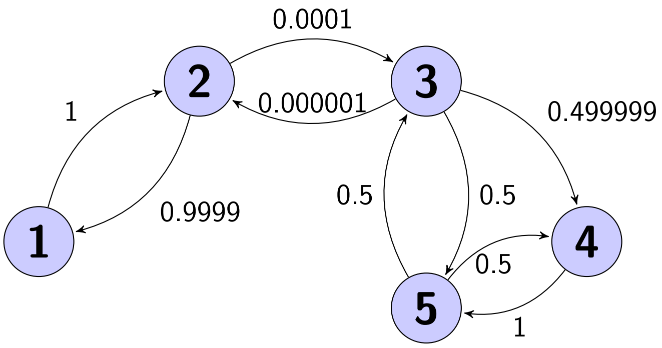

Figure 4.0.1: The transition graph of the Markov chain.

Let $f(\cdot) = \mathbb{I}(\cdot \in \{3,4,5\})$ and suppose that we want to estimate

$$\bar{f} = \mathbb{E}f(X) = \mathbb{P}(X \in \{3,4,5\}),$$

where $X \sim \boldsymbol{\pi}$ with $\boldsymbol{\pi}$ the stationary distribution of the chain. Using the ergodic theorem, we know that

$$\widehat{f} = \frac{1}{n}\sum_{k=0}^{n-1} \mathbb{I}(X_k \in \{3,4,5\})$$

is a consistent estimator of $\bar{f}$. Thus, we can estimate $\bar{f}$ using the following code.

Listing 4.1: Estimating $\bar{f}$

```
n= 10^6
rho=0.1 # for burn-in
X=rep(0, n)
f= rep(0, n)

```

```
P=rbind(c(0,1,0,0,0), c(0.9999,0,0.0001,0,0),
c(0,0.000001,0,0.499999,0.5), c(0,0,0,0,1), c(0,0,0.5,0.5,0))
X0= 3 # starting point of markov chain
X=c() # store values of markov chain
f= c()# store the values of f evaluated in components of X
Z=runif(1, min=0, max=1)
X[1]=min(which(Z<=cumsum(P[X0,]) ))
f[1]= (X[1]==3) + (X[1]==4) + (X[1]==5)
for(i in 1:(n-1))
{
  Z=runif(1, min=0, max=1)
  X[i+1]=min(which(Z<=cumsum(P[X[i],]) ))
  f[i+1]= (X[i+1]==3) + (X[i+1]==4) + (X[i+1]==5)
}
mean(f)
mean(f[(rho*n):n]) # burn in estimator
```

The stationary distribution of the Markov chain is given by

$$\boldsymbol{\pi} \approx (0.0028, 0.0028, 0.2841, 0.1420, 0.5682),$$

so we know $\bar{f} \approx 0.9943$. However, starting the chain with $X_0 = 1$, I got an estimate of $\widehat{f} = 0$ using a sample of size $n = 10^3$. For $n = 10^5$, I got an estimate of $\widehat{f} = 0.72$ and, for $n = 10^6$, I got $\widehat{f} = 0.9758$. In contrast, for $X_0 = 3$, an estimate of $\widehat{f} = 1$ was obtained using a sample size of $n = 10^6$. Note that none of these estimators are even remotely accurate.

The reason why the estimator does not work well is that the Markov chain simply does not move around enough. If it starts in $\{1,2\}$, it takes a long time on average before it reaches $\{3,4,5\}$. If it starts in $\{3,4,5\}$, it takes an even longer time on average before it moves to $\{1,2\}$. As a result, the chain takes too long to sample enough values from both parts of the state space.

Situations where a Markov chain spends too much time sampling from one high probability region of a state space (at the expense of other high probability regions) often occur in MCMC sampling. We will discuss a number of methods for avoiding such situations. It is sometimes the case that we do not know a good starting point for an MCMC algorithm. For example, we may wish to sample from a conditional distribution, where we do not know what typical samples look like (but know how to find a point within the support of the distribution). In this case, we can use what is called a “burn-in” period. That is, we discard / ignore the first part of our sample. For example, if we run an MCMC sampler for $n-1$ steps, we might discard the first $m < n$ steps. Then, our estimator of $\bar{f}$ would be of the form

$$\widehat{f} = \frac{1}{k-n}\sum_{k=m}^{n-1} f(X_k).$$

4.1 Markov chains and mixing times

In order to be sure that an MCMC-based estimator is accurate, the corresponding Markov chain, $\{X_n\}_{n\in\mathbb{N}}$, should have the following two properties:

(i) The distribution of X_n should converge quickly to $\boldsymbol{\pi}$ as $n \to \infty$.

(ii) The chain $\{X_n\}_{n\in\mathbb{N}}$ should rapidly move around the support of $\boldsymbol{\pi}$.

Both of these quantities can be measured using a concept called a *mixing time.* In order to define this, we first define the following two quantities:

$$d(n) = \max_{i\in\mathcal{X}} \|\boldsymbol{\delta}_i P^n - \boldsymbol{\pi}\|_{TV}$$

and

$$\bar{d}(n) = \max_{i,j\in\mathcal{X}} \|\boldsymbol{\delta}_i P^n - \boldsymbol{\delta}_j P^n\|_{TV}.$$

Remember that $\boldsymbol{\delta}_i P^n$ is the distribution of X_n if $X_0 = i$. So, $d(n)$ measures the furthest a Markov chain can be from stationarity at time n. Likewise, $\bar{d}(n)$ measures the biggest possible difference in the distributions of two independent versions of the Markov chain starting in different states.

Lemma 4.1.1. We have $d(n) \leq \bar{d}(n) \leq 2d(n)$.

Proof. 1. $\bar{d}(n) \leq 2d(n)$. This follows by the triangular inequality for the total variation distance. 2. $d(n) \leq \bar{d}(n)$. Let $A \subset \mathcal{X}$ be an arbitrary set and we define

$$\begin{aligned}(\delta_i P^n)(A) &= \sum_{k\in A} (\delta_i P^n)_k \\ \pi(A) &= \sum_{k\in A}\sum_{j\in\mathcal{X}} \pi_j P^n_{j,k} = \sum_{j\in\mathcal{X}} \pi_j (\delta_j P^n)(A)\end{aligned}$$

Now we get for an arbitrary $i \in \mathcal{X}$:

$$\begin{aligned}\|(\delta_i P^n) - \pi\|_{TV} &= \max_{A\subset\mathcal{X}} |\sum_{j\in\mathcal{X}} \pi_j [(\delta_i P^n)(A) - (\delta_j P^n)(A)]| \\ &\leq \sum_{j\in\mathcal{X}} \pi_j \max_{A\subset\mathcal{X}} [(\delta_i P^n)(A) - (\delta_j P^n)(A)] \\ &\leq \max_{j\in\mathcal{X}} \|\delta_i P^n - \delta_j P^n\|\end{aligned}$$

The result follows by taking the maximum over $i \in \mathcal{X}$ on both sides of the inequality. □

Definition 4.1.2 (Mixing time)**.** We define the mixing time of a Markov chain by

$$n_{\mathsf{mix}}(\epsilon) = \min\{n : d(n) \leq \epsilon\}.$$

In order for its mixing time to be small, a Markov chain has to quickly approach stationarity and also move about its state space very rapidly. In other words, Markov chains with small mixing times are ideal for MCMC simulation. In general, it is not possible to calculate mixing times exactly. However, it is sometimes possible to obtain upper and/or lower bounds. The proofs of these inequalities often make use of coupling (which, as we have seen, is a convenient way to obtain bounds on total variation distance).

Example 4.1.3 (Systematic scan Gibb's sampler for random coloring)**.** The systematic scan Gibb's sampler is a modification of the standard Gibb's sampler, where instead of updating the sites of an MRF at random, the sites are updated in order. That is, numbering the sites in S by $\{1, \ldots, K\}$, we perform the following 'scan' in each step:

- Start with $X = (X_1, \ldots, X_K)$.
- Update X_1 conditional on $X_2, \ldots, X_K$.
- Update X_2 conditional on $Y_1, X_3, \ldots, X_K$.
- ...
- Update X_K conditional on $Y_1, \ldots, Y_{K-1}$.
- Return $Y = (Y_1, \ldots, Y_K)$.

Now, consider using a systematic scan Gibb's sampler to sample uniformly from random q-colorings on some graph $G = (V, E)$. Let $k = |V|$ and $\Delta(G)$ be the maximal degree of a vertex in G. Further suppose that $q > 2\Delta(G)^2$. Then,

$$n_{\mathsf{mix}}(\epsilon) \leq k \left(\frac{\log(k) + \log(\epsilon^{-1}) - \log(\Delta(G))}{\log\left(\frac{q}{2\Delta(G)^2}\right)} \right).$$

Here n is the number of individual updates, rather than the number of scans. The proof of this is not too difficult. It uses a coupling between two Markov chains, one started from the stationary distribution and the other from an arbitrary point. It then obtains bounds on the probability the two chains differ at a particular vertex after m scans. This bound is then used to obtain a bound on the chains being different at at least one vertex. Which, in turn gives the mixing time bound in terms of the number of updates. The proof can be found in [2]. Also note that bounds of the same order can be found for less restrictive values of q.

4.1.1 Bounds using eigenvalues

If the transition matrix of a finite state space Markov chain is reversible, it has real-valued eigenvalues (see [4] for a proof), which can be ordered as

$$1 = \lambda_1 > \lambda_2 \geq \cdots \geq \lambda_{|\mathcal{X}|} \geq -1,$$

Note that $\lambda_1 = 1$ is the eigenvalue corresponding to the stationary distribution (in the sense that the associated left eigenvector is the stationary distribution when normalized). Furthermore, if P is aperiodic, it can be shown that $\lambda_{|\mathcal{X}|} \geq -1$.

As it turns out, the eigenvalues of a reversible transition matrix P can be used to describe the speed at which the associated Markov chain reaches stationarity. This provides another avenue for researchers to try to describe the speed with which a Markov chain converges to stationarity.

Definition 4.1.4 (Second largest eigenvalue modulus)**.** The *second largest eigenvalue modulus* (SLEM), λ^*, of a reversible transition matrix P on a finite state space is given by

$$\lambda^* = \max\left\{|\lambda| : \lambda \text{ is an eigenvalue of } P, \lambda \neq 1\right\}.$$

Definition 4.1.5 (Absolute spectral gap)**.** The *absolute spectral gap* of a reversible Markov chain on a finite state space is given by

$$\gamma^* = 1 - \lambda^*.$$

Using the eigenvalue approach, we consider *relaxation times* instead of mixing times.

Definition 4.1.6 (Relaxation time)**.** The relaxation time of a reversible Markov chain on a finite state space is given by

$$n_{\mathsf{rel}} = \frac{1}{\gamma^*}.$$

The smaller the relaxation time of a Markov chain, the faster it mixes. The following theorem shows that relaxation times can be used to bound mixing times

Theorem 4.1.7. Let P an be irreducible, reversible Markov chain on a finite state space with stationary distribution $\boldsymbol{\pi}$. Let $(\boldsymbol{\pi})_{\mathsf{min}} = \min_{i\in\mathcal{X}}(\boldsymbol{\pi})_i$. Then,

$$n_{\mathsf{mix}}(\epsilon) \leq \log\left(\frac{1}{\epsilon(\boldsymbol{\pi})_{\mathsf{min}}}\right) n_{\mathsf{rel}}.$$

If additionally, the Markov chain is aperiodic, then

$$n_{\mathsf{mix}}(\epsilon) \geq (n_{\mathsf{rel}} - 1)\log\left(\frac{1}{2\epsilon}\right)$$

Proof. See [4]. Theorem 12.3 and 12.4. □

It also possible to get lower bounds on mixing times using relaxation times. See [4] for more details.

4.2 Speeding things up

In general, Markov chains will mix faster if they explore the state space quickly. However, most of the algorithms we have considered so far only make *local* changes at each step (that is, the samples do not change too much from step to step). For example, the algorithms for the Ising model only change the value at one site in each step. It would clearly be better if we could make *global* changes at each step (that is, totally change the value of the Markov chain from one step to the next). The problem, however, is that global changes are very difficult to make. We certainly cannot just change everything at random and try to accept it using Metropolis-Hastings (this would be almost as bad as the acceptance rejection method, which we have seen is very bad). Much more sophisticated techniques must be used. We will focus on two: *parallel tempering* and the *Swendsen-Wang* algorithm.

4.2.1 Parallel tempering

A major challenge when designing MCMC algorithms is that many distributions we are interested in sampling from have regions of high probability surrounded by regions of low probability. For example, in the Ising model the most likely configurations are all 1s and all -1s. Configurations that are a mixture of 1s and -1s have much lower probabilities.

In general, it is hard to make samplers that can escape from high probability regions, travel quickly through low energy regions, and find other high energy regions. The idea of parallel tempering is to use multiple Markov chains to accomplish this.

We have already seen, when considering simulated annealing, that the temperature of a Boltzmann distribution

$$\pi_T(x) = \frac{1}{Z_T} \exp\left\{-\frac{1}{T}\mathcal{E}(x)\right\}$$

can be used to control the shape of the distribution. In particular, as $T \to \infty$, π_T converges to a uniform distribution and, as $T \to 0$, π_T becomes concentrated at the configurations with the lowest energy.

In parallel tempering, we suppose we want to sample from some distribution of the form

$$\pi_T(x) = \frac{1}{Z_T} \exp\left\{-\frac{1}{T}\mathcal{E}(x)\right\}.$$

With a little bit of work, almost any distribution can be written in this form. The basic idea is then to run multiple Markov chains, each at a different temperature, with the Markov chain with the lowest temperature having stationary distribution π_T. The Markov chains running at high temperatures will quickly explore the state space, while the Markov chains at lower temperatures will mainly move in small regions of high probability. Periodically, we will switch the values of two of the Markov chains using a Metropolis move (so that the stationary distributions of the Markov chains are preserved). In this way, the Markov chains with lower temperatures can 'teleport' from one region of high probability to another.

More formally, the setup is as follows. We consider a sequence of M temperatures

$$T = T_1 < T_2 < \cdots < T_M$$

and define associated Boltzmann distributions $\pi_{T_1}, \ldots, \pi_{T_M}$. We then define a Markov chain, $\{(X_n^{(1)}, \ldots, X_n^{(M)})\}_{n \in \mathbb{N}}$ on $\mathcal{X} \times \cdots \times \mathcal{X}$ with stationary distribution $\pi(x_1, \ldots, x_M) = \pi_{T_1}(x_1) \cdots \pi_{T_M}(x_M)$. Note that the projection of the first coordinate of this Markov chain is a Markov chain, $\{X_n^{(1)}\}_{n \in \mathbb{N}}$, with stationary distribution π_T. We can simulate the Markov chain $\{(X_n^{(1)}, \ldots, X_n^{(M)})\}_{n \in \mathbb{N}}$ by running separate MCMC samplers for each of its components.

Now, we wish to add the ability for the Markov chains running at different temperatures to switch places. We decide whether or not to switch two Markov chains using a Metropolis acceptance probability. This preserves the stationary distribution of $\{(X_n^{(1)}, \ldots, X_n^{(M)})\}_{n \in \mathbb{N}}$. Suppose that, at time n, we wish to switch the values of the chains $\{X_n^{(i)}\}_{n\in\mathbb{N}}$ and $\{X_n^{(i+1)}\}_{n\in\mathbb{N}}$. This is the same as a Metropolis move from the state

$$X = (X_n^{(1)}, \ldots, X_n^{(i-1)}, X_n^{(i)}, X_n^{(i+1)}, X_n^{(i+2)}, \ldots, X_n^{(M)})$$

to the state

$$Y = (X_n^{(1)}, \ldots, X_n^{(i-1)}, X_n^{(i+1)}, X_n^{(i)}, X_n^{(i+2)}, \ldots, X_n^{(M)}).$$

The probability of accepting such a move when using a Metropolis MCMC sampler is

$$\begin{aligned}
&\alpha(X,Y) = \frac{\pi(Y)}{\pi(X)} \\
&= \frac{\pi_{T_1}(X_n^{(1)}) \cdots \pi_{T_{i-1}}(X_n^{(i-1)}) \cdot \pi_{T_i}(X_n^{(i+1)}) \cdot \pi_{T_{i+1}}(X_n^{(i)}) \cdot \pi_{T_{i+1}}(X_n^{(i+2)}) \cdots , \pi_{T_M}(X_n^{(M)})}{\pi_{T_1}(X_n^{(1)}) \cdots \pi_{T_{i-1}}(X_n^{(i-1)}) \cdot \pi_{T_i}(X_n^{(i)}) \cdot \pi_{T_{i+1}}(X_n^{(i+1)}) \cdot \pi_{T_{i+1}}(X_n^{(i+2)}) \cdots , \pi_{T_M}(X_n^{(M)})} \\
&= \frac{\pi_{T_i}(X_n^{(i+1)}) \cdot \pi_{T_{i+1}}(X_n^{(i)})}{\pi_{T_i}(X_n^{(i)}) \cdot \pi_{T_{i+1}}(X_n^{(i+1)})}.
\end{aligned}$$

Putting everything together, we have the following algorithm.

Algorithm 4.2.1 (Parallel tempering)**.** Given a temperature sequence $T_1 < T_2 < \cdots < T_M$ and $\beta \in (0,1)$,

(i) Draw $X_0^{(1)} \sim \mu_0, \ldots, X_0^{(M)} \sim \mu_M$. Set $n = 0$.

(ii) With probability β attempt to switch two chains: set $X_{n+1}^{(1)} = X_n^{(1)}, \ldots, X_{n+1}^{(M)} = X_n^{(M)}$, then choose i uniformly from $\{1, \ldots, M-1\}$ and, with probability

$$\alpha = \frac{\pi_{T_i}(X^{(i+1)}) \cdot \pi_{T_{i+1}}(X^{(i)})}{\pi_{T_i}(X^{(i)}) \cdot \pi_{T_{i+1}}(X^{(i+1)})},$$

set $X_{n+1}^{(i+1)} = X_n^{(i)}$ and $X_{n+1}^{(i)} = X_n^{(i+1)}$.

(iii) Otherwise, generate $X_{n+1}^{(1)}, \ldots, X_{n+1}^{(M)}$ using MCMC samplers with stationary distributions $\pi_{T_1}, \ldots, \pi_{T_M}$.

(iv) Set $n = n + 1$ and repeat from step 2.

In order for the parallel tempering algorithm to work well, the temperatures $T_1, \ldots, T_M$ need to be chosen carefully. Basically, they should be close enough that the probability of the Markov chains switching is not too low. However, they should also be chosen far enough apart that they behave differently from one another.

Example 4.2.1 (Ising model on 2D square lattice)**.** Consider the version of the Ising model on an $m \times m$ lattice considered in Section 3.2.2. In the following code, we implement a parallel tempering algorithm for simulating from this using three chains. To test the effectiveness of the approach, we estimate the average spin per site. That is, we estimate

$$\bar{f} = \frac{1}{m^2} \sum_{s \in S} X_s.$$

Because, for every configuration, there is an equally likely configuration with the opposite spins, this value should be 0. However, as you will notice, when the MCMC sample is not mixing properly, estimates of this quantity will be far from 0.

Listing 4.2: Parallel tempering for the Ising model

```
ising_energy=function(J, H, X)
{
  m= ncol(X)

  single_energies=0
  pair_energies= 0
  for (i in 1:m)
  {
    for(j in 1:m)
    {
      single_energies=single_energies - H*X[i,j]
    }
  }
  for(i in 1:m)
  {
    for(j in 1:m)
    {
      if(i<m & j < m)
      {
        # pair to the right vertex
        pair_energies= pair_energies - J*X[i, j]*X[i, j+1]
        # pair to the below vertex
        pair_energies= pair_energies - J*X[i,j]*X[i+1, j]
      }
      else if (i==m & j < m)
      {
        # pair to the right vertex
        pair_energies= pair_energies - J*X[i, j]*X[i, j+1]
      }

      else if (i<m & j==m)
      {
        # pair to the below vertex
        pair_energies= pair_energies - J*X[i,j]*X[i+1, j]
      }
      else
      {
        pair_energies=pair_energies
      }
    }
  }
  energy= single_energies + pair_energies
  return(energy)
}

N=10^2
J0=1
H0=0
m=20
beta= 0.2
results= c()
change=rep(0, 2)
T0= c(1.8, 2.1, 2.35)

###step 1
X0= array(1, c(m, m, 3))
for(i in 1:N)
{
  U= runif(1)
  if(U< beta) ### step 2
  {
    j = sample(1:(3-1), size=1, replace=TRUE) # step 2(b)
    eps1 =ising_energy(H=H0, J=J0, X=X0[,,j]) # step 2(c)
    eps2= ising_energy(H=H0, J=J0, X=X0[,,(j+1)])
    alpha= exp(-(1/T0[j]+ 1/T0[j+1])*(eps2-eps1))
    if(runif(1) < alpha)
    {
      Y=X0[,,j]
      X0[,,j]=X0[,,(j+1)]
      X0[,,(j+1)]=Y
      change[j]=change[j]+1
    }
  }
  else
  {
    for(r in 1:3)  ### step (3)
    {
      # choose randomly a coordinate (i,j)
      k= sample(1:m, size=1, replace=TRUE)
      l= sample(1:m, size=1, replace=TRUE)

      Y=X0[,,r]
      Y[k, l]= -1 *Y[k,l] # flip this energy

      alpha= exp(1/T0[r]*(ising_energy(J=J0, H=H0, X=X0[,,r]) - ising_energy(J=J0, H=H0,
      Z= runif(1)

      if(Z< alpha){X0[,,r]=Y}
    }
  }
  results[i]= mean(X0[,,1])
}
```

Chapter 5

Markov Chain Monte Carlo on General State Spaces

5.1 Markov Chains on General State Spaces

So far we have just considered Markov chains defined on a countable state space. It seems natural to ask if we can also think about Markov chains on a state space, $\mathcal{X}$, that is not necessarily countable. Such a state space is called a *general state space.*

In order to define a Markov chain on such a state space, we need an analogue of a transition matrix. Because we are no longer guaranteed that the state space is countable, we cannot use a matrix for this purpose. Instead, we use a more abstract object called a *transition kernel.*

Definition 5.1.1 (Transition Kernel)**.** Given a general state space, $\mathcal{X}$, a transition kernel is a function, K, on $\mathcal{X} \times \mathcal{B}(\mathcal{X})$ such that

(i) $K(x, \cdot)$ is a probability measure for all $x \in \mathcal{X}$.

(ii) $K(\cdot, A)$ is measurable for all $A \in \mathcal{B}(\mathcal{X})$.

We then define time-homogeneous Markov chains on a general state space using the notion of a transition kernel.

Definition 5.1.2 (General state space time-homogeneous Markov chain)**.** A sequence $\{X_n\}_{n \in \mathbb{N}}$ is a time-homogeneous general state space Markov chain with transition kernel K if

$$\begin{aligned} &\mathbb{P}\left(X_{n+1} \in A \,|\, X_0 = x_0, \ldots, X_n = x_n\right) \\ &= \mathbb{P}\left(X_{n+1} \in A \,|\, X_n = x_n\right) \\ &= \int_A K(x_n, \mathrm{d}x). \end{aligned}$$

Example 5.1.3 (A random walk with normal increments)**.** Consider the following general state space Markov chain. We set $X_0 = 0$ and

$$X_{n+1} = X_n + \epsilon_n,$$

for $n \geq 0$, where the $\{\epsilon_n\}_{n \in \mathbb{N}}$ are iid normal random variables with mean 0 and variance σ^2. The kernel of this chain is given by

$$K(x_n, A) = \int_A \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(y - x_n)^2}{2\sigma^2}\right\} \mathrm{d}y.$$

Note that, here,

$$K(x_n, \mathrm{d}y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(y - x_n)^2}{2\sigma^2}\right\}.$$

Concepts such as irreducibility and recurrence are trickier to define for general state space Markov chains (this is basically because continuous probability distributions assign zero probability to each point in the state space, so it no longer makes sense to talk about the probability of returning to a specific point). If you are interested, have a look at [7] or [5].

What is important for us, however, is the concept of an invariant measure.

Definition 5.1.4 (Invariant measure)**.** A σ-finite measure, μ, is invariant for the transition kernel K if

$$\mu(B) = \int_{\mathcal{X}} K(x, B)\mu(\mathrm{d}x).$$

As in the countable state space case, the detailed balance conditions will play a key role in establishing that MCMC methods work.

Definition 5.1.5 (Detailed balance)**.** A Markov chain with transition kernel K is in detailed balance with a measure μ if

$$K(x, \mathrm{d}y)\mu(\mathrm{d}x) = K(y, \mathrm{d}x)\mu(\mathrm{d}y)$$

for every $(x, y) \in \mathcal{X}^2$.

Theorem 5.1.6. *If K satisfies the detailed balance equations with a probability measure π, then π is an invariant distribution of the chain.*

Proof. We have

$$\begin{aligned}\int_{\mathcal{X}} K(x, B)\pi(\mathrm{d}x) &= \int_{\mathcal{X}} \int_B K(x, \mathrm{d}y)\pi(\mathrm{d}x) \\ &= \int_B \int_{\mathcal{X}} K(y, \mathrm{d}x)\pi(\mathrm{d}y) = \int_B \pi(\mathrm{d}y) = \pi(B).\end{aligned}$$

□

Note that, as far as we are concerned, $\pi(\mathrm{d}y)$ will always be a probability density.

5.2 Metropolis-Hastings Algorithm

Equipped with a version of the detailed balance conditions, we are able to extend the Metropolis-Hastings algorithm to general state spaces.

Algorithm 5.2.1 (Metropolis-Hastings)**.** Suppose we wish to sample from a density f (or something proportional to a density) using a proposal density $q(y \,|\, x)$ (a density that is conditional on x). We can do this as follows:

(i) Draw $X_0 \sim \mu$. Set $n = 0$.

(ii) Draw $\sim q(y \,|\, X_n)$.

(iii) Set $\alpha(X_n, Y) = \min\left\{1, \frac{f(Y)}{f(X_n)} \frac{q(X_n \,|\, Y)}{q(Y \,|\, X_n)}\right\}$.

(iv) With probability $\alpha(X_n, Y)$ set $X_{n+1} = Y$. Otherwise, set $X_{n+1} = X_n$.

(v) Set $n = n + 1$ and repeat from step 2.

Example 5.2.1. We consider a random vector $B = (B_1, \dots, B_n)^T$ with $B_1 \sim \mathsf{Bin}(1, \theta)$, $\theta \in (0, 1)$ unknown, and $B_1, \dots, B_n$ are i.i.d. Consequently, the sum $S_n = \sum_{i=1}^n B_i$ holds: $S_n \sim \mathsf{Bin}(1, \theta)$. Moreover we assume that we have a prior for θ that is

$$\pi(\theta) = 2\cos^2(4\pi\theta).$$

We now want to sample from the posterior density $f(\theta) = \pi(\theta || S_n)$ by using the proposal density

$$q(\theta' | \theta) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\tfrac{(\theta' - \theta)^2}{2\sigma^2}).$$

We know that $f(\theta)$ is of the following

$$f(\theta) = C\theta^{S_n}(1 - \theta)^{n - S_n} \cos^2(4\pi\theta)$$

where C is an unknown, but finite constant. The acceptance probability is given by:

$$\alpha(x, y) = \min\{1, \left(\frac{y}{x}\right)^{S_n} \left(\frac{1 - y}{1 - x}\right)^{n - S_n} \left(\frac{\cos(4\pi Y)}{\cos(4\pi Y)}\right)^2\}$$

Listing 5.1: Independence Sampler Code

```
prior_theta= function(theta){2*cos(4*pi*theta)^2}
curve(prior_theta, xlim=c(0, 1))
integrate(prior_theta, lower =0, upper=1)

n= 10
Sn= 5
posterior_theta= function(theta){theta^Sn *(1-theta)^(n-Sn) *prior_theta(theta)}
postint= integrate(posterior_theta, lower =0, upper=1)$value
posterior2_theta= function(theta){1/postint* posterior_theta(theta)}
curve(posterior2_theta)
curve(1/(sqrt(2*pi*0.01))*exp(-(x-0.5)^2/0.2), add=TRUE, col="red")

N=10^3
X=c()
X[1]= runif(1, 0, 1)# step 1
for(i in 1:N)
```

```
{
  Y= rnorm(1, mean= X[i], sd= 0.1) # step 2
  alpha_i= min(1, posterior_theta(Y)/posterior_theta(X[i])) # step 3
  U = runif(1, 0, 1) # step 4
  if(U < alpha_i){X[i+1]= Y}
  else{X[i+1]= X[i]}
}

hist(X, freq=FALSE)
curve(posterior2_theta, add=TRUE)
```

Theorem 5.2.2. Let $\{X_n\}_{n\in\mathbb{N}}$ be the Markov chain produced by the Metropolis-Hastings algorithm. Then, $\{X_n\}_{n\in\mathbb{N}}$ is in detailed balance with f.

Proof. The transition kernel of $\{X_n\}_{n\in\mathbb{N}}$ can be written as

$$K(x, \mathrm{d}y) = \alpha(x, y)q(y \,|\, x) + (1 - \alpha^*(x))\delta_x(y),$$

where $\delta_x(y)$ is the Dirac delta function. Here, the term $(1 - \alpha^*(x))$ gives the probability that the chain rejects the proposed state and stays where it is. This is given by

$$1 - \alpha^*(x) = 1 - \int_{\mathcal{X}} \alpha(x, y)q(y \,|\, x)\mathrm{d}y.$$

Thus, showing detailed balance amounts to showing that

$$K(x, \mathrm{d}y)f(x) = K(y, \mathrm{d}x)f(y).$$

That is

$$\alpha(x, y)q(y \,|\, x)f(x)+(1-\alpha^*(x))\delta_x(y)f(x) = \alpha(y, x)q(x \,|\, y)f(y)+(1-\alpha^*(y))\delta_y(x)f(y)$$

Now, it is immediate that $(1-\alpha^*(x))\delta_x(y)f(x) = (1-\alpha^*(y))\delta_y(x)f(y)$, as $\delta_x(y) \neq 0 \Rightarrow x = y$. Thus, we simply need to show

$$\alpha(x, y)q(y \,|\, x)f(x) = \alpha(y, x)q(x \,|\, y)f(y).$$

This is trivially satisfied if $f(x) = 0$ or $f(y) = 0$, so we can ignore these cases. In order to show the identify, we take cases. First, we consider the case where $f(y)q(x \,|\, y) > f(x)q(y \,|\, x)$. Then we have

$$\begin{aligned}\alpha(x, y)q(y \,|\, x)f(x) &= \min\left\{\frac{f(y)}{f(x)}\frac{q(x \,|\, y)}{q(y \,|\, x)}, 1\right\} q(y \,|\, x)f(x) = q(y \,|\, x)f(x)\\ &= \frac{q(y \,|\, x)f(x)}{q(x \,|\, y)f(y)}q(x \,|\, y)f(y) = \min\left\{\frac{f(x)}{f(y)}\frac{q(y \,|\, x)}{q(x \,|\, y)}, 1\right\} q(x \,|\, y)f(y)\\ &= \alpha(y, x)q(x \,|\, y)f(y).\end{aligned}$$

In the second case, $f(x)q(y \,|\, x) \geq f(y)q(x \,|\, y)$. Then,

$$\alpha(x, y)q(y \,|\, x)f(x) = \min\left\{\frac{f(y)}{f(x)}\frac{q(x \,|\, y)}{q(y \,|\, x)}, 1\right\} q(y \,|\, x)f(x) = \frac{f(y)}{f(x)}\frac{q(x \,|\, y)}{q(y \,|\, x)}q(y \,|\, x)f(x)$$

$$= f(y)q(x \,|\, y)f(y) = \min\left\{\frac{f(x)}{f(y)}\frac{q(y \,|\, x)}{q(x \,|\, y)}, 1\right\} q(x \,|\, y)f(y) = \alpha(y, x)q(x \,|\, y)f(y).$$

□

There are a number of different possibilities when choosing the probability density. Two of the most popular are the *independence sampler* and the *random walk sampler*.

5.2.1 The Independence Sampler

In the independence sampler, the proposal density does not depend on the current location of the Markov chain. That is, $q(y \,|\, x) = g(y)$, where $g(y)$ is some density. The acceptance probability is then of the form

$$\alpha(x,y) = \frac{f(y)}{f(x)} \frac{g(x)}{g(y)}.$$

The independence sampler is quite similar to the acceptance-rejection algorithm. However, it actually works better in the sense that the probability of accepting a proposal is always bigger than or equal to the acceptance probability in the acceptance-rejection algorithm. Note that the name is a bit deceptive, as the samples produced by the independence sampler are not independent of on another (it is the proposal that is independent of the current state).

Example 5.2.3 (Sampling from a truncated normal distribution)**.** Suppose we wish to sample from the density

$$f(x) = \frac{\sqrt{2}}{\sqrt{\pi}} \exp\left\{-\frac{1}{2}x^2\right\} \mathbb{I}(x > 0),$$

which is the density of a normal distribution truncated to the interval $[0, \infty)$. We use the independence sampler with the proposal $g(x) = \exp\{-x\}$, which is the density of an exponential distribution with rate 1. The acceptance probability here is of the form

$$\alpha(x,y) = \frac{\frac{\sqrt{2}}{\sqrt{\pi}} \exp\left\{-\frac{1}{2}y^2\right\} \mathbb{I}(x > 0)}{\frac{\sqrt{2}}{\sqrt{\pi}} \exp\left\{-\frac{1}{2}x^2\right\}} \frac{\exp\{-x\}}{\exp\{-y\}} = \frac{\exp\{y - y^2/2\}}{\exp\{x - x^2/2\}}.$$

Listing 5.2: Independence Sampler Code

```r
N= 10^3
X= c()
X[1]= 1
for(i in 1:N)
{
  Y= rexp(1)
  alpha_i= min(1, exp(Y-Y^2/2)/exp(X[i]- X[i]^2/2))
  U=runif(1, 0, 1)
  if(U < alpha_i)
  {
    X[i+1]=Y
  }
  else X[i+1]=X[i]
}
hist(X[100:length(X)], freq=FALSE)
curve(sqrt(2/pi)*exp(-1/2*x^2), add=TRUE, col="red")
```

5.2.2 The Random Walk Sampler

A more interesting Metropolis-Hastings sampler is to let the proposal try to make steps like a random walk (with continuously distributed step sizes). In this version, we choose $q(y \,|\, x) = g(\|x - y\|)$. That is, we choose the proposal density to be a symmetric distribution centered at x. The classical example is to use a normal distribution with mean x and variance σ^2. In this case, the acceptance probability is of the form

$$\alpha(x, y) = \frac{f(y)}{f(x)}.$$

Example 5.2.4 (Sampling from a truncated normal distribution)**.** Suppose, againm, we wish to sample from the density

$$f(x) = \frac{\sqrt{2}}{\sqrt{\pi}} \exp\left\{-\frac{1}{2}x^2\right\} \mathbb{I}(x > 0).$$

We use the random walk sampler with the proposal $q(y|x) = \varphi(y; x, \sigma^2)$, where $\varphi(\cdot; \mu, \sigma^2)$ is the density of a normal distribution with mean μ and variance σ^2. The acceptance probability here is of the form

$$\alpha(x, y) = \frac{\frac{\sqrt{2}}{\sqrt{\pi}} \exp\left\{-\frac{1}{2}y^2\right\}}{\frac{\sqrt{2}}{\sqrt{\pi}} \exp\left\{-\frac{1}{2}x^2\right\}} = \frac{\exp\{-y^2/2\}}{\exp\{-x^2/2\}}.$$

Listing 5.3: Random Walk Sampler Code

```r
N=10^3
X=c()
X[1]= 1
alpha=c()
sd0= 1
for(i in 1:N)
{
  Y=X[i]+ rnorm(1, 0,sd=sd0)
  alpha[i]= min(exp(-Y^2/2 + X[i]^2/2)*(Y>0), 1)
  U=runif(1, 0, 1)
  if(U < alpha[i])
  {
    X[i+1]=Y
  }
  else X[i+1]=X[i]
}
mean(alpha)
hist(X, freq=FALSE)
curve(sqrt(2/pi)*exp(-1/2*x^2), add=TRUE, col="red")
```

Example 5.2.5 (Multi-dimensional Metropolis-Hastings)**.** Suppose we want to sample from the density

$$f(\mathbf{x}) = 0.7 \cdot \varphi(\mathbf{x}; \boldsymbol{\mu}_1, \Sigma_1) + 0.3 \cdot \varphi(\mathbf{x}; \boldsymbol{\mu}_2, \Sigma_2),$$

where $\varphi(\cdot;\boldsymbol{\mu},\Sigma)$ is the density of a multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ and

$$\boldsymbol{\mu}_1=\begin{pmatrix}4\\5\end{pmatrix},\quad \Sigma_1=\begin{bmatrix}1.0&0.7\\0.7&1.0\end{bmatrix},\quad \boldsymbol{\mu}_2=\begin{pmatrix}0.7\\3.5\end{pmatrix},\quad \Sigma_1=\begin{bmatrix}1.0&-0.7\\-0.7&1.0\end{bmatrix}.$$

The density is show in Figure 5.2.1. We can sample from this density using random

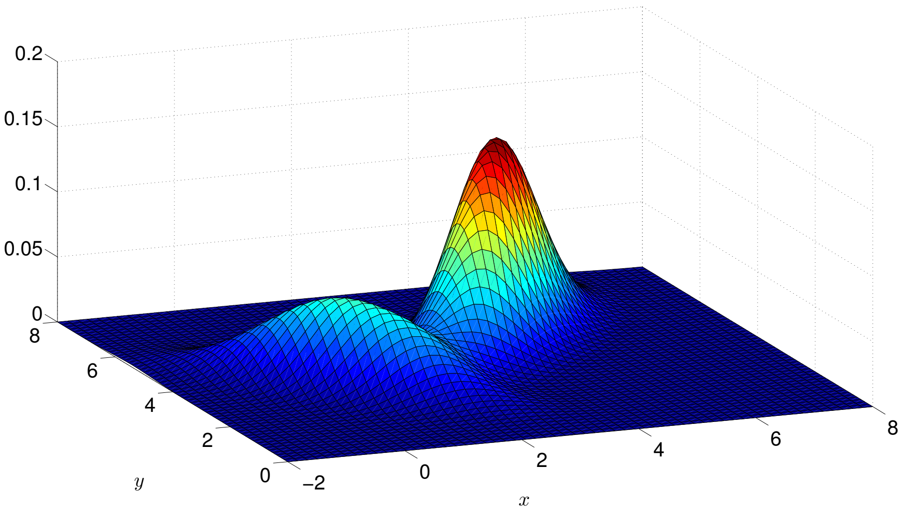

Figure 5.2.1: Plot of $f(x)$.

walk Metropolis-Hastings with a multivariate normal proposal density. That is, we set $Y=X_n+\epsilon$, where

$$\epsilon\sim\mathcal{N}\left(\begin{pmatrix}0\\0\end{pmatrix},\quad \sigma\cdot\begin{pmatrix}1&0\\0&1\end{pmatrix}\right)$$

Figure ?? shows a plot of the sample obtained from the algorithm superimposed on a contour plot of the density.

Listing 5.4: Multivariate Metropolis-Hastings Code

```
dmixed_norm= function(x, mu1, mu2, Sigma1, Sigma2, lambda)
{
  det1 = det(Sigma1)
  det2= det(Sigma2)
  temp1 = t(x-mu1) %*%solve(Sigma1)%*%(x-mu1)/2
  temp2 = t(x-mu2)%*%solve(Sigma2)%*%(x-mu2)/2
  result = lambda* 1/(2*pi)^(nrow(Sigma1)/2) * 1/det1 * exp(-temp1)
  + (1-lambda)* 1/(2*pi)^(nrow(Sigma2)/2) * 1/det2 * exp(-temp2)
  return(result)
}

## Parameters
mu01 = c(4, 5)
Sigma01 = cbind(c(1, 0.7), c(0.7, 1))
```

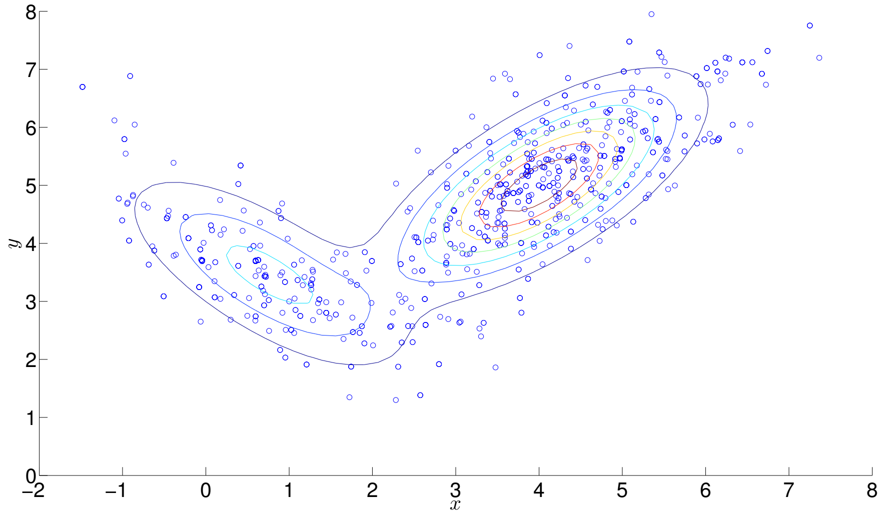

Figure 5.2.2: Plot of a sample from the MH sampler superimposed on a contour plot of the density

```
mu02 = c(0.7, 3.5)
Sigma02 = cbind(c(1, -0.7), c(-0.7, 1))
lambda0 = 0.7

### plot of density function
xseq= seq(-2, 8, length= 50)
yseq= seq(-2, 8, length= 50)
zseq= matrix(0, nrow=50, ncol=50)
for(k in 1:50)
{
  for(l in 1:50)
  {
    zseq[k,l]= dmixed_norm(x=c(xseq[k], yseq[l]), mu1=mu01,
                mu2=mu02, Sigma1=Sigma01, Sigma2=Sigma02, lambda=lambda0)

  }
}
persp(xseq, yseq, zseq, zlab=" ",
      ticktype="detailed", theta= 315, phi= 0)
contour(xseq, yseq, zseq, lty=1)

### Algorithm
N= 10^3
X= matrix(0, ncol= 2, nrow=N+1)

X[1,]= c(1, 3)
for(i in 1:N)
{
  Y= X[i,]+ rnorm(2, mean=0, sd=1)
  f_Y = dmixed_norm(x=Y, mu1=mu01, mu2=mu02,
          Sigma1=Sigma01, Sigma2=Sigma02, lambda=lambda0)
  f_X=dmixed_norm(x=X[i,], mu1=mu01, mu2=mu02,
         Sigma1=Sigma01, Sigma2=Sigma02, lambda=lambda0)
  alpha[i]= min(1, f_Y/f_X)
  U=runif(1, 0, 1)
  if(U < alpha[i])
  {
    X[i+1,]=Y
  }
  else X[i+1,]=X[i,]
}
contour(xseq, yseq, zseq, lty=1)
points(X[,1], X[,2])
```

5.3 The Gibb's Sampler

We can also consider a version of the Gibb's sampler for general state spaces. Suppose we wish to sample from the multivariate density $f(x_1,\dots,x_d)$ and we know the conditional densities

$$f(x_1 \,|\, x_2,\dots,x_d), f(x_2 \,|\, x_1,x_3,\dots,x_d),\dots,f(x_d \,|\, x_1,\dots,x_{d-1}).$$

We can use the following (systematic-scan) algorithm.

Algorithm 5.3.1 (Continuous Gibb's Sampler (Systematic Sampling Version))**.** One step of the Gibb's sampler (from $\mathbf{X}$ to $\mathbf{Y}$) is given by

1. Sample $Y_1 \sim f(x_1 \,|\, X_2,\dots,X_d)$.

2. Sample $Y_2 \sim f(x_2 \,|\, Y_1,X_3,\dots,X_d)$.

$\vdots$

d. Sample $Y_d \sim f(x_d \,|\, Y_1,\dots,Y_{d-1})$.

The trick in using the Gibb's sampler is to find the conditional densities. A simple trick for this, which is used a lot in Bayesian statistics, is to observe that (so long as $f(y) > 0$)

$$f(x \,|\, y) = \frac{f(x,y)}{f(y)} \propto f(x,y),$$

so, in some sense, the joint density gives us all information about the conditional densities except their normalizing constants. Thus, if one looks from the right perspective (this takes a little practice) one can often get the conditional density of x given y by looking at $f(x,y)$ and treating y as a constant.

Example 5.3.1 (Sampling uniformly from a ball)**.** Suppose we want to sample uniformly from $\{(x,y) \in \mathbb{R}^2 : x^2 + y^2 \leq 1\}$. We first need to find the conditional densities. We know the joint density is given by

$$f(x,y) \propto \mathbb{I}(x^2 + y^2 \leq 1).$$

Treating y as a constant, we can rearrange this to see that

$$f(x \,|\, y) \propto \mathbb{I}(x^2 + y^2 \leq 1) \propto \mathbb{I}(x^2 + \leq 1 - y^2) \propto \mathbb{I}(-\sqrt{1-y^2} \leq x \leq \sqrt{1-y^2}).$$

By the same argument, $f(y \,|\, x) \propto \mathbb{I}(-\sqrt{1-x^2} \leq y \leq \sqrt{1-x^2})$. Thus

$$X \,|\, Y \sim \mathcal{U}(-\sqrt{1-Y^2}, \sqrt{1-Y^2}) \quad \text{and} \quad Y \,|\, X \sim \mathcal{U}(-\sqrt{1-X^2}, \sqrt{1-X^2}).$$

Listing 5.5: Gibb's Sampler Code

```
N= 10^4
X=matrix(0, ncol=2, nrow=N+1)
X[1, ]= c(1, 0)
for(i in 1:N)
{
  X[i+1,1] = runif(1, min= -sqrt(1-X[i,2]^2), max=sqrt(1-X[i,2]^2) )
  X[i+1,2] = runif(1, min= -sqrt(1-X[i+1,1]^2), max=sqrt(1-X[i+1,1]^2) )
}
plot(X[,1], X[,2])
```

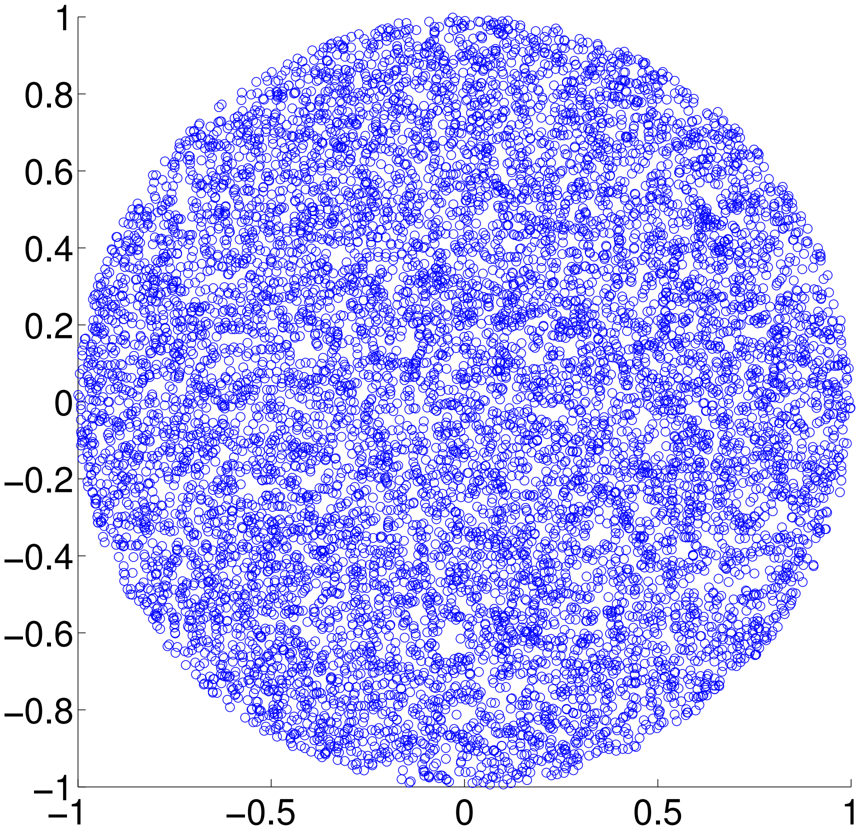

Figure 5.3.1: Plot of a sample from a uniform distribution on the unit ball generated by a Gibb's sampler.

Example 5.3.2 (Sampling from a complicated 2D density)**.** Consider a complicated 2D density, as might appear in Bayesian statistics (this example is taken from [7]). The joint density is of the form

$$f(x, y) \propto \exp\left\{-\frac{x^2}{2}\right\} \exp\left\{-[1 + (x - \lambda)^2]y/2\right\} y^{\nu-1}.$$

Treating y as a constant and completing the square, we have

$$\begin{aligned} f(x \,|\, y) &\propto \exp\left\{-\frac{x^2}{2} - \frac{(x-\lambda)^2 y}{2}\right\} \propto \exp\left\{-\frac{1}{2}\left[x^2 + x^2 y - 2x\lambda y + \lambda^2 y\right]\right\} \\ &\propto \exp\left\{-\frac{1}{2}\left[(1+y)x^2 - 2x\lambda y\right]\right\} \propto \exp\left\{-\frac{1}{2}(1+y)\left[x^2 - \frac{2x\lambda y}{1+y}\right]\right\} \\ &\propto \exp\left\{-\frac{1}{2}(1+y)\left(x - \frac{\lambda y}{1+y}\right)^2\right\}. \end{aligned}$$

Thus, $X \,|\, Y$ is normally distributed with mean $\lambda y/(1+y)$ and variance $(1+y)^{-1}$. Likewise

$$f(y \,|\, x) \propto \exp\left\{-\frac{1 + (x-\lambda)^2}{2} y\right\} y^{\nu-1}.$$

Now, note that the gamma density, with parameters α and β is given by

$$\frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}.$$

Thus, we see that $Y \,|\, X$ is gamma distributed with $\alpha = \nu$ and $\beta = \left[1 + (x-\lambda)^2\right]/2$.

Listing 5.6: Complicated 2 dimensional density

```
N= 10^3
lambda= 0.5
nu= 1.5
X=matrix(0, ncol=2, nrow=N+1)
X[1, ]= c(1, 1)
for(i in 1:N)
{
  X[i+1,1] = rnorm(1, mean= lambda*X[i, 2]/(1+X[i, 2]) , sd= (1+X[i,2])^(-1/2))
  beta= 1+ (1+(X[i+1,1]-lambda)^2)/2
  alpha= nu
  X[i+1,2] = rgamma(1,shape = alpha,scale= 1/beta)
}
plot(X[,1], X[,2])
```

Of course, we need to make sure that the Gibb's sampler works. First, we need to define the forward and backward transition densities for the Gibb's sampler. Let

$$K_{1\to n}(y \,|\, x) = \prod_{i=1}^{d} f(y_i \,|\, y_1, \ldots, y_{i-1}, x_{i+1}, \ldots, x_d)$$

and

$$K_{n\to 1}(x\,|\,y) = \prod_{i=1}^{d} f(x_i\,|\,y_1,\dots,y_{i-1},x_{i+1},\dots,x_d).$$

We then have the following theorem.

Theorem 5.3.3. Let $f(x_i)$ be the ith marginal density of $f(x)$. If $f(y) > 0$ for every $y \in \{x : f(x_i) > 0, i = 1,\dots,n\}$, then

$$f(y)K_{n\to 1}(x\,|\,y) = f(x)K_{1\to n}(y\,|\,x).$$

Proof. See [3]. □

Using this result, one sees that the Gibb's sampler gives the correct distribution by integrating both sides of the equation to get

$$f(y) = \int f(x)K_{1\to n}(y\,|\,x)\,\mathrm{d}x,$$

which shows f is a stationary distribution for the chain.

Bibliography

[1] P. Bremaud. *Markov Chains, Gibbs Fields, Monte Carlo Simulations and Queues.* Springer-Verlag, New York, 1999.

[2] O. Häggström. *Finite Markov Chains and Algorithmic Applications.* London Mathematical Society Student Texts. Cambridge University Press, Cambridge, 2002.

[3] D. P. Kroese, T. Taimre, and Z. I. Botev. *Handbook of Monte Carlo Methods.* John Wiley & Sons, New York, 2011.

[4] D. A. Levin, Y. Peres, and E. L. Wilmer. *Markov Chains and Mixing Times.* American Mathematical Society, Providence, 2009.

[5] S. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability.* Cambridge University Press, Cambridge, 2009.

[6] J. Norris. *Markov Chains.* Cambridge University Press, Cambridge, 1997.

[7] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods.* Springer-Verlag, New York, 2004.