International Series in
Operations Research & Management Science

Wai-Ki Ching
Ximin Huang
Michael K. Ng
Tak-Kuen Siu

Markov Chains

Models, Algorithms and Applications

Second Edition

Springer

International Series in Operations Research & Management Science

Volume 189

For further volumes:
http://www.springer.com/series/6161

Wai-Ki Ching • Ximin Huang
Michael K. Ng • Tak-Kuen Siu

Markov Chains

Models, Algorithms and Applications

Second Edition

Springer

Wai-Ki Ching
Department of Mathematics
The University of Hong Kong
Hong Kong, SAR

Ximin Huang
College of Management
Georgia Institute of Technology
Atlanta, Georgia, USA

Michael K. Ng
Department of Mathematics
Hong Kong Baptist University
Kowloon Tong
Hong Kong SAR

Tak-Kuen Siu
Cass Business School
City University London
London
United Kingdom

ISSN 0884-8289
ISBN 978-1-4614-6311-5 ISBN 978-1-4614-6312-2 (eBook)
DOI 10.1007/978-1-4614-6312-2
Springer New York Heidelberg Dordrecht London

Library of Congress Control Number: 2013931264

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

To Mandy and my Parents
Wai-Ki Ching
To my Parents
Ximin Huang
To Anna, Cecilia and my Parents
Michael K. Ng
To Candy and my Parents
Tak-Kuen Siu

Preface

The aim of this book is to outline the recent development of Markov chain models and their applications in queueing systems, manufacturing systems, remanufacturing systems, inventory systems, ranking the importance of a web site, and also financial risk management.

This book consists of eight chapters. In Chapter 1, we give a brief introduction to the classical theory on both discrete and continuous time Markov chains. The relationship between Markov chains of finite states and matrix theory will also be highlighted. Some classical iterative methods for solving linear systems will be introduced for finding the stationary distribution of a Markov chain. We then give the basic theories and algorithms for hidden Markov models (HMMs) and Markov decision processes (MDPs).

Chapter 2 discusses the applications of continuous time Markov chains to model queueing systems and discrete time Markov chains for computing the PageRank, a ranking of the importance of a web site in the Internet. Chapter 3 studies Markovian models for manufacturing and remanufacturing systems. We present closed form solutions and fast numerical algorithms for solving the captured systems. In Chapter 4, we present a simple hidden Markov model (HMM) with fast numerical algorithms for estimating the model parameters. We then present an application of the HMM for customer classification.

Chapter 5 discusses Markov decision processes for customer lifetime values. Customer lifetime values (CLV) is an important concept and quantity in marketing management. We present an approach based on Markov decision processes for the calculation of CLV using real data.

In Chapter 6, we consider higher-order Markov chain models. In particular, we discuss a class of parsimonious higher-order Markov chain models. Efficient estimation methods for model parameters based on linear programming are presented. Contemporary research results on applications to demand predictions, inventory control, and financial risk measurement are presented. In Chapter 7, a class of parsimonious multivariate Markov models is introduced. Again, efficient estimation methods based on linear programming are presented. Applications to demand predictions, inventory control policy, and modeling credit ratings data are discussed.

In Chapter 8, we revisit hidden Markov models. We propose a new class of hidden Markov models with efficient algorithms for estimating the model parameters. Applications to modeling interest rate, credit ratings, and default data are discussed.

The authors would like to thank Operational Research Society, Oxford University Press, Palgrave, Taylor & Francis', Wiley & Sons, *Journal of Credit Risk* Incisive Media, Incisive Financial Publishing Limited, and Yokohama Publishers for their permission to reproduce the material in this book. The authors would also like to thank Werner Fortmann, Gretel Fortmann, and Mimi Lui for their help in the preparation of this book.

Hong Kong SAR	Wai-Ki Ching
Atlanta, Georgia	Ximin Huang
Kowloon, Hong Kong SAR	Michael K. Ng
Sydney, Australia	Tak-Kuen Siu

Contents

List of Figures

List of Tables

Chapter 1
Introduction

Markov chains are named after Prof. Andrei A. Markov (1856–1922). He was born on June 14, 1856 in Ryazan, Russia and died on July 20, 1922 in St. Petersburg, Russia. Markov enrolled at the University of St. Petersburg, where he earned a master's degree and a doctorate degree. He was a professor at St. Petersburg and also a member of the Russian Academy of Sciences. He retired in 1905, but continued his teaching at the university until his death. Markov is particularly remembered for his study of Markov chains. His research works on Markov chains launched the study of stochastic processes with a lot of applications. For more details about Markov and his works, we refer our reader to the following interesting website (http://www-groups.dcs.st-and.ac.uk/~history/Mathematicians/Markov.html).

In this chapter, we first give a brief introduction to the classical theory on both discrete and continuous time Markov chains. We then present some relationships between Markov chains of finite states and matrix theory. Some classical iterative methods for solving linear systems will be introduced. The iterative methods can be employed to solving the stationary distribution of a Markov chain. We will also give some basic theory and algorithms for standard hidden Markov models (HMMs) and Markov decision processes (MDPs).

1.1 Markov Chains

This section gives a brief introduction to discrete time Markov chains. Interested readers can consult the books by Ross [181] and Häggström [111] for more details.

Markov chains model a sequence of random variables, which correspond to the states of a certain system in such a way that the state at one time depends only on the state in the previous time. We will discuss some basic properties of a Markov chain. Basic concepts and notations are explained throughout this chapter. Some important theorems in this area will also be presented.

W.-K. Ching et al., *Markov Chains*, International Series in Operations Research & Management Science 189, DOI 10.1007/978-1-4614-6312-2_1,

Let us begin with a practical problem for motivation. Marketing research has indicated that in a town there are two supermarkets only, namely Wellcome and Park'n. A marketing research indicated that a consumer of Wellcome may switch to Park'n for their next shopping with a probability of $\alpha(> 0)$, while a consumer of Park'n may switch to Wellcome for their next shopping with a probability of $\beta(> 0)$. Two important and interesting questions that a decision maker would be interested in are: (1) What is the probability that a current Wellcome's consumer will still be shopping at Wellcome for their nth shopping trip? (2) What will be the market share of the two supermarkets in the long-run? An important feature of this problem is that the future behavior of a consumer depends on their current situation. We will see later that this marketing problem can be formulated by using it as a Markov chain model.

1.1.1 Examples of Markov Chains

We consider a stochastic process

$$\{X^{(n)}, n = 0, 1, 2, \ldots\}$$

that takes on a *finite* or *countable* set M.

Example 1.1. Let $X^{(n)}$ be the weather of the nth day which can be

$$M = \{sunny, windy, rainy, cloudy\}.$$

One may have the following realization:

$$X^{(0)} = \text{sunny}, X^{(1)} = \text{windy}, X^{(2)} = \text{rainy}, X^{(3)} = \text{sunny}, X^{(4)} = \text{cloudy}, \ldots.$$

Example 1.2. Let $X^{(n)}$ be the product sales on the nth day which can be

$$M = \{0, 1, 2, \ldots, \}.$$

One may have the following realization:

$$X^{(0)} = 4, \ \ X^{(1)} = 5, \ \ X^{(2)} = 2, \ \ X^{(3)} = 0, \ \ X^{(4)} = 5, \ \ldots.$$

Remark 1.3. For simplicity of discussion we assume M, the *state space* to be

$$\{0, 1, 2, \ldots\}.$$

An element in M is called a *state* of the process.

Definition 1.4. Suppose there is a fixed probability P_{ij} independent of time such that

$$P(X^{(n+1)} = i | X^{(n)} = j, X^{(n-1)} = i_{n-1}, \ldots, X^{(0)} = i_0) = P_{ij} \quad n \geq 0$$

where $i, j, i_0, i_1, \ldots, i_{n-1} \in M$. Then this is called a Markov chain process.

Remark 1.5. One can interpret the above probability as follows: the conditional distribution of any future state $X^{(n+1)}$ given the past states

$$X^{(0)}, X^{(2)}, \ldots, X^{(n-1)}$$

and present state $X^{(n)}$, is *independent* of the *past states* and *depends* on the *present state* only.

Remark 1.6. The probability P_{ij} represents the probability that the process will make a transition to state i given that currently the process is in state j. Clearly one has

$$P_{ij} \geq 0, \quad \sum_{i=0}^{\infty} P_{ij} = 1, \quad j = 0, 1, \ldots.$$

For simplicity of discussion, in our context we adopt this convention which is different from the traditional one.

Definition 1.7. The matrix containing P_{ij}, the transition probabilities

$$P = \begin{pmatrix} P_{00} & P_{01} & \cdots \\ P_{10} & P_{11} & \cdots \\ \vdots & \vdots & \vdots \end{pmatrix}$$

is called the one-step transition probability matrix of the process.

Example 1.8. Consider the marketing problem again. Let $X^{(n)}$ be a 2-state process (taking values in the set $\{0, 1\}$) describing the behavior of a consumer. We have $X^{(n)} = 0$ if the consumer shops with Wellcome on the nth day and $X^{(n)} = 1$ if the consumer shops with Park'n on the nth day. Since the future state (which supermarket to shop in the next time) depends on the current state only, it is a Markov chain process. It is easy to check that the transition probabilities are

$$P_{00} = 1 - \alpha, \quad P_{10} = \alpha, \quad P_{11} = 1 - \beta \quad \text{and} \quad P_{01} = \beta.$$

Then the one-step transition matrix of this process is given by

$$P = \begin{pmatrix} 1-\alpha & \beta \\ \alpha & 1-\beta \end{pmatrix}.$$

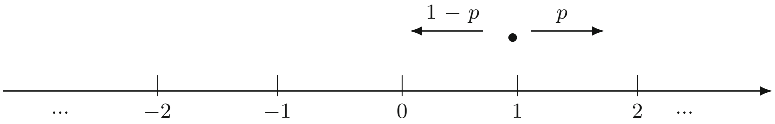


Fig. 1.1 The random walk

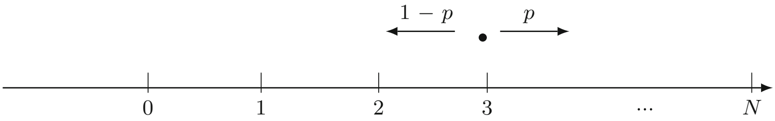


Fig. 1.2 The gambler's ruin

Example 1.9. (Random Walk) Random walks have been studied by many physicists and mathematicians for a number of years. Over time, random walk theory has seen extensions [181] and applications in many fields of study. Therefore it is obvious to discuss the idea of random walks here. Consider a person who performs a random walk on the real line with a line of real counting numbers:

$$\{\ldots,-2,-1,0,1,2,\ldots\}$$

being the state space, see for instance Fig. 1.1. Each time the person at state i can move one step forward (+1) or one step backward (-1) with probabilities p $(0 < p < 1)$ and $(1 - p)$ respectively. Therefore we have the transition probabilities as follow:

$$P_{ij} = \begin{cases} p & \text{if } i = j+1 \\ 1-p & \text{if } i = j-1 \\ 0 & \text{otherwise} \end{cases}$$

for $j = 0, \pm 1, \pm 2, \ldots$.

Example 1.10. (Gambler's Ruin) Consider a gambler gambling in a series of games, at each game, he either wins one dollar with probability p $(0 < p < 1)$ or loses one dollar with probability $(1 - p)$ (Fig. 1.2). The game ends if either he loses all his money or he attains a total amount of N dollars. Let the gambler's fortune be the state of the gambling process then the process is a Markov chain. Moreover, we have the transition probabilities

$$P_{ij} = \begin{cases} p & \text{if } i = j+1 \\ 1-p & \text{if } i = j-1 \\ 0 & \text{otherwise.} \end{cases}$$

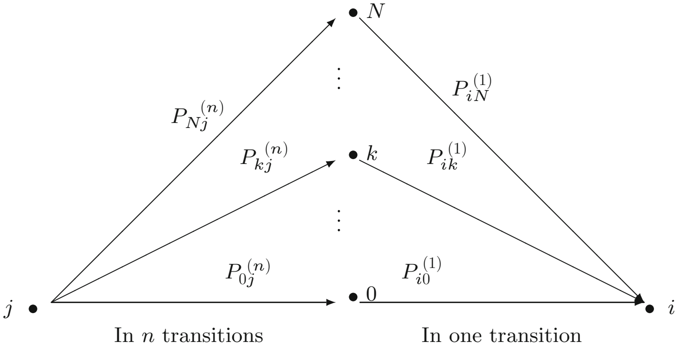


Fig. 1.3 The $(n+1)$-step transition probability

for $j = 1, 2, \ldots, N-1$ and $P_{00} = P_{NN} = 1$. Here state 0 and state N are called the *absorbing states*. The process will stay at 0 or N forever if one of the states is reached.

1.1.2 The nth-Step Transition Matrix

In the previous section, we have defined the one-step transition probability matrix P for a Markov chain process. In this section, we are going to investigate the n-step transition probability $P_{ij}^{(n)}$ of a Markov chain process.

Definition 1.11. Define $P_{ij}^{(n)}$ to be the probability that a process in state j will be in state i after n additional transitions. In particular, we have $P_{ij}^{(1)} = P_{ij}$.

Proposition 1.12. *We have $P^{(n)} = P^n$ where $P^{(n)}$ is the n-step transition probability matrix and P is the one-step transition matrix.*

Proof. We will prove the proposition by using mathematical induction. Clearly the proposition is true when $n = 1$. We then assume that the proposition is true for n. We note that

$$P^n = \underbrace{P \cdot P \cdot \ldots \cdot P}_{n\ times}.$$

Then we have (see Fig. 1.3)

$$P_{ij}^{(n+1)} = \sum_{k \in M} P_{ki}^{(n)} P_{jk}^{(1)} = \sum_{k \in M} P_{ki}^n P_{jk} = [P^{n+1}]_{ij}.$$

By the principle of mathematical induction the proposition is true for all non-negative integer n.

Remark 1.13. It is easy to see that

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

Example 1.14. We consider the marketing problem again. In the model we have

$$P = \begin{pmatrix} 1-\alpha & \beta \\ \alpha & 1-\beta \end{pmatrix}.$$

If $\alpha = 0.3$ and $\beta = 0.4$ then we have

$$P^{(4)} = P^4 = \begin{pmatrix} 0.7 & 0.4 \\ 0.3 & 0.6 \end{pmatrix}^4 = \begin{pmatrix} 0.5749 & 0.5668 \\ 0.4351 & 0.4332 \end{pmatrix}.$$

Recall that a consumer is in state 0 (1) if they are a consumer of Wellcome (Park'n). Here $P_{00}^{(4)} = 0.5749$ is the probability that a Wellcome's consumer will shop with Wellcome on their fourth shopping trip and $P_{10}^{(4)} = 0.4351$ is the probability that a Wellcome's consumer will shop with Park'n on their fourth shopping trip. And $P_{01}^{(4)} = 0.5668$ is the probability that a consumer of Park'n will shop with Wellcome on their fourth shopping trip, while $P_{11}^{(4)} = 0.4332$ is the probability that a consumer of Park'n will shop with Park'n on their fourth shopping trip.

Remark 1.15. Consider a Markov chain process having states in $\{0, 1, 2, \ldots\}$. Suppose that we are given at time $n = 0$ the probability that the process is in state i is $a_i, i = 0, 1, 2, \ldots,$ then one interesting question is the following: What is the probability that the process will be in state j after n transitions? In fact, the probability that given the process is in state i and it will be in state j after n transitions is $P_{ji}^{(n)} = [P^n]_{ji}$, where P_{ji} is the one-step transition probability from state i to state j of the process. Therefore the required probability is

$$\sum_{i=0}^{\infty} P(X^{(0)} = i) \cdot P_{ji}^{(n)} = \sum_{i=0}^{\infty} a_i \cdot [P^n]_{ji}.$$

Let

$$\mathbf{X}^{(n)} = (\tilde{X}_0^{(n)}, \tilde{X}_1^{(n)}, \ldots,)$$

be the probability distribution of the states in a Markov chain process at the nth transition. Here $\tilde{X}_i^{(n)}$ is the probability that the process is in state i after n transitions and

$$\sum_{i=0}^{\infty} \tilde{X}_i^{(n)} = 1.$$

It is easy to check that

$$\mathbf{X}^{(n+1)} = P\mathbf{X}^{(n)}$$

and

$$\mathbf{X}^{(n+1)} = P^{(n+1)}\mathbf{X}^{(0)} = P^n\mathbf{X}^{(0)}.$$

Example 1.16. Refer to the previous example. If at $n = 0$ a consumer belongs to Park'n, we may represent this information as

$$\mathbf{X}^{(0)} = (\tilde{X}_0^{(0)}, \tilde{X}_1^{(0)})^T = (0, 1)^T.$$

Where will be the consumer shop on their fourth shopping trip?

$$\mathbf{X}^{(4)} = P^{(4)}\mathbf{X}^{(0)} = \begin{pmatrix} 0.7 & 0.4 \\ 0.3 & 0.6 \end{pmatrix}^4 (0, 1)^T = (0.5668, 0.4332)^T.$$

This means that with a probability 0.4332 they will be a consumer of Park'n and with a probability of 0.5668 they will be a consumer of Wellcome on their fourth shopping trip.

1.1.3 Irreducible Markov Chain and Classifications of States

In the following, we define two definitions for the states of a Markov chain.

Definition 1.17. In a Markov chain, state i is said to be reachable from state j if $P_{ij}^{(n)} > 0$ for some $n \geq 0$. This means that starting from state j, it is possible (with positive probability) to enter state i in a finite number of transitions.

Definition 1.18. State i and state j are said to *communicate* if state i and state j are reachable from each other.

Remark 1.19. The definition of communication defines an equivalent relation.

(i) state i communicates with state i in 0 step because

$$P_{ii}^{(0)} = P(X^{(0)} = i | X^{(0)} = i) = 1 > 0.$$

(ii) If state i communicates with state j, then state j communicates with state i.

(iii) If state i communicates with state j and state j communicates with state k then state i communicates with state k. Since $P_{ji}^{(m)}, P_{kj}^{(n)} > 0$ for some m and n, we have

$$P_{ki}^{(m+n)} = \sum_{h \in M} P_{hi}^{(m)} P_{kh}^{(n)} \geq P_{ji}^{(m)} P_{kj}^{(n)} > 0.$$

Therefore state k is reachable from state i. By inter-changing the roles of i and k, state i is reachable from state k. Hence i communicates with k. The proof is then completed.

Definition 1.20. Two states that communicate are said to be in the same *class*. A Markov chain is said to be *irreducible*, if all states belong to the same class, i.e. they communicate with each other.

Example 1.21. Consider the transition probability matrix

$$\begin{array}{c} 0 \\ 1 \\ 2 \end{array} \begin{pmatrix} 0.0 & 0.5 & 0.5 \\ 0.5 & 0.0 & 0.5 \\ 0.5 & 0.5 & 0.0 \end{pmatrix}.$$

Example 1.22. Consider another transition probability matrix

$$\begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{pmatrix} 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.5 & 0.5 \\ 0.0 & 0.5 & 0.0 & 0.5 \\ 0.0 & 0.5 & 0.5 & 0.0 \end{pmatrix}.$$

We note that from state 1, 2, 3, it is not possible to visit state 0, i.e

$$P_{01}^{(n)} = P_{02}^{(n)} = P_{03}^{(n)} = 0.$$

Therefore the Markov chain is not irreducible (or it is reducible).

Definition 1.23. For any state i in a Markov chain, let f_i be the probability that starting in state i, the process will ever re-enter state i. State i is said to be *recurrent* if $f_i = 1$ and *transient* if $f_i < 1$.

We have the following proposition for a recurrent state.

Proposition 1.24. *In a finite Markov chain, a state i is recurrent if and only if*

$$\sum_{n=1}^{\infty} P_{ii}^{(n)} = \infty.$$

The proposition implies that a transient state will only be visited a finite number of times. Thus it is easy to see that in a Markov chain of finite states, we cannot have all states being transient. By using Proposition 1.24 one can prove the following proposition.

Proposition 1.25. *In a finite Markov chain, if state i is recurrent (transient) and state i communicates with state j then state j is also recurrent (transient).*

1.1.4 An Analysis of the Random Walk

Recall the classical example of a random walk (the analysis of the random walk can also be found in Ross [181]). A person performs a random walk on the real line of integers. At each time point the person at state i can move one step forward (+1) or one step backward (-1) with probabilities p $(0 < p < 1)$ and $(1-p)$ respectively. Since all the states communicate, by Proposition 1.25, then all states are either recurrent or they are all transient.

Let us consider state 0. To classify this state one can consider the following sum:

$$\sum_{m=1}^{\infty} P_{00}^{(m)}.$$

We note that

$$P_{00}^{(2n+1)} = 0$$

because in order to return to state 0, the number of forward movements should be equal to the number of backward movements and therefore the number of movements should be even and

$$P_{00}^{(2n)} = \binom{2n}{n} p^n (1-p)^n.$$

Hence we have

$$I = \sum_{m=1}^{\infty} P_{00}^{(m)} = \sum_{n=1}^{\infty} P_{00}^{(2n)} = \sum_{n=1}^{\infty} \binom{2n}{n} p^n (1-p)^n = \sum_{n=1}^{\infty} \frac{(2n)!}{n!n!} p^n (1-p)^n.$$

Recall that if I is finite then state 0 is transient or it is recurrent. We can then apply the Stirling's formula to get a conclusive result. The Stirling's formula states that if n is large then

$$n! \approx n^{n+\frac{1}{2}} e^{-n} \sqrt{2\pi}.$$

Hence one can approximate

$$P_{00}^{(2n)} \approx \frac{(4p(1-p))^n}{\sqrt{\pi n}}.$$

There are two cases to consider. If $p = \frac{1}{2}$ then we have

$$P_{00}^{(2n)} \approx \frac{1}{\sqrt{\pi n}}.$$

If $p \neq \frac{1}{2}$ then we have

$$P_{00}^{(2n)} \approx \frac{a^n}{\sqrt{\pi n}}$$

where

$$0 < a = 4p(1-p) < 1.$$

Therefore when $p = \frac{1}{2}$, state 0 is recurrent as the sum is infinite, and when $p \neq \frac{1}{2}$, state 0 is transient as the sum is finite.

1.1.5 Simulation of Markov Chains with EXCEL

Consider a Markov chain process with three states $\{0, 1, 2\}$ with the transition probability matrix as follows:

$$P = \begin{array}{cc} \begin{array}{c} 0 \\ 1 \\ 2 \end{array} & \begin{pmatrix} 0.2 & 0.5 & 0.3 \\ 0.3 & 0.1 & 0.3 \\ 0.5 & 0.4 & 0.4 \end{pmatrix} \end{array}.$$

Given that $X_0 = 0$, our objective here is to generate a sequence

$$\{X^{(n)}, n = 1, 2, \ldots\}$$

which follows a Markov chain process with the transition matrix P.

To generate $\{X^{(n)}\}$ there are three possible cases:

(i) Suppose $X^{(n)} = 0$, then we have

$$P(X^{(n+1)} = 0) = 0.2 \qquad P(X^{(n+1)} = 1) = 0.3 \qquad P(X^{(n+1)} = 2) = 0.5;$$

(ii) Suppose $X^{(n)} = 1$, then we have

$$P(X^{(n+1)} = 0) = 0.5 \qquad P(X^{(n+1)} = 1) = 0.1 \qquad P(X^{(n+1)} = 2) = 0.4;$$

(iii) Suppose $X^{(n)} = 2$, then we have

$$P(X^{(n+1)} = 0) = 0.3 \qquad P(X^{(n+1)} = 1) = 0.3 \qquad P(X^{(n+1)} = 2) = 0.4.$$

Suppose we can generate a random variable U which is uniformly distributed over $[0, 1]$. Then one can easily generate the distribution in Case (i) when $X^{(n)} = 0$ easily as follows:

$$X^{(n+1)} = \begin{cases} 0 & \text{if} \quad U \in [0, 0.2), \\ 1 & \text{if} \quad U \in [0.2, 0.5), \\ 2 & \text{if} \quad U \in [0.5, 1]. \end{cases}$$

The distribution in Case (ii) when $X^{(n)} = 1$ can be generated as follows:

$$X^{(n+1)} = \begin{cases} 0 & \text{if} \quad U \in [0, 0.5), \\ 1 & \text{if} \quad U \in [0.5, 0.6), \\ 2 & \text{if} \quad U \in [0.6, 1]. \end{cases}$$

The distribution in Case (iii) when $X^{(n)} = 2$ can be generated as follows:

$$X^{(n+1)} = \begin{cases} 0 & \text{if} \quad U \in [0, 0.3), \\ 1 & \text{if} \quad U \in [0.3, 0.6), \\ 2 & \text{if} \quad U \in [0.6, 1]. \end{cases}$$

In EXCEL, one can generate U, a random variable uniformly distributed over [0, 1] by using "=rand()". By using a simple logic statement in EXCEL, one can simulate a Markov chain easily, see for instance Fig. 1.4. The followings are some useful logic statements in EXCEL used in the demonstration file.

(i) "B1" means column B and Row 1.
(ii) "=IF(B1=0,1,-1)" return 1 if B1=0 otherwise it returns -1.
(iii) "=IF(A1 > B2,0,1)" return 0 if A1 > B2 otherwise it returns 1.
(iv) "=IF(AND(A1=1,B2>2),1,0)" return 1 if A1=1 and B2>2 otherwise it returns 0.
(v) "=max(1,2,-1) =2 " returns the maximum of the numbers.

A demonstration EXCEL file is available at http://hkumath.hku.hk/~wkc/sim.xls for reference. The program generates a Markov chain process

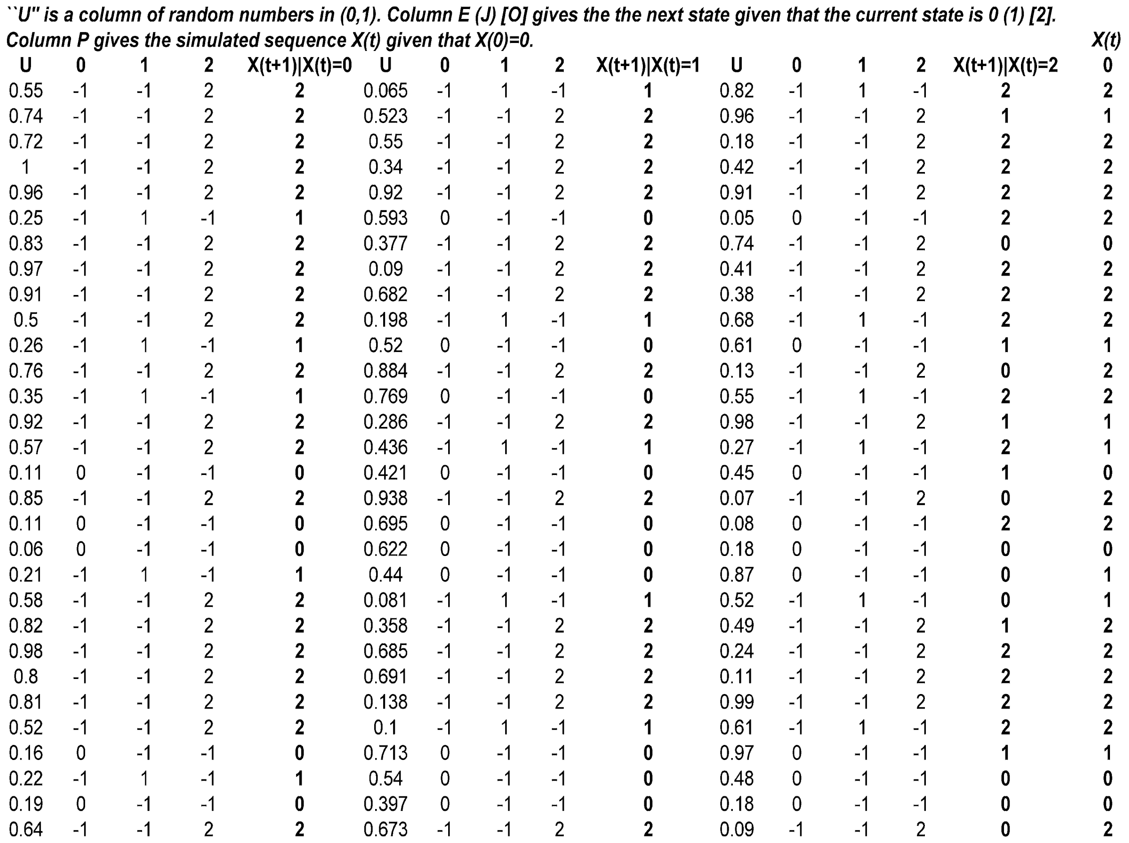
$$X^{(1)}, X^{(2)}, \ldots, X^{(30)}$$

whose transition probability is P and $X^{(0)} = 0$.

1.1.6 Building a Markov Chain Model

Given an observed data sequence $\{X^{(n)}\}$, one can find the transition frequency F_{jk} in the sequence by counting the number of transitions from state j to state k in one step. One can then construct the one-step transition matrix for the sequence $\{X^{(n)}\}$ as follows:

$$F = \begin{pmatrix} F_{11} & \cdots\cdots & F_{1m} \\ F_{21} & \cdots\cdots & F_{2m} \\ \vdots & \vdots \quad \vdots & \vdots \\ F_{m1} & \cdots\cdots & F_{mm} \end{pmatrix}. \tag{1.1}$$



``U" is a column of random numbers in (0,1). Column E (J) [O] gives the the next state given that the current state is 0 (1) [2].
Column P gives the simulated sequence X(t) given that X(0)=0.

															X(t)
U	**0**	**1**	**2**	**X(t+1)|X(t)=0**	**U**	**0**	**1**	**2**	**X(t+1)|X(t)=1**	**U**	**0**	**1**	**2**	**X(t+1)|X(t)=2**	**0**
0.55	-1	-1	2	**2**	0.065	-1	1	-1	**1**	0.82	-1	1	-1	**2**	**2**
0.74	-1	-1	2	**2**	0.523	-1	-1	2	**2**	0.96	-1	-1	2	**1**	**1**
0.72	-1	-1	2	**2**	0.55	-1	-1	2	**2**	0.18	-1	-1	2	**2**	**2**
1	-1	-1	2	**2**	0.34	-1	-1	2	**2**	0.42	-1	-1	2	**2**	**2**
0.96	-1	-1	2	**2**	0.92	-1	-1	2	**2**	0.91	-1	-1	2	**2**	**2**
0.25	-1	1	-1	**1**	0.593	0	-1	-1	**0**	0.05	0	-1	-1	**2**	**2**
0.83	-1	-1	2	**2**	0.377	-1	-1	2	**2**	0.74	-1	-1	2	**0**	**0**
0.97	-1	-1	2	**2**	0.09	-1	-1	2	**2**	0.41	-1	-1	2	**2**	**2**
0.91	-1	-1	2	**2**	0.682	-1	-1	2	**2**	0.38	-1	-1	2	**2**	**2**
0.5	-1	-1	2	**2**	0.198	-1	1	-1	**1**	0.68	-1	1	-1	**2**	**2**
0.26	-1	1	-1	**1**	0.52	0	-1	-1	**0**	0.61	0	-1	-1	**1**	**1**
0.76	-1	-1	2	**2**	0.884	-1	-1	2	**2**	0.13	-1	-1	2	**0**	**2**
0.35	-1	1	-1	**1**	0.769	0	-1	-1	**0**	0.55	-1	1	-1	**2**	**2**
0.92	-1	-1	2	**2**	0.286	-1	-1	2	**2**	0.98	-1	-1	2	**1**	**1**
0.57	-1	-1	2	**2**	0.436	-1	1	-1	**1**	0.27	-1	1	-1	**2**	**1**
0.11	0	-1	-1	**0**	0.421	0	-1	-1	**0**	0.45	0	-1	-1	**1**	**0**
0.85	-1	-1	2	**2**	0.938	-1	-1	2	**2**	0.07	-1	-1	2	**0**	**2**
0.11	0	-1	-1	**0**	0.695	0	-1	-1	**0**	0.08	0	-1	-1	**2**	**2**
0.06	0	-1	-1	**0**	0.622	0	-1	-1	**0**	0.18	0	-1	-1	**0**	**0**
0.21	-1	1	-1	**1**	0.44	0	-1	-1	**0**	0.87	0	-1	-1	**0**	**1**
0.58	-1	-1	2	**2**	0.081	-1	1	-1	**1**	0.52	-1	1	-1	**0**	**1**
0.82	-1	-1	2	**2**	0.358	-1	-1	2	**2**	0.49	-1	-1	2	**1**	**2**
0.98	-1	-1	2	**2**	0.685	-1	-1	2	**2**	0.24	-1	-1	2	**2**	**2**
0.8	-1	-1	2	**2**	0.691	-1	-1	2	**2**	0.11	-1	-1	2	**2**	**2**
0.81	-1	-1	2	**2**	0.138	-1	-1	2	**2**	0.99	-1	-1	2	**2**	**2**
0.52	-1	-1	2	**2**	0.1	-1	1	-1	**1**	0.61	-1	1	-1	**2**	**2**
0.16	0	-1	-1	**0**	0.713	0	-1	-1	**0**	0.97	0	-1	-1	**1**	**1**
0.22	-1	1	-1	**1**	0.54	0	-1	-1	**0**	0.48	0	-1	-1	**0**	**0**
0.19	0	-1	-1	**0**	0.397	0	-1	-1	**0**	0.18	0	-1	-1	**0**	**0**
0.64	-1	-1	2	**2**	0.673	-1	-1	2	**2**	0.09	-1	-1	2	**0**	**2**

Fig. 1.4 Simulation of a Markov chain

From F, one can get the estimates for P_{jk} as follows:

$$P = \begin{pmatrix} P_{11} & \cdots\cdots & P_{1m} \\ P_{21} & \cdots\cdots & P_{2m} \\ \vdots & \vdots\ \vdots & \vdots \\ P_{m1} & \cdots\cdots & P_{mm} \end{pmatrix} \tag{1.2}$$

where

$$P_{jk} = \begin{cases} \dfrac{F_{jk}}{\sum\limits_{j=1}^{m} F_{jk}} & \text{if} \quad \sum\limits_{j=1}^{m} F_{jk} > 0 \\ 0 \text{ if } \sum\limits_{j=1}^{m} F_{jk} = 0. \end{cases}$$

We consider a sequence $\{X^{(n)}\}$ of three states ($m = 3$) given by

$$\{0, 0, 1, 1, 0, 2, 1, 0, 1, 2, 0, 1, 2, 0, 1, 2, 0, 1, 0, 1\}. \tag{1.3}$$

Using the counting method (see Fig. 1.5), we can obtain the transition frequency matrix

$$F = \begin{pmatrix} 1 & 3 & 3 \\ 6 & 1 & 1 \\ 1 & 3 & 0 \end{pmatrix}. \tag{1.4}$$

Therefore the one-step transition probability matrix can be estimated as follows:

$$P = \begin{pmatrix} 1/8 & 3/7 & 3/4 \\ 3/4 & 1/7 & 1/4 \\ 1/8 & 3/7 & 0 \end{pmatrix}. \tag{1.5}$$

A demonstration EXCEL file is available at http://hkumath.hku.hk/~wkc/build.xls for reference.

1.1.7 Stationary Distribution of a Finite Markov Chain

Definition 1.26. State i is said to have period d if $P_{ii}^{(n)} = 0$ whenever n is not divisible by d, and d is the largest integer with this property. A state with period 1 is said to be aperiodic.

X(t)	P00	P01	P02	P10	P11	P12	P20	P21	P22
0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0
1	0	0	0	1	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
2	0	0	0	0	0	0	0	1	0
1	0	0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0	0	0
2	0	0	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0	0	0
2	0	0	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0	0	0
2	0	0	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0
F(ij)	1	6	1	4	1	3	3	1	0
P(ij)	0.125	0.75	0.125	0.5	0.125	0.375	0.75	0.25	0

Fig. 1.5 The construction of the transition probability matrix

Example 1.27. Consider the transition probability matrix

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

We note that

$$P^{(n)} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}^n = \frac{1}{2} \begin{pmatrix} 1 + (-1)^n & 1 + (-1)^{n+1} \\ 1 + (-1)^{n+1} & 1 + (-1)^n \end{pmatrix}.$$

We note that

$$P_{00}^{(2n+1)} = P_{11}^{(2n+1)} = 0$$

so both States 0 and 1 have a period of 2.

Definition 1.28. State i is said to be positive recurrent if it is recurrent, and starting in state i, the expected time until the process returns to state i is finite.

Definition 1.29. A state is said to be ergodic if it is positive recurrent and aperiodic.

We recall the example of the marketing problem with

$$\mathbf{X}^{(0)} = (1, 0)^T.$$

We observe that

$$\mathbf{X}^{(1)} = P\mathbf{X}^{(0)} = \begin{pmatrix} 0.7 & 0.4 \\ 0.3 & 0.6 \end{pmatrix} (1, 0)^T = (0.7, 0.3)^T,$$

$$\mathbf{X}^{(2)} = P^2\mathbf{X}^{(0)} = \begin{pmatrix} 0.61 & 0.52 \\ 0.39 & 0.48 \end{pmatrix} (1, 0)^T = (0.61, 0.39)^T,$$

$$\mathbf{X}^{(4)} = P^4\mathbf{X}^{(0)} = \begin{pmatrix} 0.5749 & 0.5668 \\ 0.4251 & 0.4332 \end{pmatrix} (1, 0)^T = (0.5749, 0.4251)^T,$$

$$\mathbf{X}^{(8)} = P^8\mathbf{X}^{(0)} = \begin{pmatrix} 0.5715 & 0.5714 \\ 0.4285 & 0.4286 \end{pmatrix} (1, 0)^T = (0.5715, 0.4285)^T,$$

$$\mathbf{X}^{(16)} = P^{16}\mathbf{X}^{(0)} = \begin{pmatrix} 0.5714 & 0.5174 \\ 0.4286 & 0.4286 \end{pmatrix} (1, 0)^T = (0.5714, 0.4286)^T.$$

It seems that

$$\lim_{n \to \infty} \mathbf{X}^{(n)} = (0.5714, 0.4286)^T.$$

In fact this limit exists and is also independent of $\mathbf{X}^{(0)}$! This means that in the long run, the probability that a consumer belongs to Wellcome (Park'n) is given by 0.57 (0.43).

We note that $\mathbf{X}^{(n)} = P\mathbf{X}^{(n-1)}$ therefore if we let

$$\lim_{n \to \infty} \mathbf{X}^{(n)} = \boldsymbol{\pi}$$

then

$$\boldsymbol{\pi} = \lim_{n \to \infty} \mathbf{X}^{(n)} = \lim_{n \to \infty} P\mathbf{X}^{(n-1)} = P\boldsymbol{\pi}.$$

This leads us to Definition 1.30. We have the following definition

Definition 1.30. A vector $\boldsymbol{\pi} = (\pi_0, \pi_1, \ldots, \pi_{k-1})^T$ is said to be a stationary distribution of a finite Markov chain if it satisfies:

(i)

$$\pi_i \geq 0 \quad \text{and} \quad \sum_{i=0}^{k-1} \pi_i = 1.$$

(ii)

$$P\boldsymbol{\pi} = \boldsymbol{\pi}, \quad \text{i.e.} \quad \sum_{j=0}^{k-1} P_{ij}\pi_j = \pi_i.$$

$$\lim_{n\to\infty} ||\mathbf{X}^{(n)} - \boldsymbol{\pi}|| = \lim_{n\to\infty} ||P^n\mathbf{X}^{(0)} - \boldsymbol{\pi}|| = 0$$

then $\boldsymbol{\pi}$ is also called the steady-state probability distribution and $||.||$ is a vector norm.

Proposition 1.31. *For any irreducible and aperiodic Markov chain having k states, there exists at least one stationary distribution.*

Proposition 1.32. *For any irreducible and aperiodic Markov chain having k states, for any initial distribution* $\mathbf{X}^{(0)}$

$$\lim_{n\to\infty} ||\mathbf{X}^{(n)} - \boldsymbol{\pi}|| = \lim_{n\to\infty} ||P^n\mathbf{X}^{(0)} - \boldsymbol{\pi}|| = 0.$$

where $\boldsymbol{\pi}$ *is a stationary distribution for the transition matrix* P *and* $||.||$ *is a vector norm.*

Proposition 1.33. *The steady-state probability distribution* $\boldsymbol{\pi}$ *in Proposition 1.32 is unique.*

Remark 1.34. An irreducible finite Markov chain has a unique stationary distribution vector but it may have no steady-state probability distribution (one may consider Example 1.27). In this case, one has to interpret the stationary distribution as follows, as it gives the proportion of the occurrence of the states in the Markov chain in the long run.

To measure the distance between two vectors, we have to introduce a norm. In fact, there are many vector norms $||.||$. In the following, we introduce the definition of a vector norm in $\mathbb{R}^n$ with three popular examples.

Definition 1.35. On the vector space $V = \mathbb{R}^n$, a norm is a function $\|\cdot\|$ from $\mathbb{R}^n$ to the set of non-negative real numbers such that

(1) $\|\mathbf{x}\| > 0$ for all $\mathbf{x} \in V$ and $\mathbf{x} \neq \mathbf{0}$
(2) $\|\lambda\mathbf{x}\| = |\lambda|\|\mathbf{x}\|$ for all $\mathbf{x} \in V, \lambda \in \mathbb{R}$
(3) $\|\mathbf{x}+\mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$ for all $\mathbf{x}, \mathbf{y} \in V$.

The followings are L_1-norm, L_∞-norm and L_2-norm defined respectively by

$$||\mathbf{v}||_1 = \sum_{i=1}^{n} |v_i|, \quad ||\mathbf{v}||_\infty = \max_i\{|v_i|\} \quad \text{and} \quad ||\mathbf{v}||_2 = \sqrt{\sum_{i=1}^{n} |v_i|^2}.$$

In general we have the following proposition. All the three vector norms are particular examples of

$$||\mathbf{v}||_p = \left(\sum_{i=1}^{n} |v_i|^p\right)^{\frac{1}{p}}$$

where $p \geq 1$. In particular, we have (left as an exercise)

$$||\mathbf{v}||_\infty = \lim_{p \to \infty} \left(\sum_{i=1}^{n} |v_i|^p\right)^{\frac{1}{p}}.$$

Proposition 1.36. *For $p \geq 1$, the following is a vector norm on $\mathbb{R}^n$*

$$||\mathbf{x}||_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}.$$

Proof. We leave the case of $p = 1$ as an exercise and we shall consider $p > 1$. We have to prove the following:

(1) It is clear that if $\mathbf{x} \neq \mathbf{0}$ then $||\mathbf{x}||_p > 0$.
(2) We have

$$||\lambda\mathbf{x}||_p = \left(\sum_{i=1}^{n} |\lambda x_i|^p\right)^{\frac{1}{p}} = |\lambda| \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}} = |\lambda|||\mathbf{x}||_p.$$

(3) Finally we have to show that $||\mathbf{x} + \mathbf{y}||_p \leq ||\mathbf{x}||_p + ||\mathbf{y}||_p$, i.e.

$$\left(\sum_{i=1}^{n} |x_i + y_i|^p\right)^{\frac{1}{p}} \leq \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}} + \left(\sum_{i=1}^{n} |y_i|^p\right)^{\frac{1}{p}}.$$

Note that if either $\mathbf{x}$ or $\mathbf{y}$ is the zero vector then the result is easy to see. Here we assume that both $\mathbf{x}$ or $\mathbf{y}$ are non-zero vectors.

In the proof we need the following inequality and the proof is left as an exercise. Let $p > 1$ and q be defined such that

$$\frac{1}{p} + \frac{1}{q} = 1$$

then for $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{y} = (y_1, \ldots, y_n)$, we have

$$\sum_{i=1}^{n} |x_i y_i| \leq \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}} \left(\sum_{i=1}^{n} |y_i|^q\right)^{\frac{1}{q}}. \tag{1.6}$$

Now for $p > 1$ and $\mathbf{x}, \mathbf{y} \neq \mathbf{0}$, we have

$$\sum_{i=1}^{n} |x_i + y_i|^p = \sum_{i=1}^{n} |x_i + y_i||x_i + y_i|^{p-1} = \sum_{i=1}^{n} |x_i||x_i + y_i|^{p-1} + \sum_{i=1}^{n} |y_i||x_i + y_i|^{p-1}$$

By (1.6), we have

$$\sum_{i=1}^{n} |x_i||x_i + y_i|^{p-1} \le \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}} \left(\sum_{i=1}^{n} |x_i + y_i|^{(p-1)q}\right)^{\frac{1}{q}}$$

and

$$\sum_{i=1}^{n} |y_i||x_i + y_i|^{p-1} \le \left(\sum_{i=1}^{n} |y_i|^p\right)^{\frac{1}{p}} \left(\sum_{i=1}^{n} |x_i + y_i|^{(p-1)q}\right)^{\frac{1}{q}}.$$

Hence we get

$$\sum_{i=1}^{n} |x_i + y_i|^p \le \left(\left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}} + \left(\sum_{i=1}^{n} |y_i|^p\right)^{\frac{1}{p}}\right) \left(\sum_{i=1}^{n} |x_i + y_i|^p\right)^{\frac{1}{q}}$$

and by re-arranging the terms we have

$$\left(\sum_{i=1}^{n} |x_i + y_i|^p\right)^{1-\frac{1}{q}} \le \left(\left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}} + \left(\sum_{i=1}^{n} |y_i|^p\right)^{\frac{1}{p}}\right).$$

The result follows as we have $1 - \frac{1}{q} = \frac{1}{p}$.

1.1.8 Applications of the Stationary Distribution

Returning to the marketing problem, the transition matrix is given by:

$$P = \begin{pmatrix} 1-\alpha & \beta \\ \alpha & 1-\beta \end{pmatrix}.$$

To solve for the stationary distribution $(\pi_0, \pi_1)^T$, we consider the following linear system of equations

$$\begin{cases} (1-\alpha)\pi_0 + \beta\pi_1 & = \pi_0 \\ \alpha\pi_0 + (1-\beta)\pi_1 & = \pi_1 \\ \pi_0 + \pi_1 & = 1. \end{cases}$$

Solving the linear system of equations we have

$$\begin{cases} \pi_0 = \beta(\alpha+\beta)^{-1} \\ \pi_1 = \alpha(\alpha+\beta)^{-1}. \end{cases}$$

Therefore in the long run, the market shares of Wellcome and Park'n are respectively given by

$$\frac{\beta}{(\alpha+\beta)} \quad \text{and} \quad \frac{\alpha}{(\alpha+\beta)}.$$

1.2 Continuous Time Markov Chain Process

In the previous section, we have discussed discrete time Markov chain processes. In many situations, a change of state does not occur at a fixed discrete time. In fact, the duration of a system state can be a continuous random variable. In our context, we are going to model queueing systems and re-manufacturing systems by using continuous time Markov processes. We first begin with a definition for a Poisson process which is commonly used in modeling continuous time Markov chain processes. We then give some important properties of the Poisson process.

A process is called a Poisson process if:

(A1) The probability of occurrence of one event in the time interval $(t, t+\delta t)$ is $\lambda\delta t + o(\delta t)$. Here λ is a positive constant and $o(\delta t)$ is such that

$$\lim_{\delta t \to 0} \frac{o(\delta t)}{\delta t} = 0.$$

(A2) The probability of occurrence of no event in the time interval $(t, t+\delta t)$ is $1 - \lambda\delta t + o(\delta t)$.

(A3) The probability of occurrences of more than one event is $o(\delta t)$.

Here an "event" can be an arrival of a bus or a departure of customer. From the above assumptions, one can derive the well-known Poisson distribution.

Let $P_n(t)$ be the probability that n events occurred in the time interval $[0, t]$. Assuming that $P_n(t)$ is differentiable, then we can get a relationship between $P_n(t)$ and $P_{n-1}(t)$ as follows:

$$P_n(t+\delta t) = P_n(t) \cdot (1 - \lambda\delta t - o(\delta t)) + P_{n-1}(t) \cdot (\lambda\delta t + o(\delta t)) + o(\delta t).$$

Rearranging the terms we get

$$\frac{P_n(t+\delta t) - P_n(t)}{\delta t} = -\lambda P_n(t) + \lambda P_{n-1}(t) + (P_{n-1}(t) + P_n(t))\frac{o(\delta t)}{\delta t}.$$

If we let δt go to zero, we have

$$\lim_{\delta t \to 0} \frac{P_n(t+\delta t) - P_n(t)}{\delta t} = -\lambda P_n(t) + \lambda P_{n-1}(t) + \lim_{\delta t \to 0} (P_{n-1}(t) + P_n(t)) \frac{o(\delta t)}{\delta t}.$$

Hence we have the differential-difference equation:

$$\frac{dP_n(t)}{dt} = -\lambda P_n(t) + \lambda P_{n-1}(t) + 0, \quad n = 0, 1, 2, \ldots.$$

Since $P_{-1}(t) = 0$, we have the *initial value problem* for $P_0(t)$ as follows:

$$\frac{dP_0(t)}{dt} = -\lambda P_0(t) \quad \text{with} \quad P_0(0) = 1.$$

The probability $P_0(0)$ is the probability that no event occurred in the time interval $[0, 0]$, so it must be one. Solving the separable ordinary differential equation for $P_0(t)$ we get

$$P_0(t) = e^{-\lambda t}$$

which is the probability that no event occurred in the time interval $[0, t]$. Thus

$$1 - P_0(t) = 1 - e^{-\lambda t}$$

is the probability that at least one event occurred in the time interval $[0, t]$. Therefore the probability density function $f(t)$ for the waiting time of the first event to occur is given by the well-known exponential distribution

$$f(t) = \frac{d(1 - e^{-\lambda t})}{dt} = \lambda e^{-\lambda t}, \quad t \geq 0.$$

We note that

$$\begin{cases} \dfrac{dP_n(t)}{dt} = -\lambda P_n(t) + \lambda P_{n-1}(t), \quad n = 1, 2, \ldots \\ P_0(t) = e^{-\lambda t}, \\ P_n(0) = 0 \quad n = 1, 2, \ldots. \end{cases}$$

Solving the above differential-difference equations, we get

$$P_n(t) = \frac{(\lambda t)^n}{n!} e^{-\lambda t}.$$

Finally, we present the important relationships among the Poisson process, Poisson distribution and the exponential distribution [46].

Proposition 1.37. *[181] The following statements (B1),(B2) and (B3) are equivalent.*

(B1) The arrival process is a Poisson process with mean rate λ.

(B2) Let $N(t)$ be the number of arrivals in the time interval $[0, t]$ then

$$P(N(t) = n) = \frac{(\lambda t)^n e^{-\lambda t}}{n!} \quad n = 0, 1, 2, \ldots.$$

(B3) The inter-arrival time follows the exponential distribution with mean λ^{-1}.

1.2.1 A Continuous Two-State Markov Chain

Consider a one-server queueing system which has two possible states: 0 (idle) and 1 (busy). Assume that the arrival process of the customers is a Poisson process with mean rate λ and the service time of the server follows the exponential distribution with mean rate μ. Let $P_0(t)$ be the probability that the server is idle at time t and $P_1(t)$ be the probability that the server is busy at time t. Using a similar argument as in the derivation of a Poisson process, we have

$$\begin{cases} P_0(t + \delta t) = (1 - \lambda\delta t - o(\delta t))P_0(t) + (\mu\delta t + o(\delta t))P_1(t) + o(\delta t) \\ P_1(t + \delta t) = (1 - \mu\delta t - o(\delta t))P_1(t) + (\lambda\delta t + o(\delta t))P_0(t) + o(\delta t). \end{cases}$$

Rearranging the terms, one gets

$$\begin{cases} \dfrac{P_0(t + \delta t) - P_0(t)}{\delta t} = -\lambda P_0(t) + \mu P_1(t) + (P_1(t) - P_0(t))\dfrac{o(\delta t)}{\delta t} \\ \dfrac{P_1(t + \delta t) - P_1(t)}{\delta t} = \lambda P_0(t) - \mu P_1(t) + (P_0(t) - P_1(t))\dfrac{o(\delta t)}{\delta t}. \end{cases}$$

Letting δt go to zero, we get

$$\begin{cases} \dfrac{dP_0(t)}{dt} = -\lambda P_0(t) + \mu P_1(t) \\ \dfrac{dP_1(t)}{dt} = \lambda P_0(t) - \mu P_1(t). \end{cases}$$

Solving the above differential equations with $P_1(0) = 1$, we have

$$P_1(t) = \frac{1}{\lambda + \mu}(\mu e^{-(\lambda+\mu)t} + \lambda)$$

and

$$P_0(t) = \frac{1}{\lambda + \mu}(\mu - \mu e^{-(\lambda+\mu)t}).$$

Here $P_0(t)$ and $P_1(t)$ are called the *transient solutions*. We note that the steady-state probabilities are given by

$$\lim_{t\to\infty} P_0(t) = \frac{\mu}{\lambda+\mu}$$

and

$$\lim_{t\to\infty} P_1(t) = \frac{\lambda}{\lambda+\mu}.$$

In fact, the steady-state probability distribution can be obtained without solving the differential equations. We write the system of differential equations in matrix form:

$$\begin{pmatrix} \frac{dP_0(t)}{dt} \\ \frac{dP_1(t)}{dt} \end{pmatrix} = \begin{pmatrix} -\lambda & \mu \\ \lambda & -\mu \end{pmatrix} \begin{pmatrix} P_0(t) \\ P_1(t) \end{pmatrix}.$$

Since in the steady-state, $P_0(t) = p_0$ and $P_1(t) = p_1$ are constants and independent of t, we have

$$\frac{dp_0(t)}{dt} = \frac{dp_1(t)}{dt} = 0.$$

The steady-state probabilities will be the solution of the following linear system:

$$\begin{pmatrix} -\lambda & \mu \\ \lambda & -\mu \end{pmatrix} \begin{pmatrix} p_0 \\ p_1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

subject to $p_0 + p_1 = 1$.

In fact, we are often interested in obtaining the steady-state probability distribution of the Markov chain. This is because indicators of system performance such as the expected number of customers, and average waiting time can be written in terms of the steady-state probability distribution, see for instance [41–43, 46]. We will also apply the concept of steady-state probability distribution in the upcoming chapters. When the number of states is large, solving the steady-state probability distribution will be time consuming. Iterative methods are popular approaches for solving large scale Markov chain problems.

1.3 Iterative Methods for Solving Linear Systems

In this section, we introduce some classical iterative methods for solving large linear systems. For a more detailed introduction to iterative methods, we refer the reader to books by Bini et al. [18], Kincaid and Cheney [132], Golub and van Loan [108] and Saad [182].

1.3.1 Some Results on Matrix Theory

We begin our discussion with some useful results in matrix theory and their proofs can be also found in [108, 119, 132]. The first result is a useful formula for solving linear systems.

Proposition 1.38. *(Sherman-Morrison-Woodbury Formula) Let M be a non-singular $n \times n$ matrix,* $\mathbf{u}$ *and* $\mathbf{v}$ *be two $n \times l$ ($l \leq n$) matrices such that the matrix $(I_l + \mathbf{v}^T M\mathbf{u})$ is non-singular. Then we have*

$$\left(M + \mathbf{u}\mathbf{v}^T\right)^{-1} = M^{-1} - M^{-1}\mathbf{u}\left(I_l + \mathbf{v}^T M^{-1}\mathbf{u}\right)^{-1}\mathbf{v}^T M^{-1}.$$

Proof. We have

$$\begin{aligned}
&\left\{A^{-1} - A^{-1}\mathbf{u}\left(I_k + \mathbf{v}^T A^{-1}\mathbf{u}\right)^{-1}\mathbf{v}^T A^{-1}\right\}\left\{A + \mathbf{u}\mathbf{v}^T\right\}\\
&= I_n + A^{-1}\mathbf{u}\mathbf{v}^T - A^{-1}\mathbf{u}(I_k + \mathbf{v}^T A^{-1}\mathbf{u})^{-1}\mathbf{v}^T - A^{-1}\mathbf{u}(I_k + \mathbf{v}^T A^{-1}\mathbf{u})^{-1}\mathbf{v}^T A^{-1}\mathbf{u}\mathbf{v}^T\\
&= I_n + A^{-1}(\mathbf{u}\mathbf{v}^T) - A^{-1}\mathbf{u}(I_k + \mathbf{v}^T A^{-1}\mathbf{u})^{-1}(I_k + \mathbf{v}^T A^{-1}\mathbf{u})\mathbf{v}^T\\
&= I_n + A^{-1}(\mathbf{u}\mathbf{v}^T) - A^{-1}\mathbf{u}I_k\mathbf{v}^T\\
&= I_n.
\end{aligned}$$

Hence we proved the equality.

The second result is on the eigenvalue of a non-negative and irreducible square matrix.

Proposition 1.39. *(Perron-Frobenius Theorem) [15, 119] Let A be a non-negative and irreducible square matrix of order m. Then we have the following results:*

(i) A has a positive real eigenvalue λ which is equal to its spectral radius, i.e., $\lambda = \max_k |\lambda_k(A)|$ where $\lambda_k(A)$ denotes the k-th eigenvalue of A.
(ii) There corresponds an eigenvector $\mathbf{z}$ *with all its entries being real and positive, such that $A\mathbf{z} = \lambda\mathbf{z}$.*
(iii) λ is a simple eigenvalue of A.

The last result is on matrix norms. There are many matrix norms $||.||_M$ one can use. In the following, we introduce the definition of a matrix norm $||.||_{M_V}$ induced by a vector norm $||.||_V$.

Definition 1.40. Given a vector $||.||_V$ in $\mathbb{R}^n$, the matrix norm $||A||_{M_V}$ for an $n \times n$ matrix A induced by the vector norm is defined as

$$||A||_{M_V} = \sup\{||A\mathbf{x}||_V : \mathbf{x} \in \mathbb{R}^n \text{ and } ||\mathbf{x}||_V = 1\}.$$

In the following proposition, we introduce three popular matrix norms.

Proposition 1.41. *Let A be an $n \times n$ real matrix, then it can be shown that the matrix 1-norm, matrix ∞-norm and matrix 2-norm induced by $||.||_1$, $||.||_\infty$ and $||.||_2$ respectively by*

$$||A||_1 = \max_j \{\sum_{i=1}^{n} |A_{ij}|\},$$

$$||A||_\infty = \max_i \{\sum_{j=1}^{n} |A_{ij}|\}$$

and

$$||A||_2 = \sqrt{\lambda_{max}(A \cdot A^T)}.$$

Another popular matrix norm is the following Frobenius norm.

Definition 1.42. The Frobenius norm of a square matrix A is defined as

$$||A||_F = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij}^2}.$$

1.3.2 Splitting of a Matrix

We begin with the concept of splitting a matrix. If we are to solve

$$A\mathbf{x} = \begin{pmatrix} \frac{1}{2} & \frac{1}{3} & 0 \\ \frac{1}{3} & 1 & \frac{1}{3} \\ 0 & \frac{1}{3} & \frac{1}{2} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 5 \\ 10 \\ 5 \end{pmatrix} = \mathbf{b}.$$

There are many ways to split the matrix A into two parts and develop iterative methods for solving the linear system.

There are at least three different ways of splitting the matrix A:

$$\begin{aligned} A &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} \frac{-1}{2} & \frac{1}{3} & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & \frac{1}{3} & -\frac{1}{2} \end{pmatrix} \quad \text{(case 1)} \\ &= \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{2} \end{pmatrix} + \begin{pmatrix} 0 & \frac{1}{3} & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & \frac{1}{3} & 0 \end{pmatrix} \quad \text{(case 2)} \\ &= \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ \frac{1}{3} & 1 & 0 \\ 0 & \frac{1}{3} & \frac{1}{2} \end{pmatrix} + \begin{pmatrix} 0 & \frac{1}{3} & 0 \\ 0 & 0 & \frac{1}{3} \\ 0 & 0 & 0 \end{pmatrix} \quad \text{(case 3)} \\ &= S + (A - S) \end{aligned}$$

Now

$$A\mathbf{x} = (S + (A - S))\mathbf{x} = \mathbf{b}$$

and therefore

$$S\mathbf{x} + (A - S)\mathbf{x} = \mathbf{b}$$

Hence we may write

$$\mathbf{x} = S^{-1}\mathbf{b} - S^{-1}(A - S)\mathbf{x}$$

where we assume that S^{-1} exists. Then given an initial guess $\mathbf{x}^{(0)}$ of the solution of $A\mathbf{x} = \mathbf{b}$, one may consider the following iterative scheme:

$$\mathbf{x}^{(k+1)} = S^{-1}\mathbf{b} - S^{-1}(A - S)\mathbf{x}^{(k)}. \tag{1.7}$$

Clearly if $\mathbf{x}^{(k)} \to \mathbf{x}$ as $k \to \infty$ then we have $\mathbf{x} = A^{-1}\mathbf{b}$. We note that (1.7) converges if and only if there is a matrix norm $||.||_M$ such that

$$||S^{-1}(A - S)||_M < 1.$$

This is because for any square matrix B, we have

$$(I - B)(I + B + B^2 + \ldots + B^n) = I - B^{n+1}$$

and

$$\sum_{k=0}^{\infty} B^k = (I - B)^{-1} \quad \text{if} \quad \lim_{n\to\infty} B^n = 0.$$

If there exists a matrix norm $||.|_M$ such that $||B||_M < 1$ then

$$\lim_{n\to\infty} ||B^n||_M \le \lim_{n\to\infty} ||B||_M^n = 0$$

because $||AB||_M \le ||A||_M ||B||_M$ for any two matrices A and B. We then have

$$\lim_{n\to\infty} B^n = 0.$$

Therefore we have the following proposition.

Proposition 1.43. *If*

$$\|S^{-1}(A - S)\|_M < 1$$

then the iterative scheme converges to the solution of $A\mathbf{x} = \mathbf{b}$.

1.3.3 Classical Iterative Methods

Throughout this section, we let A be the matrix to be split and $\mathbf{b}$ be the right hand side vector. We use $\mathbf{x}^{(0)} = (0, 0, 0)^T$ as the initial guess.

Case 1: $S = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$.

$$\begin{aligned} \mathbf{x}^{(k+1)} &= \mathbf{b} - (A - I)\mathbf{x}^{(k)} \\ &= \begin{pmatrix} 5 \\ 10 \\ 5 \end{pmatrix} - \begin{pmatrix} -\frac{1}{2} & \frac{1}{3} & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & \frac{1}{3} & -\frac{1}{2} \end{pmatrix} \mathbf{x}^{(k)} \end{aligned}$$

$$\begin{aligned} \mathbf{x}^{(1)} &= (5\ 10\ 5)^T \\ \mathbf{x}^{(2)} &= (4.1667 \quad 6.6667 \quad 4.1667)^T \\ \mathbf{x}^{(3)} &= (4.8611 \quad 7.2222 \quad 4.8611)^T \\ \mathbf{x}^{(4)} &= (5.0231 \quad 6.7593 \quad 5.0231)^T \\ &\vdots \\ \mathbf{x}^{(30)} &= (5.9983 \quad 6.0014 \quad 5.9983)^T. \end{aligned}$$

When $S = I$, this is called the *Richardson method.*

Case 2: $S = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{2} \end{pmatrix}$

Therefore

$$\begin{aligned} \mathbf{x}^{(k+1)} &= S^{-1}\mathbf{b} - S^{-1}(A - S)\mathbf{x}^{(k)} \\ &= \begin{pmatrix} 10 \\ 10 \\ 10 \end{pmatrix} - \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{2} \end{pmatrix}^{-1} \begin{pmatrix} 0 & \frac{1}{3} & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & \frac{1}{3} & 0 \end{pmatrix} \mathbf{x}^{(k)} \\ &= (10\ 10\ 10)^T - \begin{pmatrix} 0 & \frac{2}{3} & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & \frac{2}{3} & 0 \end{pmatrix} \mathbf{x}^{(k)} \end{aligned}$$

$$\begin{aligned}
\mathbf{x}^{(1)} &= (10\ 10\ 10)^T \\
\mathbf{x}^{(2)} &= (3.3333 \quad 3.3333 \quad 3.3333)^T \\
\mathbf{x}^{(3)} &= (7.7778 \quad 7.7778 \quad 7.7778)^T \\
&\ \vdots \\
\mathbf{x}^{(30)} &= (6.0000 \quad 6.0000 \quad 6.0000)^T.
\end{aligned}$$

When $S = \text{Diag}(a_{11}, \cdots, a_{nn})$, this is called the *Jacobi method.*

Case 3: $S = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ \frac{1}{3} & 1 & 0 \\ 0 & \frac{1}{3} & \frac{1}{2} \end{pmatrix}$

$$\begin{aligned}
\mathbf{x}^{(k+1)} &= S^{-1}\mathbf{b} - S^{-1}(A - S)\mathbf{x}^{(k)} \\
&= \begin{pmatrix} 10 \\ \frac{20}{3} \\ \frac{50}{9} \end{pmatrix} - \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ \frac{1}{3} & 1 & 0 \\ 0 & \frac{1}{3} & \frac{1}{2} \end{pmatrix}^{-1} \begin{pmatrix} 0 & \frac{1}{3} & 0 \\ 0 & 0 & \frac{1}{3} \\ 0 & 0 & 0 \end{pmatrix} \mathbf{x}^{(k)}
\end{aligned}$$

$$\begin{aligned}
\mathbf{x}^{(1)} &= (10\ \frac{20}{3}\ \frac{50}{9})^T \\
\mathbf{x}^{(2)} &= (5.5556 \quad 6.2963 \quad 5.8025)^T \\
\mathbf{x}^{(3)} &= (5.8025 \quad 6.1317 \quad 5.9122)^T \\
\mathbf{x}^{(4)} &= (5.9122 \quad 6.0585 \quad 5.9610)^T \\
&\ \vdots \\
\mathbf{x}^{(14)} &= (6.0000 \quad 6.0000 \quad 6.0000)^T.
\end{aligned}$$

When S is the lower triangular part of the matrix A then this method is called the *Gauss-Seidel method.*

Proposition 1.44. *If A is diagonally dominant then*

$$||D^{-1}(A - D)||_\infty < 1$$

and the Jacobi method converges to the solution of $A\mathbf{x} = \mathbf{b}$.

1.3.4 Spectral Radius

Definition 1.45. Given an $n \times n$ square matrix A the spectral radius of A is defined as

$$\rho(A) = \max\{|\lambda| : \det(A - \lambda I) = 0\}.$$

Or in other words if $\lambda_1, \lambda_2, \cdots, \lambda_n$ are the eigenvalues of A then

$$\rho(A) = \max_i\{|\lambda_i|\}.$$

Example 1.46. Let

$$A = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

then the eigenvalues of A are $\pm i$ and $|i| = |-i| = 1$. Therefore $\rho(A) = 1$ in this case.

Proposition 1.47. *For any square matrix A, $\rho(A) = \inf_{\|\cdot\|_M} \|A\|_M$.*

Remark 1.48. If $\rho(A) < 1$ then there exists a matrix norm $||.||_M$ such that $||A||_M < 1$.

Using the remark, one can show the following proposition 1.49.

Proposition 1.49. *The iterative scheme*

$$x^{(k)} = G\mathbf{x}^{(k-1)} + \mathbf{c}$$

converges to

$$(I - G)^{-1}\mathbf{c}$$

for any starting vectors $\mathbf{x}^{(0)}$ and $\mathbf{c}$ if and only if $\rho(G) < 1$.

Proposition 1.50. *The iterative scheme*

$$\mathbf{x}^{(k+1)} = S^{-1}\mathbf{b} - S^{-1}(A - S)\mathbf{x}^{(k)} = (I - S^{-1}A)\mathbf{x}^{(k)} + S^{-1}\mathbf{b}$$

converges to $A^{-1}\mathbf{b}$ if and only if $\rho(I - S^{-1}A) < 1$.

Proof. The proof is complete by taking $G = I - S^{-1}A$ and $\mathbf{c} = S^{-1}\mathbf{b}$.

Definition 1.51. An $n \times n$ matrix B is said to be strictly diagonally dominant if

$$|B_{ii}| > \sum_{j=1, j\neq i}^{n} |B_{ij}| \quad \text{for} \quad i = 1, 2, \ldots, n$$

Proposition 1.52. *If A is strictly diagonally dominant then the Gauss-Seidel method converges for any starting* $\mathbf{x}^{(0)}$.

Proof. Let S be the lower triangular part of A. From Proposition 1.50 above, we only need to show

$$\rho(I - S^{-1}A) < 1.$$

Let λ be an eigenvalue of $(I - S^{-1}A)$ and $\mathbf{x}$ be its corresponding eigenvector such that $\|\mathbf{x}\|_\infty = 1$. We want to show $|\lambda| < 1$. We note that

$$(I - S^{-1}A)\mathbf{x} = \lambda\mathbf{x}$$

and therefore

$$\begin{pmatrix} 0 & -a_{12} & \cdots & -a_{1n} \\ \vdots & 0 & & \\ \vdots & & \ddots & -a_{n-1n} \\ 0 & \cdots & & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} a_{11} & 0 & \cdots & 0 \\ a_{21} & a_{22} & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ a_{n1} & \cdots & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} \lambda x_1 \\ \lambda x_2 \\ \vdots \\ \lambda x_n \end{pmatrix}.$$

Therefore we have

$$-\sum_{j=i+1}^{n} a_{ij}x_j = \lambda \sum_{j=1}^{i} a_{ij}x_j \quad \text{for } i = 1, \cdots, n-1.$$

Since $\|x\|_\infty = 1$, there exists i such that

$$|x_i| = 1 \geq |x_j|.$$

For this i we have

$$|\lambda||a_{ii}| = |\lambda a_{ii} x_i| \leq \sum_{j=i+1}^{n} |a_{ij}| + |\lambda| \sum_{j=1}^{i-1} |a_{ij}|$$

and therefore

$$|\lambda| \leq \sum_{j=i+1}^{n} |a_{ij}| \Bigg/ \left(|a_{ii}| - \sum_{j=1}^{i-1} |a_{ij}| \right) < 1$$

1.3.5 Successive Over-Relaxation (SOR) Method

In solving $A\mathbf{x} = \mathbf{b}$, one may split A as follows:

$$A = \underbrace{L + wD} + (1 - w)D + U$$

where L is the strictly lower triangular part, D is the diagonal part and U is the strictly upper triangular part.

Example 1.53.

$$\begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 2 \end{pmatrix} = \underbrace{\begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}}_{L} + w \underbrace{\begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix}}_{D} + (1-w) \underbrace{\begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix}}_{D} + \underbrace{\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}}_{U}$$

One may consider the iterative scheme with $S = L + wD$ as follows:

$$\mathbf{x}_{n+1} = S^{-1}\mathbf{b} + S^{-1}(S - A)\mathbf{x}_n = S^{-1}\mathbf{b} + (I - S^{-1}A)\mathbf{x}_n.$$

We remark that

$$I - S^{-1}A = I - (L + wD)^{-1}A.$$

Moreover, when $w = 1$, it is just the Gauss-Seidel method [128]. This method is called the *SOR method*. It is clear that the method converges if and only if the iteration matrix has a spectral radius less than one.

Proposition 1.54. *[108] The SOR method converges to the solution of*

$$A\mathbf{x} = \mathbf{b}$$

if and only if

$$\rho(I - (L + wD)^{-1}A) < 1.$$

1.3.6 Conjugate Gradient Method

Conjugate gradient (CG) methods are iterative methods for solving linear system of equations $A\mathbf{x} = \mathbf{b}$ where A is symmetric positive definite [8, 108]. This method was first discussed by Hestenes and Stiefel [117]. The motivation of the method is that it involves the process of minimizing a quadratic function such as

$$f(\mathbf{x}) = (A\mathbf{x} - \mathbf{b})^T(A\mathbf{x} - \mathbf{b}).$$

Here A is symmetric positive definite and this minimization usually takes place over a sequence of Krylov subspaces which are generated recursively by adding a new basic vector $A^k\mathbf{r}_0$ to those of the subspace V_{k-1} generated where

$$\mathbf{r_0} = A\mathbf{x_0} - \mathbf{b}$$

is the residue of the initial vector $\mathbf{x_0}$.

Usually, a sequence of conjugate orthogonal vectors is constructed from V_k so that CG methods would be more efficient. Computing these vectors can be done recursively which involves only a few vectors if A is self-adjoint with respect to the inner product. The CG methods are attractive since they can give the exact solution after at most n steps in exact arithmetic where n is the size of the matrix A. Hence it can also be regarded as a direct method in this sense. But in the presence of round-off errors and finite precision, the number of iterations may be greater than n. Thus, CG methods can be seen as least square methods where the minimization takes place on a particular vector subspace, the Krylov space. When estimating the error of the current solution in each step, a matrix-vector multiplication is then needed. The CG methods are popular and their convergence rates can be improved by using suitable preconditioning techniques. Moreover, it is parameter free, the recursions involved are usually short in each iteration and the memory requirements and the execution time are acceptable for many practical problems.

The CG algorithm reads:

```
Given an initial guess $\mathbf{x^0}$, A, $\mathbf{b}$, Max, tol:
$\mathbf{r^0} = \mathbf{b} - A\mathbf{x^0}$;
$\mathbf{v^0} = \mathbf{r^0}$;
For $k = 0$ to Max$-1$ do
If $||\mathbf{v^k}||_2 = 0$ then stop
$t_k =< \mathbf{r^k}, \mathbf{r^k} > / < \mathbf{v^k}, A\mathbf{v^k} >$;
$\mathbf{x^{k+1}} = \mathbf{x^k} + t_k \mathbf{v^k}$;
$\mathbf{r^{k+1}} = \mathbf{r^k} - t_k A\mathbf{v^k}$;
If $||\mathbf{r^{k+1}}, \mathbf{r^{k+1}}||_2 <$ tol then stop
$\mathbf{v^{k+1}} = \mathbf{r^{k+1}} + < \mathbf{r^{k+1}}, \mathbf{r^{k+1}} > / < \mathbf{r^k}, \mathbf{r^k} > \mathbf{v^k}$;
end;
output $\mathbf{x^{k+1}}$, $||\mathbf{r^{k+1}}||_2$.
```

Given a Hermitian, positive definite $n \times n$ matrix H_n, when the conjugate gradient method is applied to solving

$$H_n \mathbf{x} = \mathbf{b}$$

the convergence rate of this method depends on the spectrum of the matrix H_n, see also Golub and van Loan [108]. For example if the spectrum of H_n is contained in an interval, i.e. $\sigma(H_n) \subseteq [a, b]$, then the error in the i-th iteration is given by

$$\frac{||\mathbf{e}_i||}{||\mathbf{e}_0||} \leq 2 \left(\frac{\sqrt{b} - \sqrt{a}}{\sqrt{b} + \sqrt{a}} \right)^i ,$$

i.e. the convergence rate is linear. Hence the approximate upper bound for the number of iterations required to make the relative error

$$\frac{||\mathbf{e_i}||}{||\mathbf{e_0}||} \leq \delta$$

is given by

$$\frac{1}{2}\left(\sqrt{\frac{b}{a}}-1\right)\log\left(\frac{2}{\delta}\right)+1.$$

Very often the CG method is used with a matrix called preconditioner to accelerate its convergence rate. A good preconditioner C should satisfy the following conditions:

(i) The matrix C can be constructed easily
(ii) Given a right hand side vector $\mathbf{r}$, the linear system $C\mathbf{y} = \mathbf{r}$ can be solved efficiently
(iii) The spectrum (or singular values) of the preconditioned system $C^{-1}A$ should be clustered around one.

In the Preconditioned Conjugate Gradient (PCG) method, we solve the linear system

$$C^{-1}A\mathbf{x} = C^{-1}\mathbf{b}$$

instead of the original linear system

$$A\mathbf{x} = \mathbf{b}.$$

We expect the fast convergence rate of the PCG method can compensate for the extra cost of solving the preconditioner system $C\mathbf{y} = \mathbf{r}$ in each iteration step of the PCG method.

Apart from the point of view of condition number, condition (iii) is also very commonly used in proving convergence rates. In the following we give the definition of clustering.

Definition 1.55. We say that a sequence of matrices S_n of size n has a clustered spectrum around one, if for all $\epsilon > 0$ there exist non-negative integers n_0 and n_1, such that for all $n > n_0$ at most n_1 eigenvalues of the matrix $S_n^* S_n - I_n$ have absolute values larger than ϵ.

One sufficient condition for the matrix to have eigenvalues clustered around one is that $H_n = I_n + L_n$, where I_n is the $n \times n$ identity matrix and L_n is a low rank matrix (rank(L_n) is bounded above and independent of the matrix size n).

1.3.6.1 Conjugate Gradient Squared Method

Given a real symmetric, positive definite matrix A of size $n \times n$, the CG method can be used to solve the linear system $A\mathbf{x} = \mathbf{b}$. But in general a non-singular matrix can be neither symmetric nor positive definite. This is true for the applications in queueing systems and re-manufacturing systems in Chaps. 2 and 3. In this case, one may consider the normal equation of the original system, i.e.,

$$A^T A\mathbf{x} = A^T\mathbf{b}.$$

Here $A^T A$ is real symmetric and positive definite so that the CG method could be applied, but the condition number would then be squared. Moreover, it also involves the matrix-vector multiplication of the form $A^T\mathbf{r}$. These will increase the computational cost. Thus in our context, we propose to employ a generalized CG algorithm, namely the Conjugate Gradient Squared (CGS) method [190]. This method does not involve the matrix-vector multiplication of the form $A^T\mathbf{r}$.

The CGS algorithm reads:

```
Given an initial guess $\mathbf{x^0}$, A, $\mathbf{b}$, tol:
$\mathbf{x} = \mathbf{x_0}$;
$\mathbf{r} = \mathbf{b} - A\mathbf{x}$;
$\mathbf{r}' = \mathbf{s} = \mathbf{p} = \mathbf{r}$;
$\mathbf{w} = A\mathbf{p}$;
$\mu = \mathbf{r}'^T\mathbf{r}$;
repeat until $\mu <$ tol;
$\gamma = \mu$;
$\alpha = \gamma/\mathbf{r}'^t\mathbf{r}$;
$\mathbf{q} = \mathbf{s} - \alpha\mathbf{w}$;
$\mathbf{d} = \mathbf{s} + \mathbf{q}$;
$\mathbf{w} = A\mathbf{d}$;
$\mathbf{x} = \mathbf{x} + \alpha\mathbf{d}$;
$\mathbf{r} = \mathbf{r} - \alpha\mathbf{w}$;
otherwise
$\mu = \mathbf{r}'^T\mathbf{r}$;
$\beta = \mu/\gamma$;
$\mathbf{s} = \mathbf{r} - \beta\mathbf{q}$;
$\mathbf{p} = \mathbf{s} + \beta(\mathbf{q} + \beta\mathbf{p})$;
end;
```

1.3.7 Toeplitz Matrices

We end this subsection by introducing a class of matrices, namely Toeplitz matrices. A Toeplitz matrix T is a matrix having constant diagonals, i.e.

$$T = \begin{pmatrix} t_0 & t_1 & t_2 & \cdots & t_{n-1} & t_n \\ t_{-1} & t_0 & t_1 & \cdots & \cdots & t_{n-1} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & t_2 \\ t_{-n+1} & \cdots & \cdots & \ddots & \ddots & t_1 \\ t_{-n} & t_{-n+1} & \cdots & \cdots & t_{-1} & t_0 \end{pmatrix}.$$

Toeplitz matrices and near-Toeplitz matrices have many applications in applied sciences and engineering such as the multi-channel least squares filtering in time series [172] and for signal and image processing problems [149]. A survey on the applications of Toeplitz systems can be found in Chan and Ng [39]. Applications in solving queueing systems and re-manufacturing systems will be discussed in the Chaps. 2 and 3.

In the above applications, solving a Toeplitz or near-Toeplitz system is the focus. Direct methods for solving Toeplitz systems based on the recursion formula are commonly used, see for instance Trench [196]. For an $n \times n$ Toeplitz matrix T, these direct methods require $O(n^2)$ operations. Faster algorithms that require $O(n \log^2 n)$ operations have also been developed when the Toeplitz matrix is symmetric and positive definite.

An important subset of Toeplitz matrices is the class of circulant matrices. A circulant $n \times n$ matrix C is a Toeplitz matrix such that each column is a cyclic shift of the previous one, i.e.

$$C = \begin{pmatrix} c_0 & c_1 & \cdots & c_{n-1} & c_n \\ c_n & c_0 & c_1 & \cdots & c_{n-1} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ c_2 & \vdots & \ddots & \ddots & c_1 \\ c_1 & c_2 & \cdots & c_n & c_0 \end{pmatrix}. \tag{1.8}$$

Very often circulant matrices are used to approximate Toeplitz matrices in preconditioning or finding an approximate solution. This is because circulant matrices have the nice property that they can be diagonalized by the discrete Fourier matrix F. More precisely,

$$FCF^* = D = \mathrm{Diag}(d_0, d_1, \ldots, d_n)$$

where F is the discrete Fourier matrix with entries given by

$$F_{j,k} = \frac{1}{\sqrt{n}} e^{-\frac{(2jk\pi)i}{n}}, \quad j,k = 0,1,\cdots,n-1,$$

and D is a diagonal matrix with elements being the eigenvalues of C, see for instance [79]. Here F^* is the conjugate transpose of F. The matrix-vector multiplication $F\mathbf{y}$ is called the Fast Fourier Transformation (FFT) of the column vector $\mathbf{y}$ and can be done in $O(n \log n)$ operations. Consider for a unit vector

$$\mathbf{e}_1 = (1,0,\ldots,0)^T,$$

we have

$$C\mathbf{e}_1 = (c_0, c_n, \ldots, c_1)^T \quad \text{and} \quad F\mathbf{e}_1 = \frac{1}{\sqrt{n}}(1,1,\ldots,1)^T$$

because the first column of F is a column vector with all entries being equal. Therefore

$$F(c_0, c_n, \ldots, c_1)^T = FC\mathbf{e}_1 = DF\mathbf{e}_1 = \frac{1}{\sqrt{n}}(d_0, d_1, \ldots, d_n)^T$$

and hence the eigenvectors of a circulant matrix C can be obtained by using the FFT in $O(n \log n)$ operations. Moreover, the solution of a circulant linear system can also be obtained in $O(n \log n)$ operations.

The FFT can be used in the Toeplitz matrix-vector multiplication. A Toeplitz matrix can be embedded in a circulant matrix as follows:

$$\tilde{C}(\mathbf{y}, \mathbf{0})^T \equiv \begin{pmatrix} T & S_1 \\ S_2 & T \end{pmatrix} \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{r} \\ \mathbf{b} \end{pmatrix}. \tag{1.9}$$

Here matrices S_1 and S_2 are such that $\tilde{C}$ is a circulant matrix. Then FFT can be applied to obtain $\mathbf{r} = T\mathbf{y}$ in $O(n \log n)$ operations.

1.4 Hidden Markov Models

Hidden Markov Models (HMMs) are widely used in bioinformatics [135], speech recognition [175] and many other areas [155]. In a HMM there are two types of states: the observable states and the hidden states. In a HMM there is no one-to-one correspondence between the hidden states and the observed symbols. It is therefore no longer possible to tell what hidden state the model is just by looking at the observation symbol generated. A HMM is usually characterized by the following elements [175]:

- N, the number of hidden states in the model. Although the states are hidden, for many practical applications there is often some physical significance to the states. We denote the individual states as

  $$S = \{s_1, s_2, \cdots, s_N\},$$

  and the state at the length t as Q_t.
- M, the number of distinct observation symbols per hidden state. The observation symbols correspond to the physical output of the system being modeled. For instance, "conforming product" and "non-conforming product" are two observation symbols in a production process. We denote the individual symbols as

  $$V = \{v_1, v_2, \cdots, v_M\}$$

  and the symbol at the length t as O_t.
- The state transition probability distribution $[A]_{ij} = \{a_{ij}\}$ where

  $$a_{ij} = P(Q_{t+1} = s_i | Q_t = s_j), \quad 1 \leq i, j \leq N.$$

- The observation symbol probability distribution in hidden state j, $[B]_{jk} = \{b_j(v_k)\}$, where

  $$b_j(v_k) = P(O_t = v_k | Q_t = s_j), \quad 1 \leq j \leq N, \quad 1 \leq k \leq M.$$

- The initial state distribution $\Pi = \{\pi_i\}$ where

  $$\pi_i = P(Q_1 = s_i), \quad 1 \leq i \leq N.$$

Given appropriate values of N, M, A, B and Π, the HMM can be used as a generator to give an observation sequence

$$O = \{O_1 O_2 O_3 \cdots O_T\}$$

where T is the number of observations in the sequence. For simplicity, we use the compact notation

$$\Lambda = (A, B, \Pi)$$

to indicate the complete parameter set of the HMM. According to the above specification, the first order transition probability distribution among the hidden states is used. There are three key problems in HMMs:

- **Problem 1:**
  Given the observation sequence $O = \{O_1 O_2 \cdots O_T\}$ and a HMM, how to efficiently compute the probability of the observation sequence?
- **Problem 2:**
  Given the observation sequence $O = \{O_1 O_2 \cdots O_T\}$ and a HMM, how to choose a corresponding state sequence $Q = \{Q_1 Q_2 \cdots Q_T\}$ which is optimal in a certain context?
- **Problem 3:**
  Given the observation sequence $O = \{O_1 O_2 \cdots O_T\}$, how to choose the model parameters in a HMM?

For **Problem 1**, a forward-backward dynamic programming procedure [11] has been formulated to calculate the probability of the observation sequence efficiently.

For **Problem 2**, we attempt to uncover the hidden part of the model, i.e., to find the "correct" state sequence. In many practical situations, we use an optimality criteria to solve the problem as best as possible. The most widely used criterion is to find a single best state sequence, i.e., maximize the likelihood $P(Q|\Lambda, O)$. This is equivalent to maximizing $P(Q, O|\Lambda)$ since

$$P(Q|\Lambda, O) = \frac{P(Q, O|\Lambda)}{P(O|\Lambda)}.$$

Viterbi algorithm [204] is a dynamic programming technique for finding this single best state sequence

$$Q = \{Q_1, Q_2, \cdots, Q_T\}$$

for the given observation sequence

$$O = \{O_1, O_2, \cdots, O_T\}.$$

For **Problem 3**, we attempt to adjust the model parameters Λ such that $P(O|\Lambda)$ is maximized by using the Expectation-Maximization (EM) algorithm. For a complete tutorial on HMMs, we refer readers to the paper by Rabiner [175] and the book by MacDonald and Zucchini [155].

1.5 Markov Decision Process

Markov Decision Process (MDP) has been successfully applied in equipment maintenance, inventory control and many other areas in management science [3, 208]. In this section, we will briefly introduce the MDP, but interested readers can also consult the books by Altman [3], Puterman [173] and White [207].

Similar to the case of a Markov chain, the MDP is a system that can move from one distinguished state to any other possible states. In each step, the decision maker has to take action on a well-defined set of alternatives. This action affects the transition probabilities of the next move and incurs an immediate gain (or loss) and subsequent gain (or loss). The problem that the decision maker faces is to determine a sequence of actions maximizing the overall gain. The process of MDP is summarized as follows:

(i) At time t, a certain state i of the Markov chain is observed.
(ii) After the observation of the state, an action, let us say k, is taken from a set of possible decisions A_i. Different states may have different sets of possible actions.
(iii) An immediate gain (or loss) $q_i^{(k)}$ is then incurred according to the current state i and the action k taken.
(iv) The transition probabilities $p_{ji}^{(k)}$ are then affected by the action k.
(v) When the time parameter t increases, transition occurs again and the above steps (i)–(iv) repeat.

A policy D is a rule of taking action. It prescribes all the decisions that should be made throughout the process. Given the current state i, the value of an optimal policy $v_i(t)$ is defined as the total expected gain obtained with t decisions or transitions remaining. For the case of one-period remaining, i.e. $t = 1$, the value of an optimal policy is given by

$$v_i(1) = \max_{k \in A_i} \{q_i^{(k)}\}. \tag{1.10}$$

Since there is only one-period remained, an action maximizing the immediate gain will be taken. For the case of two periods remaining, we have

$$v_i(2) = \max_{k \in A_i} \left\{ q_i^{(k)} + \underbrace{\alpha \sum_j p_{ji}^{(k)} v_j(1)}_{\text{subsequent gain}} \right\} \tag{1.11}$$

where α is the discount factor. Since the subsequent gain is associated with the transition probabilities which are affected by the action taken, an optimal policy should consider both the immediate and subsequent gain. The model can be easily extended to a more general situation, the process having n transitions remaining.

$$v_i(n) = \max_{k \in A_i} \left\{ q_i^{(k)} + \underbrace{\alpha \sum_j p_{ji}^{(k)} v_j(n-1)}_{\text{subsequent gain}} \right\}. \tag{1.12}$$

Table 1.1 A summary of the policy parameters

State i	Alternative k	$q_i^{(k)}$	$p_{i1}^{(k)}$	$p_{i2}^{(k)}$
1 (high volume)	1. (No action)	8	0.4	0.6
	2. (Regular Maintenance)	7	0.8	0.2
	3. (Fully Upgrade)	5	1	0
2 (low volume)	1. (No action)	4	0.1	0.9
	2. (Regular Maintenance)	3	0.4	0.6
	3. (Fully Upgrade)	1	0.8	0.2

From the above equation, the subsequent gain of $v_i(n)$ is defined as the expected value of $v_j(n-1)$. Since the number of transitions remaining is countable or finite, the process is called the discounted finite horizon MDP. For the infinite horizon MDP, the value of an optimal policy can be expressed as

$$v_i = \max_{k \in A_i} \left\{ q_i^{(k)} + \alpha \sum_j p_{ji}^{(k)} v_j \right\}. \tag{1.13}$$

The finite horizon MDP is a dynamic programming problem and the infinite horizon MDP can be transformed into a linear programming problem. Both of them can be solved easily by using an EXCEL spreadsheet.

Example 1.56. We consider an on-line game company that plans to stay in business for 4 more years and then it will be closed without any salvage value. Each year, the volume of players only depends on the volume in the last year, and it is classified as either high or low. If a high volume of players occurs, the expected profit for the company will be 8 million dollars; but the profit drops to 4 million dollars when a low volume of players is encountered. At the end of every year, the profit of this year is collected, and then the company has the option to take certain actions that influence the performance of their service and hence the volume of players in the future may be altered. But some of these actions are costly so they reduce instant profit. To be more specific, the company can choose to: take no action, which costs nothing; perform only regular maintenance to the service system, which costs 1 million; or fully upgrade the service system, which costs 3 million. When the volume of players in the last year was high, it stays in the high state in the coming year with probability 0.4 if no action is taken; this probability is 0.8 if only regular maintenance is performed; and the probability rises to 1 if the system is fully upgraded. When the volume of players in the last year was low, then the probability that the player volume stays low is 0.9 with no action taken, 0.6 with regular maintenance, and 0.2 when the service system is fully upgraded. Assume the discount factor is 0.9 and that the company experienced a low volume of players last year. Determine the optimal (profit maximizing) strategy for the company. The parameters of this problem can be summarized in Table 1.1.

By using the MDP approach, we can compute the following:

$$v_i(n) = \max_{k \in A_i} \left\{ q_i^{(k)} + \alpha \sum_j p_{ji}^{(k)} v_j(n-1) \right\}, \tag{1.14}$$

and have

$$v_1(4) = \max \left\{ \begin{array}{c} 8 + 0.9 \times [0.4 \times v_1(3) + 0.6 \times v_2(3)] \\ 7 + 0.9 \times [0.8 \times v_1(3) + 0.2 \times v_2(3)] \\ 5 + 0.9 \times [1 \times v_1(3)] \end{array} \right\};$$

$$v_2(4) = \max \left\{ \begin{array}{c} 4 + 0.9 \times [0.1 \times v_1(3) + 0.9 \times v_2(3)] \\ 3 + 0.9 \times [0.4 \times v_1(3) + 0.6 \times v_2(3)] \\ 1 + 0.9 \times [0.8 \times v_1(3) + 0.2 \times v_2(3)] \end{array} \right\};$$

$$v_1(3) = \max \left\{ \begin{array}{c} 8 + 0.9 \times [0.4 \times v_1(2) + 0.6 \times v_2(2)] \\ 7 + 0.9 \times [0.8 \times v_1(2) + 0.2 \times v_2(2)] \\ 5 + 0.9 \times [1 \times v_1(2)] \end{array} \right\};$$

$$v_2(3) = \max \left\{ \begin{array}{c} 4 + 0.9 \times [0.1 \times v_1(2) + 0.9 \times v_2(2)] \\ 3 + 0.9 \times [0.4 \times v_1(2) + 0.6 \times v_2(2)] \\ 1 + 0.9 \times [0.8 \times v_1(2) + 0.2 \times v_2(2)] \end{array} \right\};$$

$$v_1(2) = \max \left\{ \begin{array}{c} 8 + 0.9 \times [0.4 \times v_1(1) + 0.6 \times v_2(1)] \\ 7 + 0.9 \times [0.8 \times v_1(1) + 0.2 \times v_2(1)] \\ 5 + 0.9 \times [1 \times v_1(1)] \end{array} \right\};$$

$$v_2(2) = \max \left\{ \begin{array}{c} 4 + 0.9 \times [0.1 \times v_1(1) + 0.9 \times v_2(1)] \\ 3 + 0.9 \times [0.4 \times v_1(1) + 0.6 \times v_2(1)] \\ 1 + 0.9 \times [0.8 \times v_1(1) + 0.2 \times v_2(1)] \end{array} \right\};$$

$$\left\{ \begin{array}{ll} v_1(1) = \max\{8, 7, 5\} = 8, & p_1(1) = 1; \\ v_2(1) = \max\{4, 3, 1\} = 4, & p_2(1) = 1. \end{array} \right.$$

With the results from the last equations, we can solve for other values by backward substitution.

Let $p_i(n) = k^*$ such that

$$\max_{k \in A_i} \left\{ q_i^{(k)} + \alpha \sum_j p_{ji}^{(k)} v_j(n-1) \right\} = q_i^{(k^*)} + \alpha \sum_j p_{ji}^{(k^*)} v_j(n-1)$$

then $p_i(n)$ actually keeps track of the optimal policy for every single period. We can summarize all results from the calculations in Table 1.2.

Table 1.2 A summary of results

n	1	2	3	4
$v_1(n)$	8	13.48	18.15	22.27
$v_2(n)$	4	8.04	12.19	16.27
$p_1(n)$	1	2	2	2
$p_2(n)$	1	2	2	3

Table 1.3 A summary of results

n	1	2	3	4
$v_1(n)$	8	11.36	13.25	14.35
$v_2(n)$	4	6.64	8.27	9.26
$p_1(n)$	1	1	2	2
$p_2(n)$	1	1	1	1

Since the on-line gaming company started from having a low volume of players (State 2), the optimal policy for the company is as follows: with 4 more years left, choose Alternative 3 (fully upgrade); then use Alternative 2 (regular maintenance) for two consecutive years; and finally, use Alternative 1 (no action) when there is only 1 year left.

Note that the optimal policy may vary depending on the value of the discount factor. For instance, if in this example, we have a discount factor of 0.6, then we have different results as summarized in Table 1.3. If the company starts with a low volume of players, the optimal policy is to stay with Alternative 1 (no action). We leave it as an exercise for the reader to device the results themselves.

1.5.1 Stationary Policy

A stationary policy is a policy where the decision depends only on the state the system is in and is independent of n. For instance, a stationary policy D prescribes the action $D(i)$ when the current state is i. Define $\bar{D}$ as the associated one-step-removed policy, then the value of policy $w_i(D)$ is defined as

$$w_i(D) = q_i^{D(i)} + \alpha \sum_j p_{ji}^{D(i)} w_j(\bar{D}). \tag{1.15}$$

Given a Markov decision process with an infinite horizon and a discount factor α, $0 < \alpha < 1$, choose, for each i, an alternative k_i such that

$$\max_{k \in A_i} \left\{ q_i^{(k)} + \alpha \sum_j p_{ji}^{(k)} v_j \right\} = q_i^{(k_i)} + \alpha \sum_j p_{ji}^{(k_i)} v_j .$$

Define the stationary policy D by $D(i) = k_i$. Then for each i, $w_i(D) = v_i$, i.e. the stationary policy is an optimal policy.

Table 1.4 A summary of results

Policy D		Values	
$D(1)$	$D(2)$	$w_1(D)$	$w_2(D)$
1	1	50.41	44.93
1	2	53.00	48.00
1	3	52.21	47.06
2	1	55.41	47.30
2	2	58.75	52.50
2	3	59.20	53.20
3	1	50.00	44.74
3	2	50.00	45.65
3	3	50.00	45.12

Example 1.57. Determine the optimal policy and the values for the Markov decision process in Example 1.56, assuming the process has an infinite horizon and the discount factor remains equal to 0.9.

For a stationary policy D (with $D(1) = k_1$, $D(2) = k_2$), since we have

$$\begin{cases} w_1(D) = q_1^{(k_1)} + \alpha[p_{11}^{(k_1)} w_1(D) + p_{21}^{(k_2)} w_2(D)] \\ w_2(D) = q_2^{(k_2)} + \alpha[p_{12}^{(k_1)} w_1(D) + p_{11}^{(k_2)} w_2(D)], \end{cases}$$

hence $[w_1(D)\ ;\ w_2(D)]$ can be solved for every stationary policy. Results are summarized in the following table.

From the above, the optimal values are $v_1 = w_1(D) = 59.2$, $v_2 = w_2(D) = 53.2$; the optimal stationary policy is to choose Alternative 2 in state 1 and to choose Alternative 3 in state 2 (Table 1.4).

1.6 Exercises

1. Prove Proposition 1.24.
   Hint: Let $X^{(n)}$ be the state of the process after making n transitions and define

$$I_n = \begin{cases} 1, \text{ if } X^{(n)} = i \\ 0, \text{ if } X^{(n)} \neq i. \end{cases}$$

   Then show that the expected number of times the process will visit state i, given that it begins in State i is given by:

$$E\left(\sum_{n=0}^{\infty} I_n | X^{(0)} = i\right) = \sum_{n=0}^{\infty} P_{ii}^{(n)}.$$

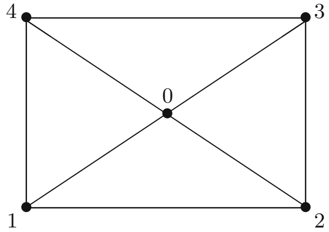


Fig. 1.6 The random walk on a square

2. Prove Proposition 1.25.
Hint: Since i communicates with j, there exist integers k and m such that $P_{ji}^{(k)} > 0$ and $P_{ij}^{(m)} > 0$. We have also the following inequality:
$$P_{jj}^{(m+n+k)} \geq P_{ij}^{(m)} \cdot P_{ii}^{(n)} \cdot P_{ji}^{(k)}.$$
Then apply Proposition 1.24 to get the conclusion.
3. Consider a random walk on a square with its centroid as shown in Fig. 1.6. Suppose that at each State $i\,(i = 0, 1, 2, 3, 4)$, the transition probabilities to other adjacent states are all equal. While the probability of staying at the same state in the next transition is assumed to be zero.

   (a) Show that the Markov chain of the random walk is irreducible and all the states are recurrent.
   (b) Find the state-state probability distribution
   $$\boldsymbol{\pi} = (\pi_0, \pi_1, \pi_2, \pi_3, \pi_4)$$
   of the Markov chain where
   $$\boldsymbol{\pi} P = \boldsymbol{\pi} \quad \text{and} \quad \sum_{i=0}^{4} \pi_i = 1 \quad \text{and} \quad \pi_i \geq 0 \quad \text{for } i = 0, 1, \ldots, 4.$$

4. Show that the transition probability matrix
$$P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$
is irreducible and has a unique stationary distribution but it has no steady-state probability distribution.
5. Given the transition probability matrix P of a Markov chain as follows:
$$P = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0.5 & 0 & 0 & 0.5 \\ 0 & 0.5 & 0 & 0.5 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$
Classify the states of the Markov chain (transient or recurrent).

6. Construct a Markov chain of four states such that its steady-state probability distribution is given by
$$(0.1, 0.2, 0.3, 0.4).$$
7. Suppose $y_1, y_2, \ldots, y_n$ are n independent Poisson random variables having the same mean 1. Let
$$z_n = y_1 + y_2 + \ldots + y_n.$$
Then we note that the sum of the n Poisson random variables is also a Poisson random variable with mean n and variance n. Furthermore, for large n, we have
$$\frac{z_n - n}{\sqrt{n}} \sim N(0, 1).$$
Argue that
$$\frac{e^{-n}n^n}{n!} = P(z_n = n) \approx \int_{\frac{-1}{\sqrt{n}}}^{0} \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}} dx \approx \frac{1}{\sqrt{2\pi n}}$$
and hence deduce the Stirling's formula.
8. Prove Proposition 1.33.
9. Prove that the following are vector norms defined on $\mathbb{R}^n$:
$$||\mathbf{v}||_1 = \sum_{i=1}^{n} |v_i|, \quad ||\mathbf{v}||_\infty = \max_i\{|v_i|\} \quad \text{and} \quad ||\mathbf{v}||_2 = \sqrt{\sum_{i=1}^{n} |v_i|^2}.$$
10. (a) Let $p > 1$ and define q such that $\frac{1}{p} + \frac{1}{q} = 1$. Then for any non-negative a and b, we have
$$a^{\frac{1}{p}} b^{\frac{1}{q}} \leq \frac{a}{p} + \frac{b}{q}.$$
(b) Let $p > 1$ and q be defined such that
$$\frac{1}{p} + \frac{1}{q} = 1.$$
Show that for two vectors $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{y} = (y_1, \ldots, y_n)$, we have
$$\sum_{i=1}^{n} |x_i y_i| \leq \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}} \left(\sum_{i=1}^{n} |y_i|^q\right)^{\frac{1}{q}}.$$
Hint: For part (a), you may wish to consider the function
$$f(x) = x^{\frac{1}{p}} - 1 - \frac{1}{p}(x - 1).$$

For part (b) Let

$$A = \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}, \quad B = \left(\sum_{i=1}^{n} |y_i|^q\right)^{\frac{1}{q}}, \quad a_i = \frac{|x_i|^p}{A^p}, \quad b_i = \frac{|y_i|^p}{B^p}$$

and apply part (a).

11. Show that for $\mathbf{v} \in \mathbb{R}^n$

$$||\mathbf{v}||_\infty = \lim_{p\to\infty} \left(\sum_{i=1}^{n} |v_i|^p\right)^{\frac{1}{p}}.$$

12. Show that for any two $n \times n$ matrices A and B, we have

$$||AB||_M \leq ||A||_M ||B||_M.$$

13. Show that for a square matrix A we have

$$||A||_{M_2} \leq \sqrt{||A||_{M_1} \cdot ||A||_{M_\infty}}.$$

14. Customers request service from a group of m servers according to a Poisson process with mean inter-arrival time λ^{-1}. Suppose the service times of the servers are mutually independent and exponentially distributed with the same mean μ^{-1}. At time zero, you find all m servers occupied and no customers waiting. Find the probability that exactly k additional customers request service from the system before the first completion of a service request.
15. Prove Proposition 1.41.
16. Prove or disprove that $\rho(A)$ is a matrix norm for any $n \times n$ square matrix A. Hint: Consider the matrix

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

17. Consider solving the linear equations $A\mathbf{x} = \mathbf{b}$ by the Jacobi method where

$$A = \begin{pmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -1 & 2 & -1 \\ 0 & \cdots & 0 & -1 & 2 \end{pmatrix} \equiv 2I_n + (A - 2I_n) \tag{1.16}$$

where I_n is the $n \times n$ identity matrix. The iterative scheme reads

$$\mathbf{x}_{n+1} = (I_n - \frac{1}{2}A)\mathbf{x}_n + \frac{1}{2}\mathbf{b}.$$

We note that A is not diagonally dominant. Prove that the Jacobi method converges by showing that

$$||I_n - \frac{1}{2}A||_{M_2} < 1.$$

Hint: Show that for $||x||_2^2 = 1$ we have

$$\left|\mathbf{x}^T(I_n - \frac{1}{2}A)\mathbf{x}\right| < 1.$$

18. Suppose A is a non-singular $n \times n$ matrix but A is neither symmetric nor positive definite, explain how the conjugate gradient method can be applied to solve for the solution of
$$A\mathbf{x} = (1, 0 \ldots, 0)^T.$$
19. (a) Implement the conjugate gradient method in MATLAB (or other software like Octave) to solve
$$A\mathbf{x} = (1, 1, \ldots, 1)^T.$$
The zero vector is employed as the initial guess and tolerance level is 10^{-6}. Here A is the matrix in (1.16).
    (b) For $n = 10, 20, 40, 80, 160, 320$ (n is the size of the matrix A) record the number of iterations for convergence.
20. In Example 1.56, show that if the discount factor is 0.6 and we start from state 2, then the optimal policy becomes: always stick to Alternative 1 (no action).

Chapter 2
Queueing Systems and the Web

In this chapter, we first discuss some more Markovian queueing systems. The queueing system is a classical application of continuous Markov chains. We then present an important numerical algorithm based on the computation of Markov chains for ranking webpages. This is a modern application of Markov chains though the numerical methods used are classical.

2.1 Markovian Queueing Systems

An important class of queueing networks is the Markovian queueing systems. The main assumptions of a Markovian queueing system are the Poisson arrival process and exponential service time. The one-server system discussed in Sect. 1.2.1 of Chap. 1 is a queueing system without waiting space. This means that when a customer arrives and finds the server is busy, the customer has to leave the system. In the following sections, we will introduce some more Markovian queueing systems. A queueing system is a classical application of a continuous time Markov chain. We will further discuss its applications in re-manufacturing systems in Chap. 3. For more details about numerical solutions for queueing systems and Markov chains, we refer the readers to the books by Ching [46], Leonard [144], Neuts [167, 168] and Stewart [191].

In the following, we begin our discussion with an M/M/1/$n-2$ queue, a Markovian queueing system with one server and $n-2$ waiting spaces. Here the first 'M' representing the arrival process is a Poisson process and the second 'M' represents the service time following the exponential distribution.

W.-K. Ching et al., *Markov Chains*, International Series in Operations Research & Management Science 189, DOI 10.1007/978-1-4614-6312-2_2,

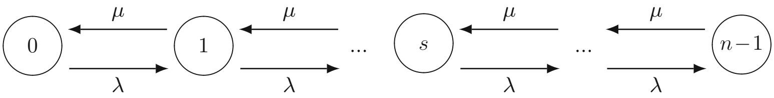


Fig. 2.1 The Markov chain for the one-queue system (one server)

2.1.1 An M/M/1/n − 2 Queueing System

Now let us consider a more general queueing system with customer arrival rate being λ. Suppose the system has one exponential server with service rate being μ and there are $n-2$ waiting spaces in the system. The queueing discipline is *First-come-first-serve*. When a newly arrived customer finds the server is busy, the customer can still wait in the queue provided that there is a waiting space available. Otherwise, the customer has to leave the queueing system. To describe the queueing system, we use the number of customers in the queue to represent the state of the system. There are n states, namely $0, 1, \ldots, n-1$. The Markov chain for the queueing system is given in Fig. 2.1. The number of customers in the system is used to represent the states in the Markov chain. Clearly it is an irreducible Markov chain.

If we order the states of the system in increasing number of customers, it is not difficult to show that the generator matrix for this queueing system is given by the following $n \times n$ tri-diagonal matrix $A_1 = A_{(n,1,\lambda,\mu)}$ where

$$A_1 = \begin{pmatrix} \lambda & -\mu & & & & & & 0 \\ -\lambda & \lambda+\mu & -\mu & & & & & \\ & \ddots & \ddots & \ddots & & & & \\ & & -\lambda & \lambda+\mu & -\mu & & & \\ & & & -\lambda & \lambda+\mu & -\mu & & \\ & & & & \ddots & \ddots & \ddots & \\ & & & & & -\lambda & \lambda+\mu & -\mu \\ 0 & & & & & & -\lambda & s\mu \end{pmatrix} \qquad (2.1)$$

and transient solution $\mathbf{p}(t)$ satisfies the following system of differential equations:

$$\frac{d\mathbf{p}(t)}{dt} = A_1\mathbf{p}(t).$$

The underlying Markov chain is irreducible and the solution for the steady-state probability distribution, i.e.,

$$\lim_{t\to\infty} \mathbf{p}(t) = \mathbf{p}_{(n,1,\lambda,\mu)} \quad \text{and} \quad \mathbf{0} = \lim_{t\to\infty} \frac{d\mathbf{p}(t)}{dt} = A_1\mathbf{p}_{(n,1,\lambda,\mu)}$$

can be shown to be $\mathbf{p}_{(n,1,\lambda,\mu)} = (p_0, p_1, \ldots, p_{n-1})^T$ where

$$p_i = \alpha \prod_{k=1}^{i+1} \frac{\lambda}{\mu} \quad \text{and} \quad \alpha^{-1} = \sum_{i=0}^{n} p_i. \tag{2.2}$$

Here p_i is the probability that there are i customers in the queueing system in the steady-state and α is the *normalization constant.*

Example 2.1. In the one-server system, the steady-state probability distribution is given by

$$p_i = \frac{\rho^i(1-\rho)}{1-\rho^n} \quad \text{where} \quad \rho = \frac{\lambda}{\mu}.$$

When the system has no limit on waiting space, we have a $M/M/1/\infty$ queue (or simply M/M/1 queue). Suppose that $\rho < 1$, the steady-state probability becomes

$$\lim_{n\to\infty} p_i = \rho^i(1-\rho).$$

The expected number of customers in the system is given by

$$\begin{aligned} L_c &= \sum_{i=0}^{\infty} i p_i = \sum_{i=0}^{\infty} i\rho^i(1-\rho) \\ &= \frac{\rho(1-\rho)}{(1-\rho)^2} = \frac{\rho}{1-\rho}. \end{aligned} \tag{2.3}$$

The expected number of customers waiting in the queue is given by

$$\begin{aligned} L_q &= \sum_{i=1}^{\infty}(i-1)p_i = \sum_{i=1}^{\infty}(i-1)\rho^i(1-\rho) \\ &= \frac{\rho}{1-\rho} - \rho = \frac{\rho^2}{1-\rho}. \end{aligned} \tag{2.4}$$

Moreover the expected number of customers in service is given by

$$L_s = 0 \cdot p_0 + 1 \cdot \sum_{i=1}^{\infty} p_i = 1 - (1-\rho) = \rho.$$

2.1.2 An M/M/$s/n - s - 1$ Queueing System

Now let us consider a more general queueing system with customer arrival rate being λ. Suppose the system has s parallel and identical exponential servers with

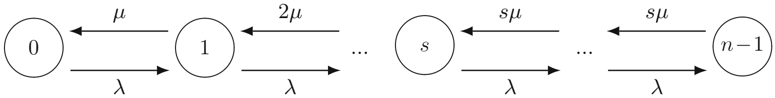


Fig. 2.2 The Markov chain for the one-queue system (s servers)

service rate being μ and there are $n - s - 1$ waiting spaces in the system. The queueing discipline is *First-come-first-serve*. Again when a customer arrives and finds all the servers are busy, the customer can still wait in the queue provided that there is a waiting space available. Otherwise, the customer has to leave the system. To apply the continuous time Markov chains for modeling this queueing system, one has to obtain the waiting time for the departure of one customer when there are more than one customer (let us say k customers) in the queueing system. We need the following lemma.

Lemma 2.2. *Suppose that $X_1, X_2, \ldots, X_k$ are independent, identical, exponential random variables with mean μ^{-1}, and consider the corresponding order statistics*

$$X_{(1)} \le X_{(2)} \le \cdots \le X_{(k)}.$$

Then $X_{(1)}$ is again exponentially distributed with mean $\frac{1}{k}$ times the mean of the original random variables.

Proof. We observe that

$$X_{(1)} = \min(X_1, X_2, \ldots, X_k).$$

Hence $X_{(1)} > x$ if and only if all $X_i > x$ $(i = 1, 2, \ldots, k)$. We therefore have

$$\begin{aligned} P\{X_{(1)} > x\} &= P\{X_1 > x\}P\{X_2 > x\} \cdots P\{X_k > x\} \\ &= (e^{-\mu x})^k \\ &= e^{-k\mu x}. \end{aligned}$$

Again it is still exponentially distributed with mean $1/(k\mu)$. If we use the number of customers in the queue to represent the state of the system, then there are n states, namely $0, 1, \ldots, n-1$. The Markov chain for the queueing system is given in Fig. 2.2. The number of customers in the system is used to represent the states in the Markov chain. Clearly it is an irreducible Markov chain.

If we order the states of the system in increasing number of customers, it is not difficult to show that the generator matrix for this queueing system is given by the following $n \times n$ tri-diagonal matrix $A_2 = A_{(n,s,\lambda,\mu)}$ where

$$A_2 = \begin{pmatrix} \lambda & -\mu & & & & & & 0 \\ -\lambda & \lambda+\mu & -2\mu & & & & & \\ & \ddots & \ddots & \ddots & & & & \\ & & -\lambda & \lambda+(s-1)\mu & -s\mu & & & \\ & & & -\lambda & \lambda+s\mu & -s\mu & & \\ & & & & \ddots & \ddots & \ddots & \\ & & & & & -\lambda & \lambda+s\mu & -s\mu \\ 0 & & & & & & -\lambda & s\mu \end{pmatrix} \qquad (2.5)$$

and the underlying Markov chain is irreducible. The solution for the steady-state probability distribution can be shown to be

$$\mathbf{p}_{(n,s,\lambda,\mu)} = (p_0, p_1, \ldots, p_{n-1})^T \qquad (2.6)$$

where

$$p_i = \alpha \prod_{k=1}^{i+1} \frac{\lambda}{\mu \min\{k, s\}}$$

and

$$\alpha^{-1} = \sum_{i=0}^{n} p_i .$$

Here p_i is the probability that there are i customers in the queueing system in steady-state and α is the *normalization constant*.

2.1.3 Allocation of the Arrivals in a System of M/M/1/∞ Queues

In this subsection, we consider a queueing system consisting of n independent M/M/1/∞ queues. The service rate of the server at the ith queue is μ_i. Again we assume that the arrival process is a Poisson process with rate λ. The allocation of arrivals is an important decision process in a queueing system, see for instance [193] and the references therein. Here we consider an allocation process proposed in [193], which is implemented such that it diverts an arrived customer to queue i with a probability given by

$$\frac{\lambda_i}{\lambda_1 + \ldots + \lambda_n} = \frac{\lambda_i}{\lambda}.$$

Then the input process of Queue i is a Poisson process with rate λ_i. The objective here is to find the parameters λ_i such that some system performance is optimized. Here we must assume $\lambda_i < \mu_i$.

There are many system performance indicators. In what follows, our main objective is to minimize the total expected number of customers in the system. We first obtain the expected number of customers in Queue i by the result in (2.3) as follows:

$$\frac{\lambda_i/\mu_i}{1-\lambda_i/\mu_i}.$$

Then the total expected number of customers in the system is

$$\sum_{i=1}^{n}\frac{\lambda_i/\mu_i}{1-\lambda_i/\mu_i}.$$

The optimization problem is then given as follows:

$$\min_{\lambda_1,\ldots,\lambda_n}\left\{\sum_{i=1}^{n}\frac{\lambda_i/\mu_i}{1-\lambda_i/\mu_i}\right\}$$

subject to

$$\sum_{i=1}^{m}\lambda_i = \lambda$$

and

$$0 \le \lambda_i < \mu \quad \text{for} \quad i = 1, 2, \ldots, n.$$

The Lagrangian function is given by

$$L(\lambda_1,\ldots,\lambda_n,m) = \sum_{i=1}^{n}\frac{\lambda_i/\mu_i}{1-\lambda_i/\mu_i} - m\left(\sum_{i=1}^{n}\lambda_i - \lambda\right).$$

Solving the following equations

$$\frac{\partial L}{\partial \lambda_i} = 0,\ i = 1, 2, \ldots, n \quad \text{and} \quad \frac{\partial L}{\partial m} = 0$$

we get the optimal solution

$$\lambda_i = \mu_i\left(1 - \frac{1}{\sqrt{m\mu_i}}\right) < \mu_i,\ \ i = 1, 2, \ldots, n$$

where

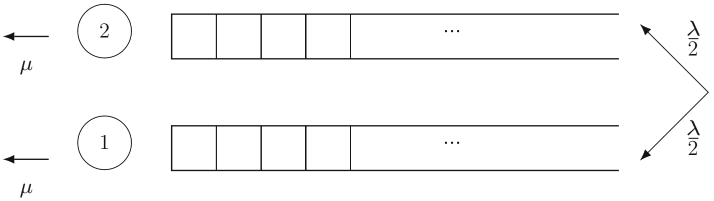


Fig. 2.3 Case (i) Two M/M/1/∞ Queues

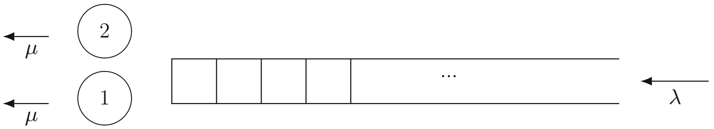


Fig. 2.4 Case (ii) One M/M/2/∞ Queue

$$m = \left(\frac{\sum_{i=1}^{n} \sqrt{\mu_i}}{\sum_{i=1}^{n} \mu_i - \lambda} \right)^2 .$$

Another possible system performance indicator is the total expected number of customers waiting in the system. One can carry out a similar analysis and we leave the problem to the readers as an exercise.

2.1.4 Two M/M/1 Queues or One M/M/2 Queue?

In an M/M/1/ queueing system with service rate μ and arrival rate λ (we assume that $\lambda < \mu$), suppose one extra identical server can be added, then which of following situations is better (we assume that $\lambda < \mu$)? (i) Separate the two operators. Therefore we have two M/M/1/∞ queues. In this case, we assume that an arriving customer can either join the first queue or the second with equal chance. (ii) Join the two operators together. Therefore we have an M/M/2/∞ queue (Figs. 2.3 and 2.4).

To determine which case is better, one can calculate the expected number of customers in both cases. Very often, in our consideration, the smaller the better. In case (i), using the result in (2.3), the expected number of customers in any one of the queueing systems will be given by

$$\frac{(\frac{\lambda}{2\mu})}{1 - (\frac{\lambda}{2\mu})}.$$

Hence the total expected number of customers in the queueing system is

$$S_1 = 2 \times \frac{(\frac{\lambda}{2\mu})}{1-(\frac{\lambda}{2\mu})} = \frac{(\frac{\lambda}{\mu})}{1-(\frac{\lambda}{2\mu})}.$$

Now we examine the second case. In Case (ii), the expected number of customers in the queueing system will be given by

$$S_2 = \frac{(\frac{\lambda}{\mu})}{1-(\frac{\lambda}{2\mu})^2} \tag{2.7}$$

and this is left as an exercise. It is straightforward to see that $S_2 < S_1$. We then conclude that Case (ii) is better. We should put all the servers together if our concern is to minimize the total number of customers in the system.

2.1.5 The Two-Queue Free System

In this subsection, we introduce a higher dimensional queueing system. Suppose that there are two one-queue systems as discussed in Sect. 2.1.2. This queueing system consists of two independent queues with the number of identical servers and waiting spaces being s_i and $n_i - s_i - 1$ $(i = 1, 2)$ respectively.

If we let the arrival rate of customers in Queue i be λ_i and the service rate of the servers be μ_i $(i = 1, 2)$ then the states of the queueing system can be represented by the elements in the following set:

$$S = \{(i, j) | 0 \le i \le n_1, 0 \le j \le n_2\}$$

where (i, j) represents the state that there are i customers in Queue 1 and j customers in Queue 2. Thus this is a two-dimensional queueing model. If we order the states lexicographically, then the generator matrix can be shown to be the following $n_1 n_2 \times n_1 n_2$ matrix in *tensor product* form [37, 46]:

$$A_3 = I_{n_1} \otimes A_{(n_2,s_2,\lambda_2,\mu_2)} + A_{(n_1,s_1,\lambda_1,\mu_1)} \otimes I_{n_2}. \tag{2.8}$$

Here $\otimes$ is the Kronecker tensor product [108, 119]. The Kronecker tensor product of two matrices A and B of sizes $p \times q$ and $m \times n$ respectively is a $(pm) \times (qn)$ matrix given as follows:

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & \cdots & a_{1q}B \\ a_{21}B & \cdots & \cdots & a_{2q}B \\ \vdots & \vdots & \vdots & \vdots \\ a_{p1}B & \cdots & \cdots & a_{pq}B \end{pmatrix}.$$

The Kronecker tensor product is a useful tool for representing generator matrices in many queueing systems and stochastic automata networks [37,46,141,142,191]. For this two-queue free queueing system, it is also not difficult to show that the steady-state probability distribution is given by the probability distribution vector

$$\mathbf{p}_{(\mathbf{n_1},\mathbf{s_1},\lambda_1,\mu_1)} \otimes \mathbf{p}_{(\mathbf{n_2},\mathbf{s_2},\lambda_2,\mu_2)}. \tag{2.9}$$

2.1.6 The Two-Queue Overflow System

Now let us add the following system dynamics to the two-queue free system discussed in Sect. 2.1.5. In this queueing system, we allow overflow of customers from Queue 2 to Queue 1 whenever Queue 2 is full and there is still waiting space in Queue 1; see for instance Fig. 2.5 (Taken from [46]). This is called the two-queue overflow system; see Kaufman [37,46,130].

In this case, the generator matrix is given by the following matrix:

$$A_4 = I_{n_1} \otimes A_{(n_2,s_2,\lambda_2,\mu_2)} + A_{(n_1,s_1,\lambda_1,\mu_1)} \otimes I_{n_2} + R \otimes {\mathbf{e}_{\mathbf{n_2}}}^T \mathbf{e}_{\mathbf{n_2}}. \tag{2.10}$$

Here $\mathbf{e}_{\mathbf{n_2}}$ is the unit vector $(0, 0, \ldots, 0, 1)$ and

$$R = \begin{pmatrix} \lambda_2 & & & & 0 \\ -\lambda_2 & \lambda_2 & & & \\ & -\lambda_2 & \ddots & & \\ & & \ddots & \lambda_2 & \\ 0 & & & -\lambda_2 & 0 \end{pmatrix}. \tag{2.11}$$

In fact

$$A_4 = A_3 + R \otimes {\mathbf{e}_{\mathbf{n_2}}}^T \mathbf{e}_{\mathbf{n_2}},$$

where $R \otimes {\mathbf{e}_{\mathbf{n_2}}}^T \mathbf{e}_{\mathbf{n_2}}$ is the matrix describing the overflow of customers from Queue 2 to Queue 1. Unfortunately, there is no analytical solution for the generator matrix A_4.

For an overflow queueing system, a closed form solution of the steady-state probability distribution may not always be available. In fact, there are a lot applications related to queueing systems whose problem size are very large [26–28, 36, 37, 46, 77, 152]. Direct methods for solving the probability distribution such as the Gaussian elimination and LU factorization can be found in [132, 191]. Another popular method is called the matrix analytic methods [141,142]. Apart from these direct methods, another class of popular numerical methods called iterative methods, exists. This includes the classical iterations introduced in Chap. 1, such as the Jacobi method, Gauss-Seidel method and SOR method. Sometimes when the generator matrix has a block structure, then the block Jacobi method, block Gauss-Seidel method and block SOR method are also popular methods [108].

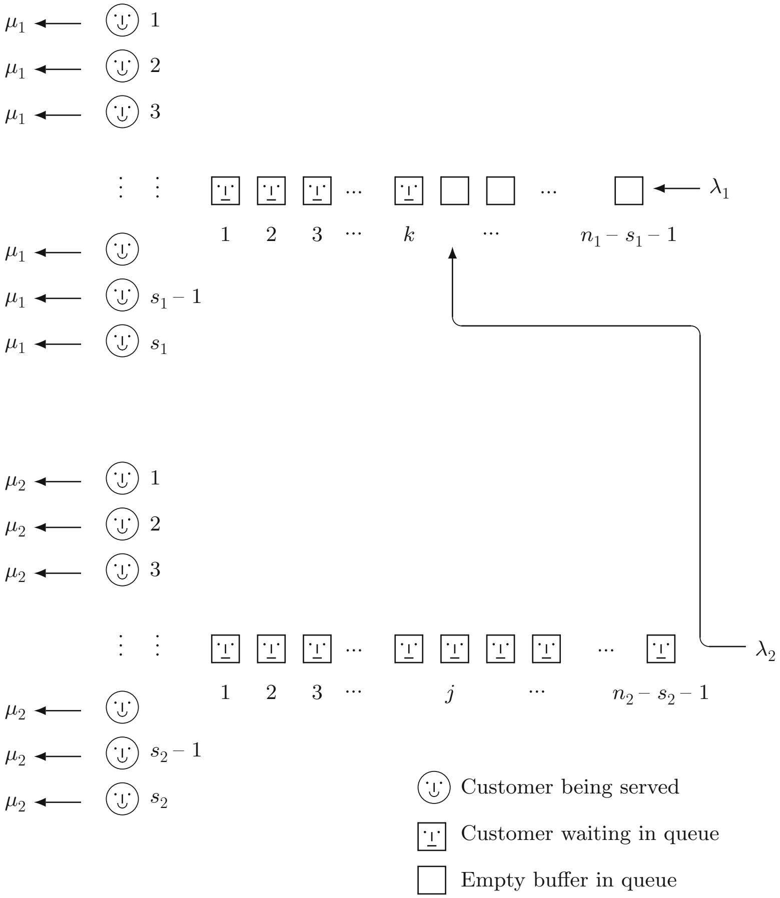


Fig. 2.5 The two-queue overflow system

A hybrid numerical algorithm which combines both SOR and a genetic algorithm has also been introduced by Ching et al. [213] for solving queueing systems. Conjugate gradient methods with circulant-based preconditioners are efficient solvers for a class of Markov chains having near-Toeplitz generator matrices. We will briefly discuss this in the following subsection.

2.1.7 The Preconditioning of Complex Queueing Systems

In many complex queueing systems, one observes both block structure, near-Toeplitz structure and sparsity in the generator matrices. Therefore an iterative method such as the CG method can be a good solver with a suitable preconditioner.

2.1.7.1 Circulant-Based Preconditioners

In this subsection, we illustrate how to get a circulant preconditioner from a generator matrix of a queueing system. The generator matrices of the queueing networks can be written in terms of the sum of tensor products of matrices. Very often, a key block structure of a queueing system is the following: $(n + s + 1) \times (n + s + 1)$ tri-diagonal matrix:

$$Q = \begin{pmatrix} \lambda & -\mu & & & & & & & 0 \\ -\lambda & \lambda + \mu & -2\mu & & & & & & \\ & \ddots & \ddots & \ddots & & & & & \\ & & -\lambda & \lambda + (s-1)\mu & -s\mu & & & & \\ & & & -\lambda & \lambda + s\mu & -s\mu & & & \\ & & & & \ddots & \ddots & \ddots & & \\ & & & & & -\lambda & \lambda + s\mu & -s\mu & \\ 0 & & & & & & -\lambda & s\mu & \end{pmatrix}. \quad (2.12)$$

This is the generator matrix of an M/M/s/n queue. In this queueing system there are s independent exponential servers, the customers arrive according to a Poisson process of rate λ and each server has a service rate of μ.

One can observe that if s is fixed and n is large, then Q is close to the following tridiagonal Toeplitz matrix Tri$[\lambda, -\lambda - s\mu, s\mu]$. In fact, if one considers the following circulant matrix $c(Q)$:

$$c(Q) = \begin{pmatrix} \lambda + s\mu & -s\mu & & & -\lambda \\ -\lambda & \lambda + s\mu & -s\mu & & \\ & \ddots & \ddots & \ddots & \\ & & -\lambda & \lambda + s\mu & -s\mu \\ -s\mu & & & -\lambda & \lambda + s\mu \end{pmatrix} \quad (2.13)$$

it is easy to see that

$$\text{rank}(c(Q) - Q) \leq s + 1$$

is independent of n for fixed s. Therefore, for fixed s and large value of n, the approximation is a good one. Moreover, $c(Q)$ can be diagonalized by the discrete Fourier Transformation and a closed form solution of its eigenvalues can be easily obtained. This is important in the convergence rate analysis of the CG method. By applying this circulant approximation to the blocks of the generator matrices, effective preconditioners are constructed and the preconditioned systems are also proved to have singular values clustered around one, see for instance Chan and Ching [37]. A number of related applications can also be found in [36,37,41,43,45,46,49].

2.1.7.2 Toeplitz-Circulant-Based Preconditioners

Another class of queueing systems with batch arrivals have been discussed by Chan and Ching in [36]. The generator matrices of queueing systems of s identical exponential servers with service rate μ take the form

$$A_n = \begin{pmatrix} \lambda & -\mu & 0 & 0 & 0 & \dots & 0 \\ -\lambda_1 & \lambda+\mu & -2\mu & 0 & 0 & \dots & 0 \\ -\lambda_2 & -\lambda_1 & \lambda+2\mu & \ddots & \ddots & & \vdots \\ \vdots & -\lambda_2 & \ddots & \ddots & -s\mu & \ddots & \\ & \vdots & \ddots & \ddots & \lambda+s\mu & \ddots & 0 \\ -\lambda_{n-2} & -\lambda_{n-3} & \cdots & & \ddots & \ddots & -s\mu \\ -r_1 & -r_2 & -r_3 & \cdots & -r_{s+1} & \cdots & s\mu \end{pmatrix}, \qquad (2.14)$$

where r_i are such that each column sum of A_n is zero, i.e.

$$r_i = \lambda - \sum_{k=n-i}^{\infty} \lambda_k.$$

Here λ is the arrival rate and $\lambda_i = \lambda p_i$ where p_i is the probability that an arrived batch is of size i. It is clear that the matrix is dense and the method of circulant approximation does not work directly in this case. A Toeplitz-circulant type of preconditioner was proposed to solve this queueing system by Chan and Ching [36]. The idea is that the generator matrix is close to a Toeplitz matrix whose generating function has a zero on the unit circle of order one. By factoring the zero, the quotient has no zero on the unit circle. Using this fact, a Toeplitz-circulant preconditioner is then constructed for the queueing system. Both the construction cost and the preconditioner system can be solved in $n\log(n)$ operations. Moreover, the preconditioned system was proved to have singular values clustered around one. Hence a very fast convergence rate is expected when the CG method is applied to solving the preconditioned system.

This idea was further applied to queueing systems with batch arrivals and negative customers, see Ching [48]. The term "negative customer" was first introduced by Gelenbe et al. [102–104] in the modelling of neural networks. Here the role of a negative customer is to remove a number of customers waiting in the queueing system. For example, one may consider a communication network in which messages are transmitted in a packet-switching mode. When a server fails (this corresponds to an arrival of a negative customer) during a transmission, parts of the messages will be lost. One may also consider a manufacturing system where a negative customer represents a cancellation of a job. These lead to many practical applications in the modelling of physical systems.

In the queueing system, we assume that the arrival process of the batches of customers follows a Poisson process of rate λ. The batch size again follows a stationary distribution of p_i $(i = 1, 2, \ldots,)$. Here p_i is the probability that an arrived batch is of size i. It is also assumed that the arrival process of negative customers is a Poisson process with rate τ. The number of customers to be removed is assumed to follow a probability distribution

$$b_i (i = 1, 2, \ldots,).$$

Furthermore, if the arrived negative customer is supposed to remove i customers in the system, but the number of customers in the system is less than i, then the queueing system will become empty. The removal (killing) strategy here is to remove the customers in the front of the queue, i.e. "Remove the Customers at the Head" (RCH). For $i \geq 1$, we let

$$\tau_i = b_i \tau$$

where b_i is the probability that the number of customers to be removed is i and therefore we have

$$\tau = \sum_{k=1}^{\infty} \tau_k.$$

The generator matrices of the queueing systems take the following form:

$$A_n = \begin{pmatrix}
\lambda & -u_1 & -u_2 & -u_3 & \ldots & \ldots & \ldots & -u_{n-1} \\
-\lambda_1 & \lambda+\tau+\mu & -2\mu-\tau_1 & -\tau_2 & -\tau_3 & \ldots & \ldots & -\tau_{n-2} \\
-\lambda_2 & -\lambda_1 & \lambda+\tau+2\mu & \ddots & \ddots & \ddots & & \vdots \\
\vdots & -\lambda_2 & \ddots & \ddots & -s\mu-\tau_1 & -\tau_2 & \ddots & \vdots \\
\vdots & \vdots & \ddots & \ddots & \lambda+\tau+s\mu & \ddots & \ddots & -\tau_3 \\
\vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & -\tau_2 \\
-\lambda_{n-2} & -\lambda_{n-3} & -\lambda_{n-4} & \cdots & \lambda_2 & -\lambda_1 & \lambda+\tau+s\mu & -s\mu-\tau_1 \\
-v_1 & -v_2 & -v_3 & \cdots & \cdots & -v_{n-2} & -v_{n-1} & \tau+s\mu
\end{pmatrix}.$$

Here

$$\lambda = \sum_{i=1}^{\infty} \lambda_i \quad \text{and} \quad \lambda_i = \lambda p_i$$

and

$$u_1 = \tau \quad \text{and} \quad u_i = \tau - \sum_{k=1}^{i-1} \tau_k \quad \text{for } i = 2, 3, \ldots$$

and v_i is defined such that the ith column sum is zero. The generator matrices enjoy the same near-Toeplitz structure. Toeplitz-circulant preconditioners can be constructed similarly and the preconditioned systems are proved to have singular values clustered around one, Ching [48].

Finally, we remark that there is another efficient iterative method for solving queueing systems which is not covered in this text, the multigrid methods. Interested readers may consult the following references Bramble [24], Chan et al. [38], Chang et al. [40] and McCormick [154].

2.2 Search Engines

In this section, we introduce a very important algorithm used by Google for ranking webpages on the Internet. In surfing the Internet, web surfers usually utilize search engines to find the related webpages satisfying their queries. Unfortunately, there are often thousands of webpages which are relevant to their queries. Therefore a proper list of the webpages in a certain order of importance is necessary. The list should also be updated regularly and frequently. Thus it is important to seek for a fast algorithm for updating the PageRank so as to reduce the time lag of updating. As it turns this problem is difficult. The reason is not just because of the huge size of the webpages on the Internet, but also their numbers keep on growing rapidly.

PageRank has been proposed by Page et al. [169] to reflect the importance of each webpage. Larry Page and Sergey Brin are the co-founders of Google. In fact, one can find the following statement at Google's website (http://www.google.com/technology/): "The heart of our software is PageRank$^{\text{TM}}$, a system for ranking web pages developed by our founders Larry Page and Sergey Brin at Stanford University. And while we have dozens of engineers working to improve every aspect of Google on a daily basis, PageRank continues to provide the basis for all of our web search tools."

A similar idea of ranking journals has been proposed by Garfield [100, 101] as a measure of standing for journals, which is called the *impact factor*. The impact factor of a journal is defined as the average number of citations per recently published papers in that journal. By regarding each webpage as a journal, this was then extended to measure the importance of a webpage in the PageRank Algorithm.

The PageRank is defined as follows. Let N be the total number of webpages in the web and we define a matrix Q called the *hyperlink matrix*. Here

$$Q_{ij} = \begin{cases} 1/k & \text{if Webpage } i \text{ is an outgoing link of Webpage } j; \\ 0 & \text{otherwise;} \end{cases}$$

and k is the total number of outgoing links of Webpage j. For simplicity of discussion, here we assume that $Q_{ii} > 0$ for all i. This means that for each webpage there is a link pointing to itself. Hence Q can be regarded as a transition probability matrix of a Markov chain of a random walk. The analogy is that one may regard a web surfer as a random walker and the webpages as the states of the Markov chain. Assuming that this underlying Markov chain is irreducible, then the steady-state probability distribution

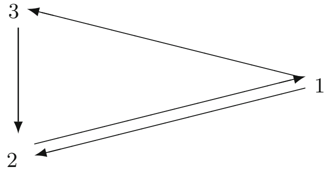


Fig. 2.6 An example of three webpages

$$(p_1, p_2, \dots, p_N)^T$$

of the states (webpages) exists. Here p_i is the proportion of time that the random walker (web surfer) spends visiting State (Webpage) i. The larger the value of p_i is, the more important Webpage i will be. Thus the PageRank of Webpage i is then defined as p_i. If the Markov chain is not irreducible then one can still follow the treatment in the next subsection.

An Example

We consider a web of an internet with 3 webpages:1, 2, 3 such that
$1 \to 1,\ 1 \to 2,\ 1 \to 3$
$2 \to 1,\ 2 \to 2,$
$3 \to 2,\ 3 \to 3.$

One can represent the relationship by the following Markov chain (Fig. 2.6).

The transition probability matrix of this Markov chain is then given by

$$Q = \begin{array}{c} 1 \\ 2 \\ 3 \end{array} \begin{pmatrix} 1/3 & 1/2 & 0 \\ 1/3 & 1/2 & 1/2 \\ 1/3 & 0 & 1/2 \end{pmatrix}.$$

The steady-state probability distribution of the Markov chain

$$\mathbf{p} = (p_1, p_2, p_3)$$

satisfies

$$\mathbf{p} = Q\mathbf{p} \quad \text{and} \quad p_1 + p_2 + p_3 = 1.$$

Solving the above linear system, we get

$$(p_1, p_2, p_3) = (\frac{3}{9}, \frac{4}{9}, \frac{2}{9}).$$

Therefore the ranking of the webpages is:

$$\text{Webpage 2} > \text{Webpage 1} > \text{Webpage 3}.$$

One can also interpret the result as follows. Both Webpages 1 and 3 point to Webpage 2 and therefore Webpage 2 is the most important. Since webpage 2 points to Webpage 1 but not Webpage 3, Webpage 1 is then more important than Webpage 3.

In terms of the actual Internet, since the size of the Markov chain is huge and the time for computing the PageRank required by Google for computing the Page Rank is just a few days, a direct method for solving the steady-state probability is not desirable. Iterative methods [10] and decomposition methods [7] have been proposed to solve the problem. Another pressing issue is that the number of webpages grows rapidly, and the PageRank of each webpage has to be updated regularly. Here we seek for adaptive and parallelizable numerical algorithms for solving the PageRank problem. One potential method is the hybrid iterative method proposed in Yuen et al. [213]. The hybrid iterative method was first proposed by He et al. [116] for solving the numerical solutions of PDEs and it has also been successfully applied to solving the steady-state probability distributions of queueing networks [213]. The hybrid iterative method combines the evolutionary algorithm and the Successive Over-Relaxation (SOR) method. The evolutionary algorithm allows the relaxation parameter w to be adaptive in the SOR method. Since the cost of the SOR method per iteration is more expensive and less efficient in the parallel computing for our problem (as the matrix system is huge), here we will also consider replacing the role of SOR method by the Jacobi Over-Relaxation (JOR) method [108, 132]. The JOR method is easier to be implemented in parallel computing environments. Here we present hybrid iterative methods based on SOR/JOR method, and the evolutionary algorithm. The hybrid method allows the relaxation parameter w to be adaptive in the SOR/JOR method. We give a brief mathematical discussion on the PageRank approach. We then briefly describe the power method, a popular approach for solving the PageRank problem.

2.2.1 The PageRank Algorithm

The PageRank Algorithm has been used successfully in ranking the importance of webpages by Google (http://www.search-engine-marketing-sem.com/Google/GooglePageRank.html.). Consider a web of N webpages with Q being the hyperlink matrix. Since the matrix Q can be reducible, to tackle this problem, one can consider the revised matrix P:

$$P = \alpha \begin{pmatrix} Q_{11} & Q_{12} & \cdots & Q_{1N} \\ Q_{21} & Q_{22} & \cdots & Q_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ Q_{N1} & Q_{N2} & \cdots & Q_{NN} \end{pmatrix} + \frac{(1-\alpha)}{N} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & \cdots & 1 \end{pmatrix} \tag{2.15}$$

where $0 < \alpha < 1$. In this case, the matrix P is irreducible and aperiodic, therefore the steady-state probability distribution exists and is unique [181]. Typical values for α are 0.85 and $(1 - 1/N)$, see for instance [10, 113]. The value $\alpha = 0.85$ is a popular one and the power method works very well for this problem [113–115]. We remark that this value can affect the original ranking of the webpages, see for instance, the example in Sect. 2.2.3.

One can interpret (2.15) as follows. The idea of the algorithm is that for a network of N webpages, each webpage has an inherent importance of $(1-\alpha)/N$. If a page P_i has an importance of p_i, then it will contribute an importance of αp_i which is shared among the webpages that it points to. The importance of webpage P_i can be obtained by solving the following linear system of equations, subject to the normalization constraint:

$$\begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_N \end{pmatrix} = \alpha \begin{pmatrix} Q_{11} & Q_{12} & \cdots & Q_{1N} \\ Q_{21} & Q_{22} & \cdots & Q_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ Q_{N1} & Q_{N2} & \cdots & Q_{NN} \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_N \end{pmatrix} + \frac{(1-\alpha)}{N} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}. \tag{2.16}$$

Since

$$\sum_{i=1}^{N} p_i = 1,$$

(2.16) can be re-written as

$$(p_1, p_2, \ldots, p_N)^T = P(p_1, p_2, \ldots, p_N)^T.$$

2.2.2 The Power Method

The power method is a popular method for solving the PageRank problem. The power method is an iterative method for solving the largest eigenvalue in modulus (the dominant eigenvalue) and its corresponding eigenvector [108]. The idea of the power method can be briefly explained as follows. Given an $n \times n$ matrix A and suppose that (a) there is a single eigenvalue of maximum modulus and the eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_n$ are labelled such that

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \cdots \geq |\lambda_n|;$$

(b) there is a linearly independent set of n unit eigenvectors. This means that there is a basis

$$\left\{\mathbf{u}^{(1)}, \mathbf{u}^{(2)}, \ldots, \mathbf{u}^{(n)}\right\}$$

such that

$$A\mathbf{u}^{(i)} = \lambda_i \mathbf{u}^{(i)}, \quad i = 1, 2, \ldots, n, \quad \text{and} \quad \|\mathbf{u}^{(i)}\| = 1.$$

Then beginning with an initial vector $\mathbf{x}^{(0)}$, one may write

$$\mathbf{x}^{(0)} = a_1\mathbf{u}^{(1)} + a_2\mathbf{u}^{(2)} + \cdots + a_n\mathbf{u}^{(n)}.$$

Now we iterate the initial vector with the matrix A as follows:

$$A^k\mathbf{x}^{(0)} = a_1A^k\mathbf{u}^{(1)} + \ldots + a_nA^k\mathbf{u}^{(n)} = a_1\lambda_1^k\mathbf{u}^{(1)} + \ldots + a_n\lambda_n^k\mathbf{u}^{(n)}$$

$$= \lambda_1^k \left\{ a_1\mathbf{u}^{(1)} + \left(\frac{\lambda_2}{\lambda_1}\right)^k a_n\mathbf{u}^{(2)} + \ldots + \left(\frac{\lambda_n}{\lambda_1}\right)^k a_n\mathbf{u}^{(n)} \right\}.$$

Since

$$\frac{|\lambda_i|}{|\lambda_1|} < 1 \quad \text{for } i = 2, \ldots, n,$$

we obtain

$$\lim_{k\to\infty} \frac{|\lambda_i|^k}{|\lambda_1|^k} = 0 \quad \text{for } i = 2, \ldots, n.$$

Hence we have

$$A^k\mathbf{x}^{(0)} \approx a_1\lambda_1^k\mathbf{u}^{(1)}.$$

To get an approximation for $\mathbf{u}^{(1)}$ we introduce a normalization in the iteration:

$$\mathbf{r}_{k+1} = \frac{A^{k+1}\mathbf{x}^{(0)}}{\|A^k\mathbf{x}^{(0)}\|_2}$$

and we have

$$\lim_{k\to\infty} \mathbf{r}_{k+1} = \lim_{k\to\infty} \frac{a_1\lambda_1^{k+1}\mathbf{u}^{(1)}}{\|a_1\lambda_1^k\mathbf{u}^{(1)}\|_2} = \lambda_1\mathbf{u}^{(1)}.$$

It turns out that for the PageRank problem, the largest eigenvalue of P is 1, and the corresponding eigenvector in normalized form is the PageRank vector. The main computational cost of this method comes from the matrix-vector multiplications. The convergence rate of the power method depends on the ratio of $|\lambda_2/\lambda_1|$ where λ_1 and λ_2 are respectively the largest and the second largest eigenvalues of the matrix P. It was proved by Haveliwala and Kamvar [113] that for the second largest eigenvalue of P, we have

$$|\lambda_2| \le \alpha \quad \text{for} \quad 0 \le \alpha \le 1.$$

Since $\lambda_1 = 1$, the convergence rate of the power method is α, see for instance [108]. A popular value for α is 0.85. With this value, it is mentioned in Kamvar et al. [129] that the power method on a web data set of over 80 million pages converges in about 50 iterations.

2.2.3 An Example

In this subsection, we consider a small example of six webpages. This example demonstrates that the value of $\alpha = 0.85$ can affect the original ranking ($\alpha = 1$) of the webpages even if the number of webpages is small. In the example, the webpages are organized as follows:

Webpage 1 $\rightarrow 1, 3, 4, 5$
Webpage 2 $\rightarrow 2, 3, 5, 6$
Webpage 3 $\rightarrow 1, 2, 3, 4, 5, 6$
Webpage 4 $\rightarrow 2, 3, 4, 5$
Webpage 5 $\rightarrow 1, 3, 5$
Webpage 6 $\rightarrow 1, 6$

From the given structure of the webpages, we have the hyperlink matrix as follows:

$$Q = \begin{pmatrix} 0.2500 & 0.0000 & 0.1667 & 0.0000 & 0.3333 & 0.5000 \\ 0.0000 & 0.2500 & 0.1667 & 0.2500 & 0.0000 & 0.0000 \\ 0.2500 & 0.2500 & 0.1667 & 0.2500 & 0.3333 & 0.0000 \\ 0.2500 & 0.0000 & 0.1667 & 0.2500 & 0.0000 & 0.0000 \\ 0.2500 & 0.2500 & 0.1667 & 0.2500 & 0.3333 & 0.0000 \\ 0.0000 & 0.2500 & 0.1667 & 0.0000 & 0.0000 & 0.5000 \end{pmatrix}$$

then the steady-state probability distribution is given by

$$(0.2260, 0.0904, 0.2203, 0.1243, 0.2203, 0.1186)^T$$

and the ranking should be $1 > 3 \geq 5 > \underline{4 > 6} > 2$. For $\alpha = 0.85$, we have

$$P = \begin{pmatrix} 0.2375 & 0.0250 & 0.1667 & 0.0250 & 0.3083 & 0.4500 \\ 0.0250 & 0.2375 & 0.1667 & 0.2375 & 0.0250 & 0.0250 \\ 0.2375 & 0.2375 & 0.1667 & 0.2375 & 0.3083 & 0.0250 \\ 0.2375 & 0.0250 & 0.1667 & 0.2375 & 0.0250 & 0.0250 \\ 0.2375 & 0.2375 & 0.1667 & 0.2375 & 0.3083 & 0.0250 \\ 0.0250 & 0.2375 & 0.1667 & 0.0250 & 0.0250 & 0.4500 \end{pmatrix}.$$

In this case, the steady-state probability distribution is given by

$$(0.2166, 0.1039, 0.2092, 0.1278, 0.2092, 0.1334)^T$$

and the ranking should be $1 > 3 \geq 5 > \underline{6 > 4} > 2$. We observe that the ranking of states 6 and 4 are inter-changed in the two approaches.

2.2.4 The SOR/JOR Method and the Hybrid Method

In this section, we present a hybrid algorithm for solving the steady-state probability of a Markov chain, Yuen et al. [213, 214]. We first give a review on the JOR method for solving linear systems, in particular solving the steady-state probability distribution of a finite Markov chain. We then introduce the hybrid algorithm based on the SOR/JOR method and the evolutionary algorithm. For the SOR method, it has been discussed in Chap. 1. When we consider a non-singular linear system $B\mathbf{x} = \mathbf{b}$, the JOR method is a classical iterative method. The idea of the JOR method can be explained as follows. We write $B = D - (D - B)$ where D is the diagonal part of the matrix B. Given an initial guess of the solution, $\mathbf{x}_0$, the JOR iteration scheme reads:

$$\begin{aligned} \mathbf{x}_{n+1} &= (I - wD^{-1}B)\mathbf{x}_n + wD^{-1}\mathbf{b} \\ &\equiv B_w\mathbf{x}_n + wD^{-1}\mathbf{b}. \end{aligned} \tag{2.17}$$

The parameter w is called the relaxation parameter and it lies between 0 and 1 [8]. Clearly if the scheme converges, the limit will be the solution of $B\mathbf{x} = \mathbf{b}$. The choice of the relaxation parameter w affects the convergence rate of the SOR/JOR method significantly, see for instance [213, 214]. In general, the optimal value of w is unknown. For more details about the SOR/JOR method and its properties, we refer readers to [8, 108].

The generator matrix P of an irreducible Markov chain is singular and has a null space of dimension one (the null vector corresponds to the steady-state probability distribution). One possible way to solve the steady-state probability distribution is to consider the following revised system:

$$A\mathbf{x} = (P + \mathbf{e}_n^T\mathbf{e}_n)\mathbf{x} = \mathbf{e}_n^T \tag{2.18}$$

where $\mathbf{e}_n = (0, 0, \ldots, 0, 1)$ is a unit vector. The steady-state probability distribution is then obtained by normalizing the solution $\mathbf{x}$, see for instance, Ching [46]. We remark that the linear system (2.18) is irreducibly diagonal dominant. The hybrid method based on He et al. [116] and Yuen et al. [213] consists of four major steps: *initialization, mutation, evaluation and adaptation* [174].

In the initialization step, we define the size of the population k of the approximate steady-state probability distribution. This means that we also define k approximates to initialize the algorithm. Then we use the JOR iteration in (2.17) as the "mutation step". In the evaluation step, we evaluate how "good" each member in the population is by measuring their residuals. In this case, it is clear that the smaller the residual the better the approximation and therefore the better the member in the population. In the adaptation step, the relaxation parameters of the "weak" members are migrated (with certain probability) towards the best relaxation parameter. The hybrid algorithm reads:

Step 1: Initialization: We first generate an initial population of k $(2 \leq k \leq n)$ identical steady-state probability distributions as follows:

$$\{\mathbf{e}_i : i = 1, 2, \ldots, k\}$$

where $\mathbf{e}_i = (0, \ldots, 0, \underbrace{1}_{ith\ entry}, 0, \ldots, 0)^T$. We then compute

$$r_i = ||B\mathbf{e}_i - \mathbf{b}||_2$$

and define a set of relaxation parameters $\{w_1, w_2, \ldots, w_k\}$ such that

$$w_i = \tau + \frac{(1-2\tau)(k-i)}{k-1}, \quad i = 1, 2, \ldots, k.$$

Here $\tau \in (0, 1)$ and therefore $w_i \in [\tau, 1-\tau]$. We set $\tau = 0.01$ in our numerical experiments. We then obtain a set of ordered triples

$$\{(\mathbf{e}_i, w_i, r_i) : i = 1, 2, \ldots, k\}.$$

Step 2: Mutation: The mutation step is carried out by doing a SOR/JOR iteration on each member $\mathbf{x}_i$ ($\mathbf{x}_i$ is used as the initial in the SOR/JOR) of the population with their corresponding w_i. We then get a new set of approximate steady-state probability distributions: $\mathbf{x}_i$ for $i = 1, 2, \ldots, k$. Hence we have a new set of

$$\{(\mathbf{x}_i, w_i, r_i) : i = 1, 2, \ldots, k\}.$$

Go to Step 3.

Step 3: Evaluation: For each $\mathbf{x}_i$, we compute and update its residual

$$r_i = ||B\mathbf{x}_i - \mathbf{b}||_2.$$

This is used to measure how "good" an approximate $\mathbf{x}_i$ is. If $r_j < tol$ for some j then stop and output the approximate steady-state probability distribution $\mathbf{x}_j$. Otherwise we update r_i of the ordered triples

$$\{(\mathbf{x}_i, w_i, r_i) : i = 1, 2, \ldots, k\}$$

and go to Step 4.

Step 4: Adaptation: In this step, the relaxation factors w_k of the weak members (relatively large r_i) in the population are moving towards the best one with certain probability. This process is carried out by first performing a linear search on $\{r_i\}$ to find the best relaxation factor, w_j. We then adjust all the other w_k as follows:

$$w_k = \begin{cases} (0.5+\delta_1)*(w_k+w_j) & \text{if } (0.5+\delta_1)*(w_k+w_j) \in [\tau, 1-\tau] \\ w_k & \text{otherwise,} \end{cases}$$

where δ_1 is a random number in $[-0.01, 0.01]$. Finally the best w_j is also adjusted by

$$w_j = \delta_2 * w_j + (1-\delta_2) * \frac{(w_1 + w_2 + \ldots + w_{j-1} + w_{j+1} + \ldots + w_k)}{k-1}$$

where δ_2 is a random number in $[0.99, 1]$. A new set of $\{w_i\}$ is then obtained and hence

$$\{(\mathbf{x}_i, w_i, r_i) : i = 1, 2, \ldots, k\}.$$

Go to Step 2.

2.2.5 Convergence Analysis

In this section, we consider the linear system $B\mathbf{x} = \mathbf{b}$ where B is strictly diagonal dominant, i.e.

$$|B_{ii}| > \sum_{j=1, j\neq i}^{N} |B_{ij}| \quad \text{for} \quad i = 1, 2, \ldots, N$$

where N is the size of the matrix.

We first prove that the hybrid algorithm converges for a range of w with the SOR method. We begin with the following lemma.

Lemma 2.3. *Let B be a strictly diagonal dominant square matrix and*

$$K = \max_i \left\{ \sum_{j=1, j\neq i}^{m} \frac{|B_{ij}|}{|B_{ii}|} \right\} < 1,$$

then

$$||B_w||_{M_\infty} < 1 \quad \text{for} \quad 0 < w < 2/(1+K)$$

where B_w is defined in (2.15). Here the definition of the matrix $||.||_{M_\infty}$ can be found in Sect. 1.3.1.

Proof. Let $\mathbf{x}$ be an $n \times 1$ vector such that $||\mathbf{x}||_\infty = 1$. We are going to prove that

$$||B_w\mathbf{x}||_\infty \leq 1 \quad \text{for} \quad 0 < w < 2/(1+K).$$

Consider

$$\mathbf{y} = (D - wL)^{-1}((1-w)D + wU)\mathbf{x}$$

and we have

$$(D - wL)\mathbf{y} = ((1 - w)D + wU)\mathbf{x}$$

i.e.,

$$\begin{pmatrix} B_{11} & 0 & \cdots & \cdots & 0 \\ -wB_{21} & B_{22} & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & 0 \\ -wB_{m1} & \cdots & \cdots & -wB_{m,m-1} & B_{mm} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ \vdots \\ y_m \end{pmatrix}$$

$$= \begin{pmatrix} (1-w)B_{11} & wB_{12} & \cdots\cdots & wB_{1m} \\ 0 & (1-w)B_{22} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & wB_{m-1,m} \\ 0 & \cdots & \cdots\ 0 & (1-w)B_{mm} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_m \end{pmatrix}.$$

Case 1: $1 \le w < 2/(K + 1)$.

For the first equation, we have

$$B_{11}y_1 = (1 - w)B_{11}x_1 + w\sum_{j=2}^{m} B_{1j}x_j.$$

Since

$$|x_i| \le 1 \quad \text{and} \quad \sum_{j=2}^{m} |B_{1j}| < K|B_{11}|,$$

we have

$$|y_1| \le |1 - w| + wK = w(1 + K) - 1 < 1.$$

For the second equation, we have

$$B_{22}y_2 = (1 - w)B_{22}x_2 + wB_{21}y_1 + w\sum_{j=3}^{m} B_{2j}x_j.$$

Since

$$|y_1| \le 1, \ |x_i| \le 1$$

and

$$\sum_{j=1, j\neq 2}^{m} |B_{2j}| < K|B_{22}|,$$

we have

$$|y_2| \leq |1 - w| + wK = w(1 + K) - 1 < 1.$$

Inductively, we have $|y_i| < 1$ and hence $||\mathbf{y}||_\infty < 1$. Therefore we have proved that

$$||B_w||_{M_\infty} < 1 \quad \text{for} \quad 1 \leq w < 2/(1 + K).$$

Case 2: $0 < w < 1$.

For the first equation, we have

$$B_{11} y_1 = (1 - w) B_{11} x_1 + w \sum_{j=2}^{m} B_{1j} x_j.$$

Since

$$|x_i| \leq 1 \quad \text{and} \quad \sum_{j=2}^{m} |B_{1j}| < |B_{11}|,$$

we have

$$|y_1| < 1 - w + w = 1.$$

For the second equation, we have

$$B_{22} y_2 = (1 - w) B_{22} x_2 + w B_{21} y_1 + w \sum_{j=3}^{m} B_{2j} x_j.$$

Since

$$|y_1| \leq 1, \; |x_i| \leq 1 \quad \text{and} \quad \sum_{j=1, j \neq 2}^{m} |B_{2j}| < |B_{22}|,$$

we have

$$|y_2| < 1 - w + w = 1.$$

Inductively, we have $|y_i| < 1$ and hence $||\mathbf{y}||_\infty < 1$. Therefore

$$||B_w||_{M_\infty} < 1 \quad \text{for} \quad 0 < w < 1.$$

Combining the results, we have

$$||B_w||_{M_\infty} < 1 \quad \text{for} \quad 0 < w < 2/(1 + K).$$

Proposition 2.4. *The hybrid algorithm converges for* $w \in [\tau, 2/(1 + K) - \tau]$ *where* $0 < \tau < 1/(1 + K)$.

Proof. We note that

$$f(\tau) = \max_{w \in [\tau, 2/(1+K)-\tau]} \{||(B_w)||_{M_\infty}\}$$

exists and is less than one. Let us denote it by $0 \leq f(\tau) < 1$. Therefore in each iteration of the hybrid method, the matrix norm ($||.||_{M_\infty}$) of the residual is decreased by a fraction not less than $f(\tau)$. By using the fact that

$$||ST||_{M_\infty} \leq ||S||_{M_\infty}||T||_{M_\infty},$$

the hybrid algorithm is convergent.

We then prove that the hybrid algorithm with the JOR method converges for a range of w. We have the following lemma.

Lemma 2.5. *Let B be a strictly diagonal dominant square matrix and*

$$K = \max_i \left\{ \sum_{j=1, j \neq i}^{N} \frac{|B_{ji}|}{|B_{ii}|} \right\} < 1,$$

then

$$||B_w||_{M_1} \leq 1 - (1-K)w < 1 \quad \text{for} \quad \tau < w < 1 - \tau$$

where B_w is defined in (2.15). Here the definition of the matrix norm $||.||_{M_1}$ can be found in Sect. 1.3.1 .

By using the similar approach as in Proposition 2.4, one can prove the following proposition.

Proposition 2.6. *The hybrid iterative method converges for $w \in [\tau, 1-\tau]$.*

Proof. We observe that

$$f(\tau) = \max_{w \in [\tau, 1-\tau]} \{||B_w||_{M_1}\}$$

exists and is less than one. Let us denote it by $0 \leq f(\tau) < 1$. Therefore in each iteration of the hybrid method, the matrix norm ($||.||_{M_1}$) of the residual is decreased by a fraction not less than $f(\tau)$. By using the fact that

$$||ST||_{M_1} \leq ||S||_{M_1}||T||_{M_1},$$

the hybrid algorithm is convergent.

We note that the matrix A in (2.16) is irreducibly diagonal dominant only, but not strictly diagonal dominant. Therefore the conditions in Lemmas 2.3 and 2.5 are not satisfied. However, one can always consider a regularized linear system as follows:

$$(A + \epsilon I)\mathbf{x} = \mathbf{b}.$$

Table 2.1 Number of iterations for convergence ($\alpha = 1 - 1/N$)

JOR	Data		set	1	Data		set	2	Data		set	3
N	100	200	300	400	100	200	300	400	100	200	300	400
$k = 2$	41	56	42	42	57	95	58	70	31	26	32	25
$k = 3$	56	60	42	42	56	75	57	61	31	35	43	25
$k = 4$	46	59	42	42	55	72	58	62	31	32	38	25
$k = 5$	56	60	43	43	56	68	57	60	32	30	36	26

SOR	Data		set	1	Data		set	2	Data		set	3
N	100	200	300	400	100	200	300	400	100	200	300	400
$k = 2$	20	18	17	17	16	15	16	15	18	14	19	15
$k = 3$	30	27	17	25	16	23	16	23	18	21	29	15
$k = 4$	25	24	19	22	17	21	16	21	18	19	26	18
$k = 5$	30	28	19	23	17	21	16	20	20	20	25	17

Table 2.2 Number of iterations for convergence ($\alpha = 0.85$)

JOR	Data		set	1	Data		set	2	Data		set	3
N	100	200	300	400	100	200	300	400	100	200	300	400
$k = 2$	42	56	44	47	61	82	66	64	18	28	32	26
$k = 3$	55	60	45	52	62	81	63	62	18	36	42	26
$k = 4$	53	59	45	49	58	71	62	62	18	33	38	26
$k = 5$	53	65	45	49	61	70	64	62	18	32	37	26

SOR	Data		set	1	Data		set	2	Data		set	3
N	100	200	300	400	100	200	300	400	100	200	300	400
$k = 2$	19	17	17	16	16	14	15	15	15	14	19	16
$k = 3$	28	26	17	24	16	22	15	23	15	23	29	16
$k = 4$	24	23	19	21	16	20	16	21	17	20	25	16
$k = 5$	28	26	19	21	17	21	16	20	16	20	23	16

Here I is the identity matrix and $\epsilon > 0$ can be chosen as small as desired. Then the matrix $(A + \epsilon I)$ is strictly diagonal dominant, although this will introduce a small error of $O(\epsilon)$ to the linear system. Numerical results in Yuen et al. [213,214] indicate that the hybrid algorithm is very efficient in solving for the steady-state probability distribution of queueing systems and ranking webpages in the Web. Here we present some small scale numerical results (three different data sets) for two typical values of α in Tables 2.1 and 2.2 (Taken from [214]). Here k is the size of population and N is the number of webpages.

2.3 Summary

In this chapter, we discussed two important applications of Markov chains: the classical Markovian queueing networks and the Modern PageRank algorithm. The latter application comes from the measurement of prestige in a network. The computation of prestige in a network is an important issue [22, 23] and has many other applications, such as social networks [206] and disease transmission [12]. A number of methods based on the computation of eigenvectors have been proposed in the literature, see for instance Langville and Meyer [139, 140] and the references therein. Further research can be done in developing models and algorithms for the case when there are negative relations in the network [192]. In a network, being chosen or nominated by a popular or powerful person (webpage) would add to one's popularity. Instead of supporting a member, a negative relation means being rejected by a member in the network.

2.4 Exercise

1. We consider an M/M/1/∞ queue. The mean service time of the exponential server is μ^{-1} and the customers arrive according to a Poisson process with mean rate λ where $\lambda < \mu$. An arrived customer may leave the system directly with a probability of

$$\frac{i}{i+1} \quad i = 0, 1, \ldots,$$

when they find there are already i customers in the system.

(a) Find the steady-steady probability distribution of the queueing system in terms of λ and μ.
(b) Assuming that the queueing system is in steady-state, find the loss probability (probability that an arrived customer chooses not to join the queueing system).
(c) If the mean service time μ^{-1} can be assigned as fast as possible (i.e. $\lambda << \mu$), recommend the service rate μ if the loss probability has to be less than 1 %.

2. Consider an exponential server system with service rate μ having no waiting space. Customers arrive according to a Poisson process with mean rate λ. As the server is failure-prone, the normal time of the server follows the exponential distribution with mean θ^{-1}. When it breaks down, it will be repaired at once. The repair time of the server follows an exponential distribution with mean γ^{-1}. The following are the four possible states of the system

$$\{(N, 0), (D, 0), (N, 1), (D, 1)\}$$

where N represents the server is normal, D represents the server is down, 0 represents the system is idle and 1 represents the system is busy.

(a) Write down the generator matrix of this Markov process.
(b) Obtain the steady-state probability distribution of the system.

(c) Assuming that the system is in steady-state, find the conditional probability that the server is under repair given that the system is idle.

3. (a) We consider a system of n identical failure-prone machines. A repairman is assigned to repair the broken machines. The mean normal time and the repair time of a machine are exponentially distributed with means λ^{-1} and μ^{-1} respectively. The system is said to be in state $i \, (i = 0, 1, \cdots, n)$ if there are i broken machines.
   (a) Write down the Markov chain and the generator matrix for the states of this machine repairing model.
   (b) Find the steady-state probability p_i that there are i broken machines in the system.
   (c) Find the steady-state probability that the repairman is idle.
4. A company is going to hire a repairman to look after four identical unreliable machines. The machines break down randomly according to a Poisson process of mean rate of 3 per day. The non-productive time of any machine costs the firm 200 dollars per day. The firm can hire a slow, cheap repairman charging 500 dollars per day who repairs at an average rate of 4 machines per day. Alternatively the firm can hire a fast, experienced repairman charging $1,000$ dollars per day who repairs at an average rate of 5 machines per day. In either case repair time is assumed to be exponentially distributed. Compute the expected running costs in both cases and determine which repairman should be hired.
5. Consider the allocation problem discussed in Sect. 2.1.3.

   (a) Apply the result in (2.4) to show that the expected number of customers waiting in Queue i is given by

   $$\frac{(\lambda_i/\mu_i)^2}{1 - \lambda_i/\mu_i}.$$

   Hence show that the total expected number of customers waiting in the system is given by

   $$\sum_{i=1}^{n} \frac{(\lambda_i/\mu_i)^2}{1 - \lambda_i/\mu_i}.$$

   (b) Solve the following optimization problem:

   $$\min_{\lambda_1,\ldots,\lambda_n} \left\{ \sum_{i=1}^{n} \frac{(\lambda_i/\mu_i)^2}{1 - \lambda_i/\mu_i} \right\}.$$

   subject to

   $$\sum_{i=1}^{m} \lambda_i = \lambda$$

and

$$0 \le \lambda_i < \mu \quad \text{for} \quad i = 1, 2, \ldots, n$$

and get the optimal allocation.

6. Prove the formula (2.7).
7. In a post office, postmen must retrieve their assignments from the office which is managed by an assistant. Suppose the mean number of postmen retrieving their assignments per hour is 4 and they are paid 10 dollars per hour. There are two possible assistants A or B to be employed for managing the office. On average assistant A takes 11.5 min to handle one request and is paid 9.5 dollars per hour. While assistant B takes 12.5 min to handle one request and is paid 7 dollars per hour. Assume that the inter-arrival time of postmen and the processing time of the assistants are exponentially distributed. Which assistant should be employed?
8. Let

$$P = \begin{pmatrix} 0.75 & 0.25 & 0 & 0 \\ 0.2 & 0.5 & 0.2 & 0.1 \\ 0.3 & 0.25 & 0.25 & 0.2 \\ 0 & 0.15 & 0.25 & 0.6 \end{pmatrix}.$$

Apply the power method for 10 iterations to get an approximate of the steady-state probability distribution.

9. Apply the power method to the following $n \times n$ transition probability matrix

$$P = \begin{pmatrix} 0.75 & 0.25 & 0 & \cdots & \cdots & 0 \\ 0.25 & 0.50 & 0.25 & \ddots & & \vdots \\ 0 & 0.25 & 0.50 & 0.25 & \ddots & \vdots \\ 0 & & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & 0.25 & 0.50 & 0.25 \\ 0 & \cdots & \cdots & 0 & 0.25 & 0.75 \end{pmatrix} \tag{2.19}$$

to get an approximation of the steady-state probability distribution for $n = 10, 20, 40, 80, 160, 320$. Record the number of iterations required for the convergence under the following stopping criterion

$$||\mathbf{x}_k - \mathbf{p}||_1 < 10^{-6}.$$

Here $\mathbf{p}$ is the true solution and $\mathbf{x}_k$ is the approximate solution obtained in kth iteration. We remark that the true solution of the steady-state probability distribution is given by

$$\mathbf{p} = \frac{1}{n}(1, 1, \cdots, 1)^T.$$

10. Consider a general PageRank matrix

$$P = (cQ + (1 - c)\mathbf{1}\mathbf{u}^T) \quad \text{for} \quad c \in [0, 1)$$

where **u** is a positive probability distribution vector.

(a) By using Sherman-Morrison-Woodbury's formula, show that the stationary distribution is given by

$$\mathbf{x} = (1 - c)(I - cP)^{-1}\mathbf{u}.$$

(b) Hence show that

$$\mathbf{x} = (1 - c)\sum_{n=0}^{\infty} c^n (P^n \mathbf{u}).$$

Chapter 3
Manufacturing and Re-manufacturing Systems

3.1 Introduction

In this chapter, we consider the application of the Markovian queueing systems discussed in Chap. 2 in modeling manufacturing systems and re-manufacturing systems. We adopt *Hedging Point Production* (HPP) policy as a production control policy. We note that in a queueing system, there are servers, customers, and waiting spaces. To model a *make-to-order* manufacturing system by a queueing system, one may regard a server as a machine. The customers can be regarded as the inventory of product or the jobs to be processed respectively; see for instance Buzacott and Shanthikumar [34]. In a *manufacturing* system, a certain amount of inventory (called the hedging point) is kept to cope with the fluctuation of demand and therefore production control is necessary. The system will stop production when this level of inventory is attained.

We then discuss the inventory controls of demands and returns of single-item inventory systems [164]. In fact, there are many research papers on inventory control of repairable items and returns, most of them describing the system as a closed-loop queueing network with a constant number of items inside [75, 137, 165, 195]. Disposal of returns [131, 136, 178] is allowed in the models presented here. The justification for disposal is that accepting all returns will lead to extremely high inventory level and hence very high inventory cost. Sometimes transshipment of returns is allowed among the inventory systems to reduce the rejection rate of returns. Other re-manufacturing models can be found in [122, 136, 138, 194] and good reviews and current advances of the related topics can be found in [21, 80, 99, 133, 163].

As a modern marketing strategy to encourage the customers to buy products, the customers are allowed to return the bought product and obtain a full refund within a period of one week. As a result, many customers may take advantage of this policy and the manufacturers have to handle many such returns. Very often, the returns are still in good condition, and can be put back to the market after checking and packaging. The first model we introduce here attempts to model this situation.

W.-K. Ching et al., *Markov Chains*, International Series in Operations Research
& Management Science 189, DOI 10.1007/978-1-4614-6312-2_3,

The model is a single-item inventory system for handling returns captured by a queueing network. In this model, the demands and the returns are assumed to follow two independent Poisson processes. The returns are tested and repaired according to the standard requirements. Repaired returns will be put into the serviceable inventory and non-repairable returns will be disposed of. The repairing time is assumed to be negligible. A similar inventory model with returns has been discussed in [118]. However, the model in [118] includes neither the replenishment costs nor the transshipment of returns. In this model, the inventory system is controlled by a popular (r, Q) continuous review policy. The inventory level of the serviceable product is modelled as an irreducible continuous time Markov chain. The generator matrix for the model is given and a closed form solution for the system steady-state probability distribution is also derived.

Next, two independent and identical inventory systems are considered and transshipment of returns from one inventory system to another is allowed. The joint inventory levels of the serviceable product is modeled as a two-dimensional irreducible continuous time Markov chain. The generator matrix for this advanced model is given and a closed form approximation of the solution of the system steady-state probability distribution is derived. Analysis of the average running cost of the joint inventory system can be carried out by using the approximated probability distribution. The focus is on the inventory cost and the replenishment cost of the system because the replenishment lead time is assumed to be zero and there is no backlog or loss of demand. It is shown that in the transshipment model, the rejection rate of the returns is extremely small and decreases significantly when the re-order size $(Q + 1)$ is large. The model is then extended to multiple inventory/return systems with a single depot. This kind of model is of particular interest when the re-manufacturer has several recycling locations. Since the locations can be easily connected by an information network, excessive returns can be forwarded to nearby locations or to the main depot directly. This will greatly cut down the disposal rate. The handling of used machines in IBM (a big recovery network) serves as a good example for the application of this model [99]. More examples and related models can be found in [99, pp. 106–131].

Finally, a hybrid system consisting of a re-manufacturing process and a manufacturing process is discussed. The hybrid system captures the re-manufacturing process and the system can produce serviceable product when the return rate is zero.

The remainder of this chapter is organized as follows. In Sect. 3.2, we discuss some Markovian queueing models for manufacturing systems. In Sect. 3.3, a single-item inventory model for handling returns is presented. In Sect. 3.4, the model is extended to the case that lateral transshipment of returns is allowed among the inventory systems. In Sect. 3.5, we discuss a hybrid re-manufacturing system. Finally, concluding remarks are given in Sect. 3.6.

3.2 Manufacturing Systems

In this section, we study some Markovian queueing models for manufacturing systems. We first consider reliable machine manufacturing systems and then discuss failure-prone machine manufacturing systems.

3.2.1 Reliable Machine Manufacturing Systems

We first consider a Markovian model of a one-machine manufacturing system. We then study the case of two machines in tandem. In both cases, the machines are reliable.

3.2.1.1 One-Machine Manufacturing System

In the one-machine manufacturing system, the production time for one unit of product is exponentially distributed with a mean time of μ^{-1}. The inter-arrival time of demand is also exponentially distributed with a mean time of λ^{-1}. The demand is served in a first come first serve manner. In order to retain the customers, there is no backlog limit in the system. However, there is an upper limit $n (n \geq 0)$ for the inventory level. The machine keeps on producing until this inventory level is reached and the production is stopped once this level is attained. We therefore seek the optimal value of n, called the hedging point or the *safety stock*, which minimizes the expected running cost. The running cost consists of a deterministic inventory cost and a backlog cost. In fact, the optimal value of n is the best amount of inventory to be kept in the system so as to hedge against the fluctuations in demand. The notation is summarized as follows (Fig. 3.1):

(a) I, the unit inventory cost
(b) B, the unit backlog cost
(c) $n \geq 0$, the hedging point
(d) μ^{-1}, the mean production time for one unit of product
(e) λ^{-1}, the mean inter-arrival time of demand

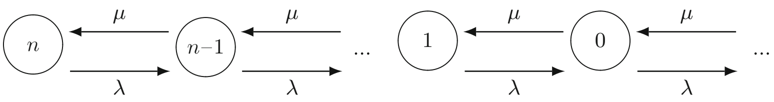


Fig. 3.1 The Markov Chain (M/M/1 Queue) for the manufacturing system

If the inventory level (negative inventory level means backlog) is used to represent the state of the system, one may write down the Markov chain for the system as follows. Here we assume that $\mu > \lambda$, so that the steady-state probability distribution of the above M/M/1 queue exists and has an analytic solution

$$q(i) = (1-p)p^{n-i}, \quad i = n, n-1, n-2, \cdots$$

where $p = \lambda/\mu$ and $q(i)$ is the steady-state probability that the inventory level is i. Hence it is straightforward to write down the expected running cost of the system (sum of the inventory cost and the backlog cost) when the hedging point is n as follows:

$$E(n) = I\left(\sum_{i=0}^{n}(n-i)(1-p)p^i\right) + B\left(\sum_{i=n+1}^{\infty}(i-n)(1-p)p^i\right).$$

The first term is the expected inventory cost and the second term is the expected backlog cost. The following proposition gives the optimal hedging point h.

Proposition 3.1. *The expected running cost $E(n)$ is minimized if the hedging point n is chosen such that*

$$p^{n+1} \le \frac{I}{I+B} \le p^n.$$

3.2.1.2 Two-Machine Manufacturing System

We study a two-stage manufacturing system [43]. The manufacturing system is composed of two reliable machines (no break down) producing one type of product. Every product has to go through the manufacturing process in both stages. Here it is assumed that an infinite supply of raw material is available for the manufacturing process at the first machine. Again the mean processing time for one unit of product in the first and second machine are exponentially distributed with parameters μ_1^{-1} and μ_2^{-1} respectively. Two buffers are placed immediately after the machines. Buffer B_1 of size b_1 is placed between the two machines to store the partially finished products. Final products are then stored in Buffer B_2 which has a maximum size of b_2. The demand is assumed to follow a Poisson process of mean rate λ. A finite backlog of finished product is allowed in the system. The maximum allowable backlog of product is m. When the inventory level of the finished product is $-m$, any arrival demand will be rejected. Hedging Point Production (HPP) policy is employed as the inventory control in both buffers B_1 and B_2. For the first machine, the hedging point is b_1 and the inventory level is non-negative. For the second machine, the hedging point is h; however, the inventory level of buffer B_2 can be negative, because we allow a maximum backlog of m. The machine-inventory system is modeled as a Markov chain problem. It turns out that the process is an irreducible continuous time Markov chain. We now give the generator matrix for the process.

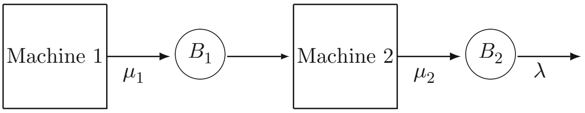


Fig. 3.2 A two-machine manufacturing system

We note that the inventory level of the first buffer cannot be negative or exceed the buffer size b_1. Thus the total number of inventory levels in the first buffer is $b_1 + 1$. For the second buffer, under the HPP policy, the maximum possible inventory level is $h(h \leq b_2)$ (Fig. 3.2).

If we let $I_1(t)$ and $I_2(t)$ be the inventory levels of the first and second buffer at time t respectively then $I_1(t)$ and $I_2(t)$ are going to take integral values in $[0, b_1]$ and $[-m, h]$ respectively. Thus the joint inventory process $\{(I_1(t), I_2(t)), t \geq 0\}$ can be shown to be a continuous time Markov chain process taking values in the state-space

$$S = \{(z_1(t), z_2(t)) : z_1 = 0, \ldots, b_1,\ z_2 = -m, \ldots, h.\}.$$

To obtain the steady-state probability distribution of the system, we order inventory states lexicographically, according to I_1 first and then I_2. One can then obtain the following generator matrix governing the steady-state probability distribution

$$A_1 = \begin{pmatrix} \Lambda + \mu_1 I_n & \Sigma & & & 0 \\ -\mu_1 I_n & \Lambda + D + \mu_1 I_n & \Sigma & & \\ & \ddots & \ddots & \ddots & \\ & & -\mu_1 I_n & \Lambda + D + \mu_1 I_n & \Sigma \\ 0 & & & -\mu_1 I_n & \Lambda + D \end{pmatrix}, \quad (3.1)$$

where

$$\Lambda = \begin{pmatrix} 0 & -\lambda & & 0 \\ & \lambda & \ddots & \\ & & \ddots & -\lambda \\ 0 & & & \lambda \end{pmatrix}, \quad \Sigma = \begin{pmatrix} 0 & & & 0 \\ -\mu_2 & \ddots & & \\ & \ddots & \ddots & \\ 0 & & -\mu_2 & 0 \end{pmatrix}, \qquad (3.2)$$

I_n is the $n \times n$ identity matrix and D is the $n \times n$ diagonal matrix

$$D = \text{Diag}(\mu_2, \ldots, \mu_2, 0). \qquad (3.3)$$

If we can get the steady-state probability distribution $\mathbf{p}$ of the generator matrix A_1, then many useful quantities such as the throughput of the system

$$\left(1-\sum_{j=-m}^{h} p(0,j)\right)\mu_2$$

and the mean number of products in buffers B_1 and B_2 (work-in-process)

$$\sum_{i=1}^{b_1}\left(\sum_{j=-m}^{h} p(i,j)\right) i$$

and

$$\sum_{j=1}^{h}\left(\sum_{i=0}^{b_1} p(i,j)\right) j$$

can be obtained. We remark that the generator A_1 is irreducible, has zero column sum, positive diagonal entries and non-positive off-diagonal entries, so that A_1 has a one-dimensional null space with a right positive null vector. The steady-state probability distribution $\mathbf{p}$ is then equal to the normalized form of the positive null vector. Similar to the previous chapters, we consider the following equivalent linear system

$$G_1\mathbf{x} \equiv (A_1 + \mathbf{e}_1\mathbf{e}_1^T)\mathbf{x} = \mathbf{e}_1, \tag{3.4}$$

where $\mathbf{e}_1 = (1,0,\ldots,0)^T$ is the $(b_1+1)(m+h+1)$ unit vector. Similarly one can show that the linear system (3.4) is non-singular and hence the steady-state probability distribution can be obtained by normalizing the solution of (3.4). We have the following lemma.

Lemma 3.2. *The matrix G_1 is non-singular.*

However, the closed form solution of $\mathbf{p}$ does not exist in this case. Thus classical iterative methods including Conjugate Gradient (CG) type methods can be applied to solve the problem. We remark that the idea of the Markovian model can be extended to the case of multiple machines in tandem. However, the computational cost for solving the steady-state probability distribution will increase exponentially with respect to the number of machines in tandem. A heuristic method based on Markovian approximation has been proposed in [47] to obtain an approximate solution.

3.2.1.3 Multiple Unreliable Machines Manufacturing System

Manufacturing systems of m multiple unreliable machines producing one type of product have been studied in [37, 46, 49]. Again the arrival of demand is assumed to be a Poisson process and the machines are unreliable; when a machine breaks down it is subject to an exponential repairing process. The normal time (functioning time) of a machine and the processing time of one unit of product are both exponentially distributed. Moreover there is an inventory cost for holding each unit of product and a shortfall cost for each unit of backlog. In [37,49], the production process of the machines is then a Markov Modulated Poisson Process (MMPP). Each machine has two states, either “normal operation” or “under repair”. Since there are m machines, there are 2^m states for the system of machines. The states of the machines and the inventory level can then be modeled as an irreducible continuous time Markov chain. For different values of the hedging point n, the average running cost can be written in terms of the steady-state probability distribution of the Markov chain. Therefore the optimal hedging point can be obtained by varying the value of n. In [49], numerical algorithm based on circulant preconditioning discussed in Chap. 2 has been designed to obtain the steady-state probability distribution efficiently. An extension to the case of batch arrival of demands can be found in [44].

3.3 An Inventory Model for Returns

In this section, a single-item inventory system is presented. The demands and returns of the product are assumed to follow two independent Poisson processes with mean rates λ and μ respectively. The maximum inventory capacity of the system is Q. When the inventory level is Q, any arrived return will be disposed of. A returned product is checked/repaired before being put into the serviceable inventory. Here it is assumed that only a stationary proportion, let us say $a \times 100\,\%$ of the returned product is repairable and a non-repairable return will be disposed of. The checking/repairing time of a returned product is assumed to be negligible. The notations for later discussions are listed below:

(a) λ^{-1}, the mean inter-arrival time of demands
(b) μ^{-1}, the mean inter-arrival time of returns
(c) a, the probability that a returned product is repairable
(d) Q, maximum inventory capacity
(e) I, unit inventory cost
(f) R, cost per replenishment order

An (r, Q) inventory control policy is employed as inventory control. Here, the lead time of a replenishment is assumed to be negligible. For simplicity of discussion, here we assume that $r = 0$. In a traditional $(0, Q)$ inventory control policy, a replenishment size of Q is placed whenever the inventory level is 0. Here, we assume that there is no loss of demand in our model. A replenishment order of size $(Q + 1)$ is placed when the inventory level is 0 and there is an arrived demand. This will then clear the arrived demand and bring the inventory level up to Q, see Fig. 3.3 (Taken from [72]). In fact, State ‘-1’ does not exist in the Markov chain, see Fig. 3.4 (Taken from [72]) for instance.

The states of the Markov chain are ordered according to the inventory levels in ascending order and give the following Markov chain.

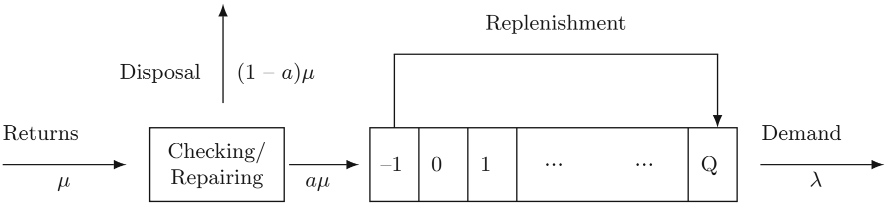


Fig. 3.3 The single-item inventory model

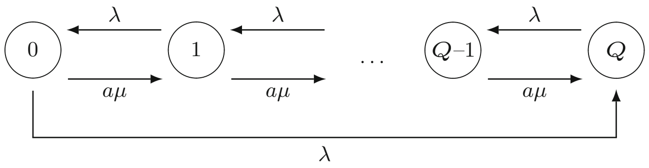


Fig. 3.4 The Markov chain

The $(Q+1)\times(Q+1)$ system generator matrix is given as follows:

$$A_2 = \begin{array}{c} 0\\ 1\\ \vdots\\ \vdots\\ Q \end{array} \left(\begin{array}{ccccc} \lambda+a\mu & -\lambda & & & 0\\ -a\mu & \lambda+a\mu & -\lambda & & \\ & \ddots & \ddots & \ddots & \\ & & -a\mu & \lambda+a\mu & -\lambda\\ -\lambda & & & -a\mu & \lambda \end{array}\right). \qquad (3.5)$$

The steady-state probability distribution **p** of the system satisfies

$$A_2\mathbf{p} = \mathbf{0} \quad \text{and} \quad \mathbf{1}^T\mathbf{p} = 1. \qquad (3.6)$$

By direct verification the following proposition can be obtained.

Proposition 3.3. *The steady-state probability distribution* **p** *is given by*

$$p_i = K(1-\rho^{i+1}), i = 0, 1, \ldots, Q \qquad (3.7)$$

where

$$\rho = \frac{a\mu}{\lambda}$$

and

$$K = \frac{1-\rho}{(1+Q)(1-\rho)-\rho(1-\rho^{Q+1})}.$$

By using the result of the steady-state probability in Proposition 3.3, the following corollary is obtained.

Corollary 3.4. *The expected inventory level is*

$$\sum_{i=1}^{Q} i p_i = \sum_{i=1}^{Q} K(i - i\rho^{i+1}) = K\left(\frac{Q(Q+1)}{2} + \frac{Q\rho^{Q+2}}{1-\rho} - \frac{\rho^2(1-\rho^Q)}{(1-\rho)^2}\right),$$

the average rejection rate of returns is

$$\mu p_Q = \mu K(1 - \rho^{Q+1})$$

and the mean replenishment rate is

$$\frac{\lambda p_0 \lambda^{-1}}{\lambda^{-1} + (a\mu)^{-1}} = \frac{\lambda K(1-\rho)\rho}{(1+\rho)}.$$

Proposition 3.5. *If $\rho < 1$ and Q is large then*

$$K \approx (1+Q)^{-1}$$

and the approximated average running cost (inventory and replenishment cost) is

$$C(Q) \approx \frac{QI}{2} + \frac{\lambda(1-\rho)\rho R}{(1+\rho)(1+Q)}.$$

The optimal replenishment size is

$$Q^* + 1 \approx \sqrt{\frac{2\lambda(1-\rho)\rho R}{(1+\rho)I}} = \sqrt{\frac{2a\mu R}{I}\left(\frac{2\lambda}{\lambda + a\mu} - 1\right)}. \tag{3.8}$$

One can observe that the optimal replenishment size Q^* either λ or R increases or I decreases. We end this section with the following remarks:

1. The model can be extended to the multi-item case when there is no limit in the inventory capacity. The trick is to use independent queueing networks to model individual products. Suppose there are s different products and their demand rates, return rates, unit inventory costs, cost per replenishment order and the probability of getting a repairable return are given by $\lambda_i, \mu_i, I_i, R_i$ and a_i respectively. Then the optimal replenishment size of each product i will be given by (3.8)

$$Q_i^* + 1 \approx \sqrt{\frac{2a_i\mu_i R_i}{I_i}\left(\frac{2\lambda_i}{\lambda_i + a_i\mu_i} - 1\right)} \qquad \text{for} \quad i = 1, 2, \ldots, s.$$

2. To include the inventory capacity in the system, one can have approximations for the steady-state probability distributions for the inventory levels of the returns

and the serviceable product by assuming that capacity for storing returns is large. Then the inventory levels of the returns forms an M/M/1 queue and the output process of an M/M/1 queue in steady-state is again a Poisson process with the same mean rate, see the lemma below.

Lemma 3.6. *The output process of an M/M/*1 *queue in steady-state is again a Poisson process with the same mean as the input rate.*

Proof. We first note that if X and Y are two independent exponential random variables with means λ^{-1} and μ^{-1} respectively. Then the probability density function for the random variable $Z = X + Y$ is given by

$$f(z) = \frac{\lambda\mu}{\mu - \lambda} e^{-\lambda z} - \frac{\lambda\mu}{\mu - \lambda} e^{-\mu z}.$$

Let the arrival rate of the M/M/1 queue be λ and the service rate of the server be μ. There are two cases to be considered: the server is idle (the steady-state probability is $(1 - \lambda/\mu)$ (see Chap. 2) and the server is not idle (the steady-state probability is λ/μ.)

For the former case, the departure time follows $f(z)$ (a waiting time for an arrival plus a service time). For the latter case, the departure time follows $\mu e^{-\mu z}$. Thus the probability density function $g(z)$ for the departure time is given by

$$\begin{aligned}(1 - \frac{\lambda}{\mu}) f(z) + \frac{\lambda}{\mu}(\mu e^{-\mu z}) &= \frac{\lambda\mu}{\mu - \lambda} e^{-\lambda z} - \frac{\lambda\mu}{\mu - \lambda} e^{-\mu z} \\ &\quad - \frac{\lambda^2}{\mu - \lambda} e^{-\lambda z} + \frac{\lambda^2}{\mu - \lambda} e^{-\mu z} + \lambda e^{-\mu z}.\end{aligned}$$

Thus

$$g(z) = \lambda e^{-\lambda z}$$

is the exponential distribution. This implies that the departure process is a Poisson process. From Proposition 1.37 it is evident that the departure process is a Poisson process with mean λ if and only if the inter-departure time follows the exponential distribution with mean λ^{-1}.

3. One can also take into account the lead time of a replenishment and the checking/repairing time of a return. In this case, it becomes a tandem queueing network and the analytic solution for the system steady-state probability distribution is not available in general. A numerical method based on a preconditioned conjugate gradient method has been applied to solve this type of tandem queueing system, see for instance [36, 37, 41, 43, 46, 49].

3.4 The Lateral Transshipment Model

In this section, an inventory model which consists of two independent inventory systems as described in the previous section is considered. For simplicity of discussion, both of them are assumed to be identical. A special feature of this model is that lateral transshipment of returns between the inventory systems is allowed. Lateral transshipment of demands has been studied in a number of papers [9,42,72]. Substantial savings can be realized by the sharing of inventory via the lateral transshipment of demands [180]. Here, this concept is extended to the handling of returns. Recall that an arrived return will be disposed of if the inventory level is Q in the previous model. In the new model, lateral transshipment of returns between the inventory systems is allowed whenever one of them is full (whenever the inventory level is Q) and the other is not yet full (the inventory level is less than Q). Denote $x(t)$ and $y(t)$ to be the inventory levels of the serviceable product in the first and the second inventory system at time t respectively. Then, the random variables $x(t)$ and $y(t)$ take integral values in [0, Q]. Thus, the joint inventory process

$$\{(x(t), y(t)),\ t \geq 0\}$$

is again a continuous time Markov chain taking values in the state space

$$S = \{(x, y) : x = 0, \cdots, Q, \quad y = 0, \cdots, Q.\}.$$

The inventory states are ordered lexicographically, according to x first, and then y. The generator matrix of the joint inventory system can be written by using the Kronecker tensor product as follows:

$$A_3 = I_{Q+1} \otimes A_2 + A_2 \otimes I_{Q+1} + \varDelta \otimes \varLambda + \varLambda \otimes \varDelta \tag{3.9}$$

where

$$\varLambda = \begin{pmatrix} 1 & & & & 0 \\ -1 & 1 & & & \\ & \ddots & \ddots & & \\ & & -1 & 1 & \\ 0 & & & -1 & 0 \end{pmatrix} \tag{3.10}$$

and

$$\varDelta = \begin{pmatrix} 0 & & & & 0 \\ & 0 & & & \\ & & \ddots & & \\ & & & 0 & \\ 0 & & & & a\mu \end{pmatrix} \tag{3.11}$$

and I_{Q+1} is the $(Q+1) \times (Q+1)$ identity matrix. The steady-state probability vector $\mathbf{q}$ satisfies

$$A_3\mathbf{q} = \mathbf{0} \quad \text{and} \quad \mathbf{1}^T\mathbf{q} = 1. \tag{3.12}$$

We note that the generator A_3 is irreducible and it has a one-dimensional null-space with a right positive null vector, see [108, 203]. The steady-state probability vector $\mathbf{q}$ is the normalized form of the positive null vector of A_3. Let q_{ij} be the steady-state probability that the inventory level of the serviceable product is i in the first inventory system and j in the second inventory system. Many important quantities of the system performance can be written in terms of q_{ij}. For example, the return rejection probability is q_{QQ}. Unfortunately, a closed form solution of $\mathbf{q}$ is not generally available. By making use of the block structure of the generator matrix B, classical iterative methods such as the Block Gauss-Seidel (BGS) method can be applied to solve the steady-state probability distribution [43, 108, 203]. In the following, instead of solving the steady-state probability distribution numerically, a closed form approximation for the probability distribution $\mathbf{q}$ is derived under some assumptions.

Proposition 3.7. *Let* $\mathbf{p}$ *be the steady-state probability distribution for the generator matrix* A *in Proposition 3.3. If* $\rho < 1$ *then*

$$||A_3(\mathbf{p} \otimes \mathbf{p})||_\infty \le \frac{4a\mu}{(Q+1)^2(1-\rho)^2}.$$

The probability vector $\mathbf{q} = \mathbf{p} \otimes \mathbf{p}$ *is an approximation of the steady-state probability vector when* Q *is large.*

Proof. The probability vector $\mathbf{p}$ is just the solution of (3.6). By direct verification, we have $\mathbf{1}^T(\mathbf{p} \otimes \mathbf{p}) = 1$ and

$$(I \otimes A_2 + A_2 \otimes I)(\mathbf{p} \otimes \mathbf{p}) = (\mathbf{p} \otimes A_2\mathbf{p} + A_2\mathbf{p} \otimes \mathbf{p}) = (\mathbf{p} \otimes \mathbf{0} + \mathbf{0} \otimes \mathbf{p}) = \mathbf{0}.$$

Therefore from (3.9)

$$A_3(\mathbf{p} \otimes \mathbf{p}) = (\Lambda \otimes \Delta)(\mathbf{p} \otimes \mathbf{p}) + (\Delta \otimes \Lambda)(\mathbf{p} \otimes \mathbf{p}) = (\Lambda\mathbf{p} \otimes \Delta\mathbf{p}) + (\Delta\mathbf{p} \otimes \Lambda\mathbf{p}).$$

One could observe that

$$||\Lambda||_{M_\infty} = 2, \quad ||\Delta||_{M_\infty} = a\mu \quad \text{and} \quad ||\mathbf{p}||_\infty \le K.$$

Therefore,

$$\begin{aligned} ||A_3(\mathbf{p} \otimes \mathbf{p})||_\infty &\le 2||\Lambda||_{M_\infty}||\mathbf{p}||_\infty||\Delta||_{M_\infty}||\mathbf{p}||_\infty \\ &= 4a\mu K^2 \\ &\le \frac{4a\mu}{(Q+1)^2(1-\rho)^2}. \end{aligned} \tag{3.13}$$

if we adopt $\mathbf{q} = \mathbf{p} \otimes \mathbf{p}$ to be the system steady-state probability distribution, then the approximated optimal replenishment size of each inventory system is the same as in Proposition 3.5. By allowing transshipment of returns, the rejection rate of returns of the two inventory systems will be decreased from

$$2\mu K(1-\rho^{Q+1}) \approx \frac{2\mu}{Q+1}$$

to

$$\mu K^2(1-\rho^{Q+1})^2 \approx \frac{\mu}{(Q+1)^2}.$$

Note that the approximation is valid only if Q is large, and the error is of order $O(Q^{-2})$.

3.5 The Hybrid Re-manufacturing System

In this section, we propose a hybrid system. This is a system that consists of a re-manufacturing process and a manufacturing process. The proposed hybrid system captures the re-manufacturing process and can produce serviceable product when the return rate is zero. The demands and returns are assumed to follow independent Poisson processes. The serviceable product inventory level and the outside procurements are controlled by a popular (r, Q) continuous review policy. The inventory level of the serviceable product is modelled as an irreducible continuous time Markov chain and the generator matrix is constructed. It is found that the generator matrix has a near-Toeplitz structure.

A direct method is then proposed for solving steady-state probabilities. The direct method is based on Fast Fourier Transforms (FFTs) and the Sherman-Morrison-Woodbury Formula (Proposition 1.38). The complexity of the method is then given and some special cases are also discussed.

3.5.1 The Hybrid System

In this subsection, an inventory model which captures the re-manufacturing process is proposed. Disposal of returned product is allowed when the return capacity is full. In the model, there are two types of inventory to be managed, the serviceable product and the returned product. The demands and the returns are assumed to follow independent Poisson processes with mean rates λ and γ respectively. The re-manufacturing process is then modelled by an M/M/1/N queue: a returned product acts as a customer and a reliable re-manufacturing machine (with processing rate μ) acts as the server in the queue. The re-manufacturing process is stopped whenever there is no space for placing the serviceable product (i.e. when the

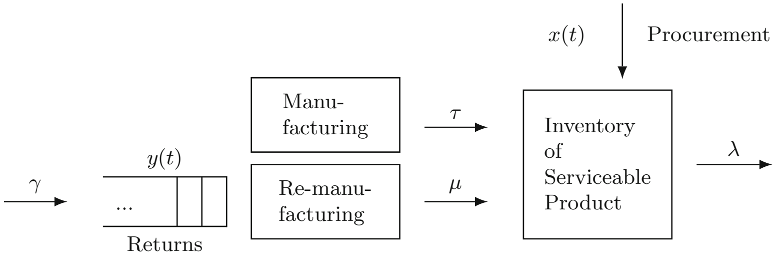


Fig. 3.5 The hybrid system

serviceable product inventory level is Q). Here we also assume that when the return level is zero, the system can produce at a rate of τ (exponentially distributed).

The serviceable product inventory level and the outside procurements are controlled by a popular (r, Q) continuous review policy. This means that when the inventory level drops to r, an outside procurement order of size $(Q - r)$ is placed and arrives immediately. For simplicity of discussion, the procurement level r is assumed to be -1. This means that whenever there is no serviceable product in the system and there is an arrival of demand then a procurement order of size $(Q + 1)$ is placed and arrives immediately. Therefore the procurement can clear the backlogged demand and bring the serviceable product inventory to Q. We also assume that it is always possible to purchase the required procurement. The inventory levels of both the returns and the serviceable product are modelled as Markovian processes. The capacity N for the returns and the capacity Q for serviceable product Q are assumed to be large. Figure 3.5 (Taken from [65, 67]) gives the framework of the re-manufacturing system.

3.5.2 The Generator Matrix of the System

In this subsection, the generator matrix for the re-manufacturing system is constructed. Let $x(t)$ and $y(t)$ be the inventory levels of the returns and the inventory levels of the serviceable products at time t respectively. Then $x(t)$ and $y(t)$ take integral values in $[0, N]$ and $[0, Q]$ respectively. The joint inventory process

$$\{(x(t), y(t)),\ t \geq 0\}$$

is a continuous time Markov chain taking values in the state space

$$S = \{(x, y) : x = 0, \cdots, N, \quad y = 0, \cdots, Q\}.$$

By ordering the joint inventory states lexicographically, according to x first and then y, the generator matrix for the joint inventory system can be written as follows:

$$A_4 = \begin{pmatrix} B_0 & -U & & & 0 \\ -\gamma I_{Q+1} & B & -U & & \\ & \ddots & \ddots & \ddots & \\ & & -\gamma I_{Q+1} & B & -U \\ 0 & & & -\gamma I_{Q+1} & B_N \end{pmatrix}, \tag{3.14}$$

where

$$U = \begin{pmatrix} 0 & & & 0 \\ \mu & 0 & & \\ & \ddots & \ddots & \\ & & \ddots & \ddots \\ 0 & & \mu & 0 \end{pmatrix}, \tag{3.15}$$

$$B_0 = \gamma I_{Q+1} + \begin{pmatrix} \tau+\lambda & -\lambda & & & 0 \\ -\tau & \tau+\lambda & -\lambda & & \\ & -\tau & \ddots & -\lambda & \\ & & \ddots & \tau+\lambda & -\lambda \\ -\lambda & & & -\tau & \lambda \end{pmatrix}, \tag{3.16}$$

$$B = \gamma I_{Q+1} + \begin{pmatrix} \lambda+\mu & -\lambda & & & 0 \\ & \lambda+\mu & -\lambda & & \\ & & \ddots & -\lambda & \\ & & & \lambda+\mu & -\lambda \\ -\lambda & & & & \lambda \end{pmatrix}, \tag{3.17}$$

$$B_N = B - \gamma I_{Q+1}.$$

Here, I_{Q+1} is the $(Q + 1) \times (Q + 1)$ identity matrix . The steady-state probability distribution $\mathbf{p}$ is required if one wants to obtain the performance of the system. Note that the generator A_4 is irreducible, and from the Perron and Frobenius theory [108] it is known that it has a 1-dimensional null-space with a right positive null vector. Hence, as mentioned in Sect. 3.2.1, one can consider an equivalent linear system instead.

$$G_4\mathbf{x} \equiv (A_4 + \mathbf{f}\mathbf{f}^{\mathbf{T}})\mathbf{x} = \mathbf{f}, \quad \text{where} \quad \mathbf{f} = (0, \ldots, 0, 1)^T. \tag{3.18}$$

Proposition 3.8. *The matrix G_4 is nonsingular.*

However, the closed form solution of $\mathbf{p}$ is not generally available. Iterative methods, such as the PCG methods, are efficient in solving the probability vector $\mathbf{p}$ when one of the parameters N and Q are fixed, see for instance [41,43,46,49]. However, when both Q and N are getting larger, the fast convergence rate of the PCG method cannot be guaranteed, especially when the smallest singular value tends to zero very fast [42, 47]. Other approximation methods for solving the problem can be found in [43]. In the following subsection, a direct method is proposed for solving (3.18).

3.5.3 The Direct Method

We consider taking circulant approximations to the matrix blocks in A_4. We define the following circulant matrices:

$$c(G) = \begin{pmatrix} c(\bar{B}_0) & -c(U) & & & \\ -\gamma I_{Q+1} & c(B) & -c(U) & & \\ & \ddots & \ddots & \ddots & \\ & & -\gamma I_{Q+1} & c(B) & -c(U) \\ & & & -\gamma I_{Q+1} & c(B_N) \end{pmatrix}, \tag{3.19}$$

where

$$c(U) = \begin{pmatrix} 0 & & & & \mu \\ \mu & 0 & & & \\ & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ 0 & & & \mu & 0 \end{pmatrix}, \tag{3.20}$$

$$c(\bar{B}_0) = \gamma I_{Q+1} + \begin{pmatrix} \tau+\lambda & -\lambda & & & -\tau \\ -\tau & \tau+\lambda & -\lambda & & \\ & \ddots & \ddots & -\lambda & \\ & & \ddots & \tau+\lambda & -\lambda \\ -\lambda & & & -\tau & \tau+\lambda \end{pmatrix}, \tag{3.21}$$

$$c(B) = \gamma I_{Q+1} + \begin{pmatrix} \lambda+\mu & -\lambda & & & 0 \\ & \lambda+\mu & -\lambda & & \\ & & \ddots & -\lambda & \\ & & & \lambda+\mu & -\lambda \\ -\lambda & & & & \lambda+\mu \end{pmatrix}, \tag{3.22}$$

$$c(B_N) = c(B) - \gamma I_{Q+1}. \tag{3.23}$$

We observe that

$$c(U) - U = \mu \mathbf{e}_1^T \mathbf{e}_{Q+1}, \quad c(\bar{B}_0) - \bar{B}_0 = -\tau \mathbf{e}_1^T \mathbf{e}_{Q+1},$$

$$c(B) - B = \mu \mathbf{e}_{Q+1}^T \mathbf{e}_{Q+1}, \quad \text{and} \quad c(B_N) - B_N = \mu \mathbf{e}_{Q+1}^T \mathbf{e}_{Q+1},$$

where

$$\mathbf{e}_1 = (1, 0, \ldots, 0) \quad \text{and} \quad \mathbf{e}_{Q+1} = (0, \ldots, 0, 1)$$

are 1-by-$(Q+1)$ unit vectors. Here we remark that

$$\bar{B}_0 = B_0 + \tau \mathbf{e}_{Q+1}^T \mathbf{e}_{Q+1}.$$

Therefore the matrix G is a sum of a circulant block matrix and another block matrix, with small rank except the first and the last diagonal blocks.

In view of the above formulation, the problem is equivalent to the solution of the linear system having the form $A_4\mathbf{z} = \mathbf{b}$ where A is a block-Toeplitz matrix given by

$$A_4 = \begin{pmatrix} A_{11} & \ldots\ldots & A_{1m} \\ A_{21} & \ldots\ldots & A_{2m} \\ \vdots & \vdots \; \vdots & \vdots \\ A_{m1} & \ldots\ldots & A_{mm} \end{pmatrix}. \tag{3.24}$$

Here

$$A_{ij} = C_{i-j} + \mathbf{u}_{i-j}^T \mathbf{v}, \tag{3.25}$$

where C_{i-j} is an $n \times n$ circulant matrix, and $\mathbf{u}_{i-j}$ and $\mathbf{v}$ are $k \times n$ matrices and k is much smaller than both m and n, so that A_{ij} is an $n \times n$ near-circulant matrix, i.e., with finite rank being less than or equal to k. We remark that the class of matrices A_4 is closely related to the generator matrices of many Markovian models,such as queueing systems [43, 141, 142], manufacturing systems [41, 43, 46, 49, 56], and re-manufacturing systems [72, 99, 137].

Next, we note that an $n \times n$ circulant matrix can be diagonalized by using the discrete Fourier matrix F_n. Moreover, its eigenvalues can be obtained in $O(n \log n)$

operations by using the FFT, see for instance Davis [79]. In view of this advantage, consider

$$
\begin{aligned}
(I_m \otimes F_n^*)A_4(I_m \otimes F_n) &= \begin{pmatrix} D_{11} & \dots & D_{1m} \\ D_{21} & \dots & D_{2m} \\ \vdots & \vdots & \vdots \\ D_{m1} & \dots & D_{mm} \end{pmatrix} + \begin{pmatrix} E_{11} & \dots & E_{1m} \\ E_{21} & \dots & E_{2m} \\ \vdots & \vdots & \vdots \\ E_{m1} & \dots & E_{mm} \end{pmatrix} \\
&\equiv D + E.
\end{aligned}
\qquad (3.26)
$$

Here D_{ij} is a diagonal matrix containing the eigenvalues of C_{i-j} and

$$
E_{ij} = (F_n^* \mathbf{u}_{i-j}^T)(\mathbf{v}F_n) \equiv (\mathbf{x}_{i-j}^T)(\mathbf{y}). \qquad (3.27)
$$

We note that

$$
\begin{aligned}
E &= \begin{pmatrix} \mathbf{x}_0^T\mathbf{y} & \dots\dots & \mathbf{x}_{1-m}^T\mathbf{y} \\ \mathbf{x}_1^T\mathbf{y} & \dots\dots & \mathbf{x}_{2-m}^T\mathbf{y} \\ \vdots & \vdots \quad \vdots & \vdots \\ \mathbf{x}_{m-1}^T\mathbf{y} & \dots\dots & \mathbf{x}_0^T\mathbf{y} \end{pmatrix} \\
&= \begin{pmatrix} \mathbf{x}_0^T & \dots & \mathbf{x}_{1-m}^T \\ \mathbf{x}_1^T & \dots & \mathbf{x}_{2-m}^T \\ \vdots & \vdots & \vdots \\ \mathbf{x}_{m-1}^T & \dots & \mathbf{x}_0^T \end{pmatrix} \begin{pmatrix} \mathbf{y} & 0 & \dots & 0 & 0 \\ 0 & \mathbf{y} & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & 0 & \mathbf{y} \end{pmatrix} \\
&\equiv XY.
\end{aligned}
\qquad (3.28)
$$

Moreover, we also note that D is still a block-Toeplitz matrix and there exists a permutation matrix P such that

$$
PDP^T = \operatorname{diag}(T_1, T_2, \dots, T_n) \qquad (3.29)
$$

where T_i is an $m \times m$ Toeplitz matrix. In fact direct methods for solving Toeplitz systems that are based on the recursion formula are in constant use, see for instance Trench [196]. For an $m \times m$ Toeplitz matrix T_i, these methods require $O(m^2)$ operations. Faster algorithms that require $O(m \log^2 m)$ operations have been developed for symmetric positive definite Toeplitz matrices, see Ammar and Gragg [4]. The stability properties of these direct methods are discussed in Bunch [32]. Hence by using direct methods, the linear system $D\mathbf{z} = \mathbf{b}$ can be solved in $O(nm^2)$ operations. The matrix X is an $mn \times mk$ matrix and the matrix Y is an $mk \times mn$ matrix.

To solve the linear system, we apply the Sherman-Morrison-Woodbury Formula (Proposition 1.38). The solution of $A_4\mathbf{z} = \mathbf{b}$ can be written as follows:

$$
\mathbf{z} = D^{-1}\mathbf{b} - D^{-1}X(I_{mk} + YD^{-1}X)^{-1}YD^{-1}\mathbf{b}. \qquad (3.30)
$$

3.5.4 The Computational Cost

In this section, the computational cost of the proposed method is discussed. The main computational cost of (3.30) consists of

(C0) FFT operations in (3.27)
(C1) Solving $\mathbf{r} = D^{-1}\mathbf{b}$
(C2) Solving $W = D^{-1}X$
(C3) Matrix multiplication of YW
(C4) Matrix multiplication of $Y\mathbf{r}$
(C5) Solving $(I_{mk} + YD^{-1}X)^{-1}\mathbf{r}$

The operational cost for (C0) is of $O(mn \log n)$. The operational cost for (C1) is at most $O(nm^2)$ operations by using direct solvers for Toeplitz systems. The cost for (C2) is at most $O(knm^3)$ operations in view of (C1). The operational cost for (C3) is of $O(k^2nm^2)$, because of the sparse structure of Y. The cost for (C4) is $O(knm)$ operations. Finally, the cost of (C5) is $O((km)^3)$ operations. Hence the overall cost will be $(km^3(n + k^2))$ operations.

In fact, the nice structure of D allows us to solve $D\mathbf{r} = \mathbf{b}$ in a parallel computer. Moreover $DW = X$ consists of n separate linear systems (a multiple right hand side problem). Again, this can also be solved in a parallel computer. Therefore the cost of (C1) and (C2) can be reduced by using parallel algorithms. Assuming that k is small, the costs of (C1) and (C2) can be reduced to $O(m^2)$ and $(O(m^3))$ operations respectively when n parallel processors are used.

3.5.5 Special Case Analysis

In this section, k is assumed to be small and some special cases of solving (3.30) are discussed.

Case (i): When all the $\mathbf{u}_{i-j}$ in (3.25) are equal, we see that all the columns of X are equal and the cost (C2) will be at most $O(nm^2)$ operations. Hence the overall cost will be $O(m^2(m + n) + mn \log n)$ operations.

Case (ii): If the matrix A_4 is a block-circulant matrix, then all the matrices T_i in (3.29) are circulant matrices. The cost of (C1) and (C2) can be reduced to $O(nm \log m)$ and $O(nm^2 \log m)$ operations respectively. Hence the overall cost will be $O(m^3 + nm(m \log m + \log n))$ operations.

Case (iii): If the matrix A_4 is a block tri-diagonal matrix, then all the matrices T_i in (3.29) are tri-diagonal matrices. The cost of (C0) will be $O(n \log n)$. The cost of (C1) and (C2) can be reduced to $O(nm)$ and $O(nm^2)$ operations respectively. Hence the overall cost will be $O(m^3 + n(m^2 + \log n))$ operations.

We end this section with the following proposition. The proposition gives the complexity for solving the steady-state probability distribution **p** for the generator matrix (3.14) when $Q \approx N$.

Proposition 3.9. *The steady-state probability distribution* **p** *can be obtained in* $O(N^3)$ *operations when* $Q \approx N$.

Proof. In the view of case (iii) in this section, the complexity of our method for solving (3.18) is $O(N^3)$ when $Q \approx N$, while the complexity of solving (3.18) by LU decomposition is $O(N^4)$.

3.6 Summary

In this chapter, we present the concept of re-manufacturing systems. Several stochastic models for re-manufacturing systems are discussed. The steady-state probability distributions of the models are either obtained in closed form or can be solved by fast numerical algorithms. The models here concern only single-items, and it will be interesting to extend the results to multi-item cases.

3.7 Exercises

1. Prove Proposition 3.1.
2. Prove Lemma 3.2.
3. Develop a computer program to solve (3.4) by Preconditioned Conjugate Gradient Method with a preconditioner, by taking the circulant approximation of A_1, i.e.,

$$c(\Lambda) = \begin{pmatrix} \lambda & -\lambda & & 0 \\ & \lambda & \ddots & \\ & & \ddots & -\lambda \\ -\lambda & & & \lambda \end{pmatrix}, \quad c(\Sigma) = \begin{pmatrix} 0 & & & -\mu_2 \\ -\mu_2 & \ddots & & \\ & \ddots & \ddots & \\ 0 & & -\mu_2 & 0 \end{pmatrix}$$

   and

$$c(D) = \text{Diag}(\mu_2, \ldots, \mu_2, \mu_2).$$

4. Prove Proposition 3.3.
5. Prove Proposition 3.5.

Chapter 4
A Hidden Markov Model for Customer Classification

4.1 Introduction

In this chapter, a new simple Hidden Markov Model (HMM) is proposed. The process of the proposed HMM can be explained by the following example.

4.1.1 A Simple Example

We consider the process of choosing a die of four faces (a tetrahedron) and recording the number of dots obtained by throwing the die [175]. Suppose we have two dice A and B, each of them has four faces $(1, 2, 3$ and $4)$. Moreover, Die A is fair and Die B is biased. The probability distributions of dots obtained by throwing dice A and B are given in Table 4.1.

Each time a die is to be chosen, we assume that with probability α, Die A is chosen, and with probability $(1-\alpha)$, Die B is chosen. This process is hidden as we don't know which die is chosen. The value of α is to be determined. The chosen die is then thrown and the number of dots (this is observable) obtained is recorded. The following is a possible realization of the whole process:

$$A \to 1 \to A \to 2 \to B \to 3 \to A \to 4 \to B \to 1 \to B \to 2 \to \cdots \to .$$

We note that the whole process of the HMM can be modeled by a classical Markov chain model with the transition probability matrix being given by

W.-K. Ching et al., *Markov Chains*, International Series in Operations Research & Management Science 189, DOI 10.1007/978-1-4614-6312-2_4,

Table 4.1 Probability distributions of Die A and Die B

Die	1	2	3	4
A	1/4	1/4	1/4	1/4
B	1/6	1/6	1/3	1/3

$$\begin{array}{c} A \\ B \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \left(\begin{array}{cc|cccc} 0 & 0 & \alpha & \alpha & \alpha & \alpha \\ 0 & 0 & 1-\alpha & 1-\alpha & 1-\alpha & 1-\alpha \\ \hline 1/4 & 1/6 & 0 & 0 & 0 & 0 \\ 1/4 & 1/6 & 0 & 0 & 0 & 0 \\ 1/4 & 1/3 & 0 & 0 & 0 & 0 \\ 1/4 & 1/3 & 0 & 0 & 0 & 0 \end{array} \right).$$

The rest of the chapter is organized as follows. In Sect. 4.2, the estimation method will be demonstrated by the example giving in Sect. 4.1. In Sect. 4.3, the proposed method is extended to a general case. In Sect. 4.4, some analytic results of a special case are presented. In Sect. 4.5, an application in customer classification, with practical data taken from a computer service company is presented and analyzed. Finally, a brief summary is given in Sect. 4.6 to conclude this chapter.

4.2 Parameter Estimation

In this section, we introduce a simple estimation method of α (Ching and Ng [58]). In order to define the HMM, one has to estimate α from an observed data sequence. If we suppose that the distribution of dots (in steady-state) is given by

$$\left(\frac{1}{6}, \frac{1}{4}, \frac{1}{4}, \frac{1}{3}\right)^T$$

then the question is: how to estimate α? We note that

$$P^2 = \left(\begin{array}{cc|cccc} \alpha & \alpha & 0 & 0 & 0 & 0 \\ 1-\alpha & 1-\alpha & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & \frac{1}{6}+\frac{\alpha}{12} & \frac{1}{6}+\frac{\alpha}{12} & \frac{1}{6}+\frac{\alpha}{12} & \frac{1}{6}+\frac{\alpha}{12} \\ 0 & 0 & \frac{1}{6}+\frac{\alpha}{12} & \frac{1}{6}+\frac{\alpha}{12} & \frac{1}{6}+\frac{\alpha}{12} & \frac{1}{6}+\frac{\alpha}{12} \\ 0 & 0 & \frac{1}{3}-\frac{\alpha}{12} & \frac{1}{3}-\frac{\alpha}{12} & \frac{1}{3}-\frac{\alpha}{12} & \frac{1}{3}-\frac{\alpha}{12} \\ 0 & 0 & \frac{1}{3}-\frac{\alpha}{12} & \frac{1}{3}-\frac{\alpha}{12} & \frac{1}{3}-\frac{\alpha}{12} & \frac{1}{3}-\frac{\alpha}{12} \end{array} \right) \equiv \begin{pmatrix} R & 0 \\ 0 & \tilde{P} \end{pmatrix}.$$

If we ignore the hidden states (the first diagonal block R), then the observable states follow the transition probability matrix given by the following matrix

$$\tilde{P} = \begin{pmatrix} \frac{1}{6}+\frac{\alpha}{12} & \frac{1}{6}+\frac{\alpha}{12} & \frac{1}{6}+\frac{\alpha}{12} & \frac{1}{6}+\frac{\alpha}{12} \\ \frac{1}{6}+\frac{\alpha}{12} & \frac{1}{6}+\frac{\alpha}{12} & \frac{1}{6}+\frac{\alpha}{12} & \frac{1}{6}+\frac{\alpha}{12} \\ \frac{1}{3}-\frac{\alpha}{12} & \frac{1}{3}-\frac{\alpha}{12} & \frac{1}{3}-\frac{\alpha}{12} & \frac{1}{3}-\frac{\alpha}{12} \\ \frac{1}{3}-\frac{\alpha}{12} & \frac{1}{3}-\frac{\alpha}{12} & \frac{1}{3}-\frac{\alpha}{12} & \frac{1}{3}-\frac{\alpha}{12} \end{pmatrix}$$
$$= \begin{pmatrix} \frac{1}{6}+\frac{\alpha}{12} \\ \frac{1}{6}+\frac{\alpha}{12} \\ \frac{1}{3}-\frac{\alpha}{12} \\ \frac{1}{3}-\frac{\alpha}{12} \end{pmatrix} (1,1,1,1)\,.$$

Thus it is easy to see that the stationary probability distribution of $\tilde{P}$ is given by

$$\mathbf{p} = (\frac{1}{6}+\frac{\alpha}{12}, \frac{1}{6}+\frac{\alpha}{12}, \frac{1}{3}-\frac{\alpha}{12}, \frac{1}{3}-\frac{\alpha}{12})^T.$$

This probability distribution $\mathbf{p}$ should be consistent with the observed distribution $\mathbf{q}$ of the observed sequence, i.e.

$$\mathbf{p} = (\frac{1}{6}+\frac{\alpha}{12}, \frac{1}{6}+\frac{\alpha}{12}, \frac{1}{3}-\frac{\alpha}{12}, \frac{1}{3}-\frac{\alpha}{12})^T \approx \mathbf{q} = (\frac{1}{6}, \frac{1}{4}, \frac{1}{4}, \frac{1}{3})^T.$$

This suggests a natural method to estimate α. The unknown transition probability α can then be obtained by solving the minimization problem:

$$\min_{0\le\alpha\le 1} ||\mathbf{p}-\mathbf{q}||.$$

If we choose $||.||$ to be the $||.||_2$, then one may consider the following minimization problem:

$$\min_{0\le\alpha\le 1} ||\mathbf{p}-\mathbf{q}||_2^2 = \min_{0\le\alpha\le 1} \sum_{i=1}^{4} (\mathbf{p}_i - \mathbf{q}_i)^2. \tag{4.1}$$

In this case, it is a standard constrained least squares problem, and can be solved easily. For more detailed discussion on statistical inference of a HMM, we refer readers to the book by MacDonald and Zucchini [155].

4.3 An Extension of the Method

In this section, the parameter estimation method is extended to a general HMM with m hidden states and n observable states. In general, the number of hidden states can be more than two. Suppose the number of hidden states is m and the stationary distribution of the hidden states is given by

$$\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m).$$

Suppose the number of observable states is n, and when the hidden state is $i (i = 1, 2, \ldots, m)$, the stationary distribution of the observable states is

$$(p_{i1}, p_{i2}, \ldots, p_{in}).$$

We assume that m, n and p_{ij} are known. Given an observed sequence of the observable states, one can calculate the occurrences of each state in the sequence and hence the observed distribution $\mathbf{q}$. Using the same trick discussed in Sect. 4.2, if we ignore the hidden states, the observable states follow the one-step transition probability matrix:

$$\tilde{P}_2 = \begin{pmatrix} p_{11} & p_{21} & \cdots & p_{m1} \\ p_{12} & p_{22} & \cdots & p_{m2} \\ \vdots & \vdots & \vdots & \vdots \\ p_{1n} & p_{2n} & \cdots & p_{mn} \end{pmatrix} \begin{pmatrix} \alpha_1 & \alpha_1 & \cdots & \alpha_1 \\ \alpha_2 & \alpha_2 & \cdots & \alpha_2 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_m & \alpha_m & \cdots & \alpha_m \end{pmatrix} = \mathbf{p}(1, 1, \ldots, 1) \tag{4.2}$$

where

$$\mathbf{p} = (\sum_{k=1}^{m} \alpha_k p_{k1}, \sum_{k=1}^{m} \alpha_k p_{k2}, \ldots, \sum_{k=1}^{m} \alpha_k p_{kn})^T.$$

It is easy to check that

$$\tilde{P}_2 \mathbf{p} = \mathbf{p} \quad \text{and} \quad \sum_{k=1}^{n} \mathbf{p}_k = 1.$$

Thus the following proposition can easily be proved.

Proposition 4.1. *The vector* $\mathbf{p}$ *is the stationary distribution of* $\tilde{P}_2$.

Therefore the transition probabilities of the hidden states

$$\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m)$$

can be obtained by solving

$$\min_{\alpha} ||\mathbf{p} - \mathbf{q}||_2^2$$

subject to

$$\sum_{k=1}^{m} \alpha_k = 1 \quad \text{and} \quad \alpha_k \geq 0.$$

4.4 A Special Case Analysis

In this section, a detailed discussion is given for the model having two hidden states. In this case one may re-write (4.2) as follows:

$$\bar{P} = \begin{pmatrix} p_{11} & p_{21} \\ p_{12} & p_{22} \\ \vdots & \vdots \\ p_{1n} & p_{2n} \end{pmatrix} \begin{pmatrix} \alpha_1 & \alpha_1 & \cdots & \alpha_1 \\ 1-\alpha_1 & 1-\alpha_1 & \cdots & 1-\alpha_1 \end{pmatrix} = \mathbf{p}(1, 1, \ldots, 1) \qquad (4.3)$$

where

$$\mathbf{p} = (\alpha p_{11} + (1-\alpha)p_{21}, \alpha p_{12} + (1-\alpha)p_{22}, \ldots, \alpha p_{1n} + (1-\alpha)p_{2n})^T.$$

It is easy to check that

$$\bar{P}\mathbf{p} = \mathbf{p} \quad \text{and} \quad \sum_{i=1}^{n} \mathbf{p}_i = 1$$

and therefore **p** is the steady-state probability distribution.

Suppose the observed distribution **q** of the observable states is given, then α can be estimated by the following minimization problem:

$$\min_{\alpha} ||\mathbf{p} - \mathbf{q}||_2^2$$

subject to $0 \le \alpha \le 1$ or equivalently

$$\min_{0 \le \alpha \le 1} \sum_{k=1}^{n} \{\alpha p_{1k} + (1-\alpha)p_{2k} - q_k\}^2 .$$

The following proposition can be obtained by direct verification.

Proposition 4.2. *Let*

$$\tau = \frac{\displaystyle\sum_{j=1}^{n}(q_j - p_{2j})(p_{1j} - p_{2j})}{\displaystyle\sum_{j=1}^{n}(p_{1j} - p_{2j})^2},$$

then the optimal value of α is given as follows:

$$\alpha = \begin{cases} 0 \text{ if } \tau \le 0; \\ \tau \text{ if } 0 < \tau < 1; \\ 1 \text{ if } \tau \ge 1. \end{cases}$$

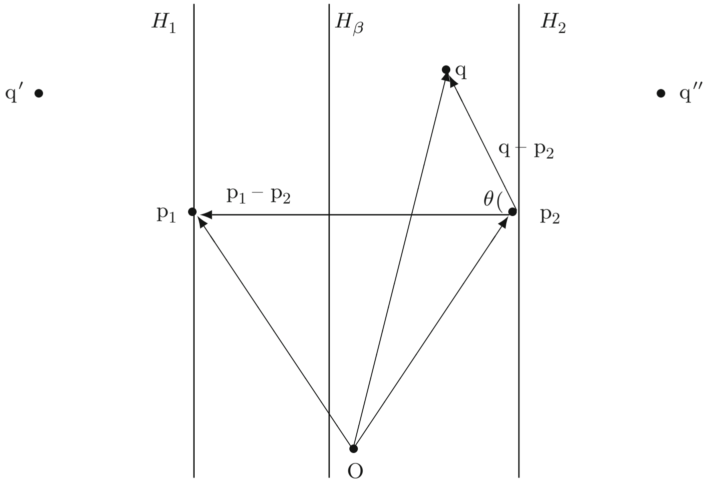


Fig. 4.1 The graphical interpretation of Proposition 4.2

One may interpret the result in Proposition 4.2 as follow:

$$\tau = \frac{< (\mathbf{q} - \mathbf{p}_2), (\mathbf{p}_1 - \mathbf{p}_2) >}{< (\mathbf{p}_1 - \mathbf{p}_2), (\mathbf{p}_1 - \mathbf{p}_2) >} = \frac{||\mathbf{q} - \mathbf{p}_2||_2 \cos(\theta)}{||\mathbf{p}_1 - \mathbf{p}_2||_2}. \tag{4.4}$$

Here $< .,. >$ is the standard inner product on the vector space R^n,

$$\mathbf{p}_1 = (p_{11}, p_{12}, \ldots, p_{1n})^T$$

and

$$\mathbf{p}_2 = (p_{21}, p_{22}, \ldots, p_{2n})^T.$$

Moreover, $||.||_2$ is the L_2-norm on R^n and θ is the angle between the vectors

$$(\mathbf{q} - \mathbf{p}_2) \quad \text{and} \quad (\mathbf{p}_1 - \mathbf{p}_2).$$

Two hyperplanes H_1 and H_2 are defined in R^n. Both hyperplanes are perpendicular to the vector $(\mathbf{p}_1 - \mathbf{p}_2)$ and H_i contains the point $\mathbf{p}_i$ (distribution) for $i = 1, 2$, see Fig. 4.1 (Taken from [63]). From (4.4), Proposition 4.2 and Fig. 4.1, any point $\mathbf{q}'$ on the left of the hyperplane H_1 has the following property:

$$||\mathbf{q} - \mathbf{p}_2||_2 \cos(\theta) \geq ||\mathbf{p}_1 - \mathbf{p}_2||_2.$$

Hence for such $\mathbf{q}'$, the optimal α is 1. For a point $\mathbf{q}''$ on the right of the hyperplane H_2, $\cos(\theta) \leq 0$ and hence the optimal α is zero. Lastly, for a point $\mathbf{q}$ in between the two hyperplanes, the optimal α lies between 0 and 1 and the optimal value is given by τ in (4.4). This special case motivates us to apply the HMM in the classification of customers.

4.5 Applying HMM to the Classification of Customers

In this section, the HMM discussed in Sect. 4.4 is applied to the classification of the customers of a computer service company. We remark that there are a number of classification methods such as machine learning [16, 143, 179] and Bayesian learning, interested readers can consult the book by Young and Calvert [212]. In this problem, HMM is an efficient and effective classification method but we make no claim that HMM is the best one.

A computer service company offers four types of long distance call services I, II, III and IV (four different periods of a day). From the customer database, the information of the expenditure distribution of 71 randomly chosen customers is obtained. A longitudinal study was then carried out for half a year to investigate the customers. Their behavior and responses were captured and monitored during the period of investigation. For simplicity of discussion, the customers are classified into two groups. Among them 22 customers are known to be loyal customers (Group A) and the other 49 customers are not loyal customers (Group B). This classification is useful to marketing managers when they plan any promotions. The customers in Group A will be given promotions on new services and products. While the customers in Group B will be offered discounts on the current services to prevent them from switching to the competitor companies.

Two-thirds of the data are used to build the HMM and the remaining data are used to validate the model. Therefore, 16 candidates are randomly taken (these customers are labeled as the first 16 customers in Table 4.2) from Group A and 37 candidates from group B. The remaining 6 candidates (the first 6 customers in Table 4.4) from Group A and 12 candidates from Group B are used for validating the constructed HMM. A HMM having four observable states (I, II, III and IV) and two hidden states (Group A and Group B) is then built.

From the information of the customers in Group A and Group B in Table 4.2, the average expenditure distributions for both groups are computed in Table 4.3. This means that a customer in Group A (Group B) is characterized by the expenditure distribution in the first (second) row of Table 4.3.

An interesting problem is the following: Given the expenditure distribution of a customer, how does one classify the customer correctly (Group A or Group B) based on the information in Table 4.4? To tackle this problem, one can apply the method discussed in the previous section to compute the transition probability α in the hidden states. This value of α can be used to classify a customer. If α is close to 1 then the customer is likely to be a loyal customer. If α is close to 0 then the customer is likely to be an unloyal customer.

The values of α for all the 53 customers are listed in Table 4.2. It is interesting to note that the values of α of all the first 16 customers (Group A) lie in the interval $[0.83, 1.00]$. While the values of α of all the other customers (Group B) lie in the interval $[0.00, 0.69]$. Based on the values of α obtained, the two groups of customers can be clearly separated by setting the cutoff value β to be 0.75. A possible decision

Table 4.2 Two-third of the data are used to build the HMM

Customer	I	II	III	IV	α	Customer	I	II	III	IV	α
1	1.00	0.00	0.00	0.00	1.00	2	1.00	0.00	0.00	0.00	1.00
3	0.99	0.01	0.00	0.00	1.00	4	0.97	0.03	0.00	0.00	1.00
5	0.87	0.06	0.04	0.03	0.98	6	0.85	0.15	0.00	0.00	0.92
7	0.79	0.18	0.02	0.01	0.86	8	0.77	0.00	0.23	0.00	0.91
9	0.96	0.01	0.00	0.03	1.00	10	0.95	0.00	0.02	0.03	1.00
11	0.92	0.08	0.00	0.00	1.00	12	0.91	0.09	0.00	0.00	1.00
13	0.83	0.00	0.17	0.00	0.97	14	0.82	0.18	0.00	0.00	0.88
15	0.76	0.04	0.00	0.20	0.87	16	0.70	0.00	0.00	0.30	0.83
17	0.62	0.15	0.15	0.08	0.69	18	0.57	0.14	0.00	0.29	0.62
19	0.56	0.00	0.39	0.05	0.68	20	0.55	0.36	0.01	0.08	0.52
21	0.47	0.52	0.00	0.01	0.63	22	0.46	0.54	0.00	0.00	0.36
23	0.25	0.75	0.00	0.00	0.04	24	0.22	0.78	0.00	0.00	0.00
25	0.21	0.01	0.78	0.00	0.32	26	0.21	0.63	0.00	0.16	0.03
27	0.18	0.11	0.11	0.60	0.22	28	0.18	0.72	0.00	0.10	0.00
29	0.15	0.15	0.44	0.26	0.18	30	0.07	0.93	0.00	0.00	0.00
31	0.04	0.55	0.20	0.21	0.00	32	0.03	0.97	0.00	0.00	0.00
33	0.00	0.00	1.00	0.00	0.10	34	0.00	1.00	0.00	0.00	0.00
35	0.00	0.00	0.92	0.08	0.10	36	0.00	0.94	0.00	0.06	0.00
37	0.03	0.01	0.96	0.00	0.13	38	0.02	0.29	0.00	0.69	0.00
39	0.01	0.97	0.00	0.02	0.00	40	0.01	0.29	0.02	0.68	0.00
41	0.00	0.24	0.00	0.76	0.00	42	0.00	0.93	0.00	0.07	0.00
43	0.00	1.00	0.00	0.00	0.00	44	0.00	1.00	0.00	0.00	0.00
45	0.00	0.98	0.02	0.00	0.00	46	0.00	0.00	0.00	1.00	0.06
47	0.00	1.00	0.00	0.00	0.00	48	0.00	0.96	0.00	0.04	0.00
49	0.00	0.91	0.00	0.09	0.00	50	0.00	0.76	0.03	0.21	0.00
51	0.00	0.00	0.32	0.68	0.07	52	0.00	0.13	0.02	0.85	0.01
53	0.00	0.82	0.15	0.03	0.00						

Table 4.3 The average expenditures of Group A and Group B

Group	I	II	III	IV
A	0.8806	0.0514	0.0303	0.0377
B	0.1311	0.5277	0.1497	0.1915

rule can therefore be defined as follows: Classify a customer to Group A if $\alpha \geq \beta$, otherwise classify the customer to Group B. Referring to Fig. 4.1, it is clear that the customers are separated by the hyperplane H_β. The hyperplane H_β is parallel to the two hyperplanes H_1 and H_2 such that it has a perpendicular distance of β from H_2.

The decision rule is applied to the remaining 22 captured customers. Among them, 6 customers (the first six customers in Table 4.4) belong to Group A and 12 customers belong to Group B. Their α values are computed and listed in Table 4.4. It is clear that if the value of β is set to be 0.75, all the customers will be classified correctly.

Table 4.4 The remaining one-third of the data for validation of the HMM

Customer	I	II	III	IV	α	Customer	I	II	III	IV	α
1'	0.98	0.00	0.02	0.00	1.00	2'	0.88	0.01	0.01	0.10	1.00
3'	0.74	0.26	0.00	0.00	0.76	4'	0.99	0.01	0.00	0.00	1.00
5'	0.99	0.01	0.00	0.00	1.00	6'	0.89	0.10	0.01	0.00	1.00
7'	0.00	0.00	1.00	0.00	0.10	8'	0.04	0.11	0.68	0.17	0.08
9'	0.00	0.02	0.98	0.00	0.09	10'	0.18	0.01	0.81	0.00	0.28
11'	0.32	0.05	0.61	0.02	0.41	12'	0.00	0.00	0.97	0.03	0.10
13'	0.12	0.14	0.72	0.02	0.16	14'	0.00	0.13	0.66	0.21	0.03
15'	0.00	0.00	0.98	0.02	0.10	16'	0.39	0.00	0.58	0.03	0.50
17'	0.27	0.00	0.73	0.00	0.38	18'	0.00	0.80	0.07	0.13	0.00

Table 4.5 Probability distributions of dice A and dice B

Dice	1	2	3	4	5	6
A	1/6	1/6	1/6	1/6	1/6	1/6
B	1/12	1/12	1/4	1/4	1/4	1/12

Table 4.6 Observed distributions of dots

Dice	1	2	3	4	5	6
A	1/8	1/8	1/4	1/4	1/8	1/8

4.6 Summary

In this chapter, we propose a simple HMM with estimation methods. The framework of the HMM is simple and the model parameters can be estimated efficiently. Application to customer classification with practical data taken from a computer service company is presented and analyzed. Further discussions on new HMMs and applications will be given in Chap. 8.

4.7 Exercises

1. Solve the minimization problem (4.1) to get the optimal α.
2. Consider the process of choosing a die again. Suppose that this time we have two dice A and B, and that each of them has six faces ($1, 2, 3, 4, 5$ and 6). The probability distribution of dots obtained by throwing dice A and B are given in Table 4.5. Each time a die is to be chosen, we assume that with probability α, Die A is chosen, and with probability $(1 - \alpha)$, Die B is chosen. This process is hidden as we don't know which die is chosen. After a long-run observation, the observed distribution of dots is given in Table 4.6. Use the method in Sect. 4.2 to estimate the value of α.
3. Prove Proposition 4.1.
4. Prove Proposition 4.2.

Table 4.7 The new average expenditures of Group A and Group B

Group	I	II	III	IV
A	0.6000	0.1000	0.0500	0.2500
B	0.1500	0.5000	0.1500	0.2000

5. Consider the classification problem in Sect. 4.5. Suppose the average expenditures of Groups A and Group B have been changed and are recorded in Table 4.7, then find the new α for the classification rule.

Chapter 5
Markov Decision Processes for Customer Lifetime Value

5.1 Introduction

In this chapter a stochastic dynamic programming model with a Markov chain is proposed to capture customer behavior. The advantage of using Markov chains is that the model can take into account the customers switching between the company and its competitors. Therefore customer relationships can be described in a probabilistic way, see for instance Pfeifer and Carraway [170]. Stochastic dynamic programming is then applied to solve the optimal allocation of the promotion budget for maximizing the Customer Lifetime Value (CLV). The proposed model is then applied to practical data in a computer services company.

The customer equity should be measured in making the promotion plan so as to achieve an acceptable and reasonable budget. A popular approach is the CLV. Kotler and Armstrong [134] define a profitable customer as "a person, household, or company whose revenues over time exceeds, by an acceptable amount, the company costs of attracting, selling, and servicing that customer." This excess is called the CLV. In some literature, CLV is also referred to as "customer equity" [14]. In fact, some researchers define CLV as the customer equity less the acquisition cost. Nevertheless, in this chapter CLV is defined as the present value of the projected net cash flow that a firm expects to receive from the customer over time [35]. Recognizing its importance in decision making, CLV has been successfully applied to the problems of pricing strategy [13], media selection [121] and setting optimal promotion budget [20, 61].

To calculate the CLV, a company should estimate the expected net cash flow that they expect to receive from the customer over time. The CLV is the present value of that stream of cash flow. However, it is a difficult task to estimate the net cash flow to be received from the customer. In fact, one needs to answer, the following questions:

1. How many customers one can attract given a specific advertising budget?
2. What is the probability that the customer will stay with the company?
3. How does this probability change with respect to the promotion budget?

W.-K. Ching et al., *Markov Chains*, International Series in Operations Research & Management Science 189, DOI 10.1007/978-1-4614-6312-2_5,

To answer the first question, there are a number of advertising models found in the book by Lilien, Kotler and Moorthy [148]. The second and the third questions give rise to an important concept, the retention rate. The retention rate [123] is defined as "the chance that the account will remain with the vendor for the next purchase, provided that the customer has bought from the vendor on each previous purchase". Jackson [123] proposed an estimation method for the retention rate based on historical data. Other models for the retention rate can also be found in [82, 148].

Blattberg and Deighton [20] proposed a formula for the calculation of CLV (customer equity). The model is simple and deterministic. Using their notations (see also [13, 14]), the CLV is the sum of two net present values: the return from acquisition spending and the return from retention spending. In their model, CLV is defined as

$$\begin{aligned} \text{CLV} &= \underbrace{am - A}_{acquisition} + \underbrace{\sum_{k=1}^{\infty} a(m - \frac{R}{r})[r(1+d)^{-1}]^k}_{retention} \\ &= am - A + a(m - \frac{R}{r}) \times \frac{r}{(1+d-r)} \end{aligned} \tag{5.1}$$

where a is the acquisition rate, A is the level of acquisition spending, m is the margin on a transaction, R is the retention spending per customer per year, r is the yearly retention rate (a proportion), and d is the yearly discount rate appropriate for marketing investment. Moreover, they also assume that the acquisition rate a and retention rate r are functions of A and R respectively, and are given by

$$a(A) = a_0(1 - e^{-K_1 A})$$

and

$$r(R) = r_0(1 - e^{-K_2 R})$$

where a_0 and r_0 are the estimated ceiling rates and K_1 and K_2 are two positive constants. In this chapter, a stochastic model (Markov decision process) is proposed for the calculation of CLV and promotion planning.

The rest of the chapter is organized as follows. In Sect. 5.2, the Markov chain model for modelling the behavior of customers is presented. Section 5.4 extends the model to consider multi-period promotions. In Sect. 5.3, stochastic dynamic programming is then used to calculate the CLV of the customers for three different scenarios:

1. infinite horizon without constraint (without limit in the number of promotions),
2. finite horizon (with a limited number of promotions), and
3. infinite horizon with constraints (with a limited number of promotions).

In Sect. 5.5, we consider higher-order Markov decision processes with applications to the CLV problem. Finally a summary is given to conclude the chapter in Sect. 5.5.

5.2 Markov Chain Models for Customer Behavior

In this section, a Markov chain model for modelling the customers' behavior in a market is introduced. According to the usage of the customer, a company customer can be classified into N possible states

$$\{0, 1, 2, \ldots, N-1\}.$$

For example, customers can be classified into four states ($N = 4$): low-volume user (state 1), medium-volume user (state 2) and high-volume user (state 3) and in order to classify all customers in the market, state 0 is introduced. A customer is said to be in state 0 if they are either a customer of the competitor company or they did not purchase the service during the period of observation. Therefore at any time, a customer in the market belongs to exactly one of the states in $\{0, 1, 2, \ldots, N-1\}$. With this notation, a Markov chain is a good choice to model the transitions of customers among the states in the market.

A Markov chain model is characterized by an $N \times N$ transition probability matrix P. Here $P_{ij} (i, j = 0, 1, 2, \ldots, N-1)$ is the transition probability that a customer will move to state i in the next period given that currently they are in state j. Hence the retention probability of a customer in state $i (i = 0, 1, \ldots, N-1)$ is given by P_{ii}. If the underlying Markov chain is assumed to be irreducible then the stationary distribution $\mathbf{p}$ exists, see for instance [181]. This means that there is a unique

$$\mathbf{p} = (p_0, p_1, \ldots, p_{N-1})^T$$

such that

$$\mathbf{p} = P\mathbf{p}, \quad \sum_{i=0}^{N-1} p_i = 1, \quad p_i \geq 0. \tag{5.2}$$

By making use of the stationary distribution $\mathbf{p}$, one can compute the retention probability of a customer as follows:

$$\begin{aligned} \sum_{i=1}^{N-1} \left(\frac{p_i}{\sum_{j=1}^{N-1} p_j} \right) (1 - P_{i0}) &= 1 - \frac{1}{1 - p_0} \sum_{i=1}^{N-1} p_i P_{0i} \\ &= 1 - \frac{p_0(1 - P_{00})}{1 - p_0}. \end{aligned} \tag{5.3}$$

This is the probability that a customer will purchase service with the company in the next period. Apart from the retention probability, the Markov model can also help

Table 5.1 The four classes of customers

State	0	1	2	3
Hours	0.00	$1-20$	$21-40$	>40

us in computing the CLV. In this case c_i is defined to be the revenue obtained from a customer in state i. Then the expected revenue is given by

$$\sum_{i=0}^{N-1} c_i p_i . \tag{5.4}$$

The above retention probability and the expected revenue are computed under the assumption that the company makes no promotion (in a non-competitive environment) throughout the period. The transition probability matrix P can be significantly different when there is a promotion made by the company. To demonstrate this, an application is given in the following subsection. Moreover, when promotions are allowed, what is the best promotion strategy such that the expected revenue is maximized? Similarly, what is the best strategy when there is a fixed budget for the promotions, e.g. the number of promotions is fixed? These issues will be discussed in the following section by using the stochastic dynamic programming model.

5.2.1 Estimation of the Transition Probabilities

In order to apply the Markov chain model, one has to estimate the transition probabilities from the practical data. In this subsection, an example in the computer service company is used to demonstrate the estimation. In the captured database of customers, each customer has four important attributes (A, B, C, D): A is the "Customer Number", each customer has a unique identity number. B is the "Week", the time (week) when the data was captured. C is the "Revenue" which is the total amount of money the customer spent in the captured week. D is the "Hour", the number of hours that the customer consumed in the captured week.

The total number of weeks of data available is 20. Among these 20 weeks, the company has a promotion for 8 consecutive weeks and no promotion for the other 12 consecutive weeks. The behavior of customers in the period of promotion and no-promotion will be investigated. For each week, all the customers are classified into four states $\{0, 1, 2, 3\}$ according to the amount of "hours" consumed, see Table 5.1. We recall that a customer is said to be in state 0, if they are a customer of a competitor company or they did not use the service for the whole week.

From the data, one can estimate two transition probability matrices, one for the promotion period (8 consecutive weeks) and the other one for the no-promotion period (12 consecutive weeks). For each period, the number of customers switching from state i to state j is recorded. Then, divide this number the total number of

Table 5.2 The average revenue of the four classes of customers

State	0	1	2	3
Promotion	0.00	6.97	18.09	43.75
No-promotion	0.00	14.03	51.72	139.20

customers in the state i, and one obtains the estimates for the one-step transition probabilities. Hence the transition probability matrices under the promotion period $P^{(1)}$ and the no-promotion period $P^{(2)}$ are given respectively below:

$$P^{(1)} = \begin{pmatrix} 0.8054 & 0.4163 & 0.2285 & 0.1372 \\ 0.1489 & 0.4230 & 0.3458 & 0.2147 \\ 0.0266 & 0.0992 & 0.2109 & 0.2034 \\ 0.0191 & 0.0615 & 0.2148 & 0.4447 \end{pmatrix}$$

and

$$P^{(2)} = \begin{pmatrix} 0.8762 & 0.4964 & 0.3261 & 0.2380 \\ 0.1064 & 0.4146 & 0.3837 & 0.2742 \\ 0.0121 & 0.0623 & 0.1744 & 0.2079 \\ 0.0053 & 0.0267 & 0.1158 & 0.2809 \end{pmatrix}.$$

$P^{(1)}$ is very different from $P^{(2)}$. In fact, there can be more than one type of promotion in general, as the transition probability matrices for modelling the behavior of the customers can be more than two.

5.2.2 Retention Probability and CLV

The stationary distributions of the two Markov chains having transition probability matrices $P^{(1)}$ and $P^{(2)}$ are given respectively by

$$\mathbf{p}^{(1)} = (0.2306, 0.0691, 0.0738, 0.6265)^T$$

and

$$\mathbf{p}^{(2)} = (0.1692, 0.0285, 0.0167, 0.7856)^T.$$

The retention probabilities (cf. (5.3)) in the promotion period and no-promotion period are given respectively by 0.6736 and 0.5461. It is clear that the retention probability is significantly higher when the promotion is carried out.

From the customer data in the database, the average revenue of a customer is obtained in different states in both the promotion period and no-promotion period, see Table 5.2. We remark that in the promotion period, a big discount was given to the customers and therefore the revenue was significantly less than the revenue in the no-promotion period.

From (5.4), the expected revenue from a customer in the promotion period (assume that the only promotion cost is the discount rate) and no-promotion period are given by 2.42 and 17.09 respectively.

Although one can obtain the CLVs of the customers in the promotion period and the no-promotion period, one would expect to calculate the CLV in a mixture of promotion and no-promotion periods. This is especially true when the promotion budget is limited (the number of promotions is fixed) and one would like to obtain the optimal promotion strategy. Stochastic dynamic programming with Markov process provides a good approach for solving the above problems. Moreover, the optimal stationary strategy for the customers in different states can also be obtained by solving the stochastic dynamic programming problem.

5.3 Stochastic Dynamic Programming Models

The problem of solving the optimal promotion strategy can be fitted into the framework of stochastic dynamic programming models. In this section, stochastic dynamic programming models are presented for maximizing the CLV under an optimal promotion strategy. The notations of the model are given as follows:

1. N, the total number of states (indexed by $i = 0, 1, \ldots, N-1$)
2. A_i, the set containing all the actions in state i (indexed by k)
3. T, number of months remaining in the planning horizon (indexed by $t = 1, \ldots, T$)
4. d_k, the resources required for carrying out the action k in each period;
5. $c_i^{(k)}$, the revenue obtained from a customer in state i with the action k in each period
6. $p_{ij}^{(k)}$, the transition probability for a customer moving from state j to state i under the action k in each period
7. α, discount rate

Similar to the MDP introduced in Chap. 1, the value of an optimal policy $v_i(t)$ is defined to be the total expected revenue obtained in the stochastic dynamic programming model with t months remaining for a customer in state i for $i = 0, 1, \ldots, N-1$ and $t = 1, 2, \ldots, T$. Therefore, the recursive relation for maximizing the revenue is given as follows:

$$v_i(t) = \max_{k \in A_i} \left\{ c_i^{(k)} - d_k + \alpha \sum_{j=0}^{N-1} p_{ji}^{(k)} v_j(t-1) \right\}. \tag{5.5}$$

In the following subsections, three different CLV models based on the above recursive relation are considered. They are infinite horizon without constraints, finite horizon with hard constraints and infinite horizon with constraints. For each case, an application with practical data in a computer service company is presented.

5.3.1 Infinite Horizon Without Constraints

in this section, we consider an infinite horizon stochastic dynamic programming formulation for the promotion problem. From the standard results in stochastic dynamic programming [208], for each i, the optimal values v_i for the discounted infinite horizon Markov decision process satisfy the relationship

$$v_i = \max_{k \in A_i} \left\{ c_i^{(k)} - d_k + \alpha \sum_{j=0}^{N-1} p_{ji}^{(k)} v_j \right\}. \tag{5.6}$$

Therefore we have

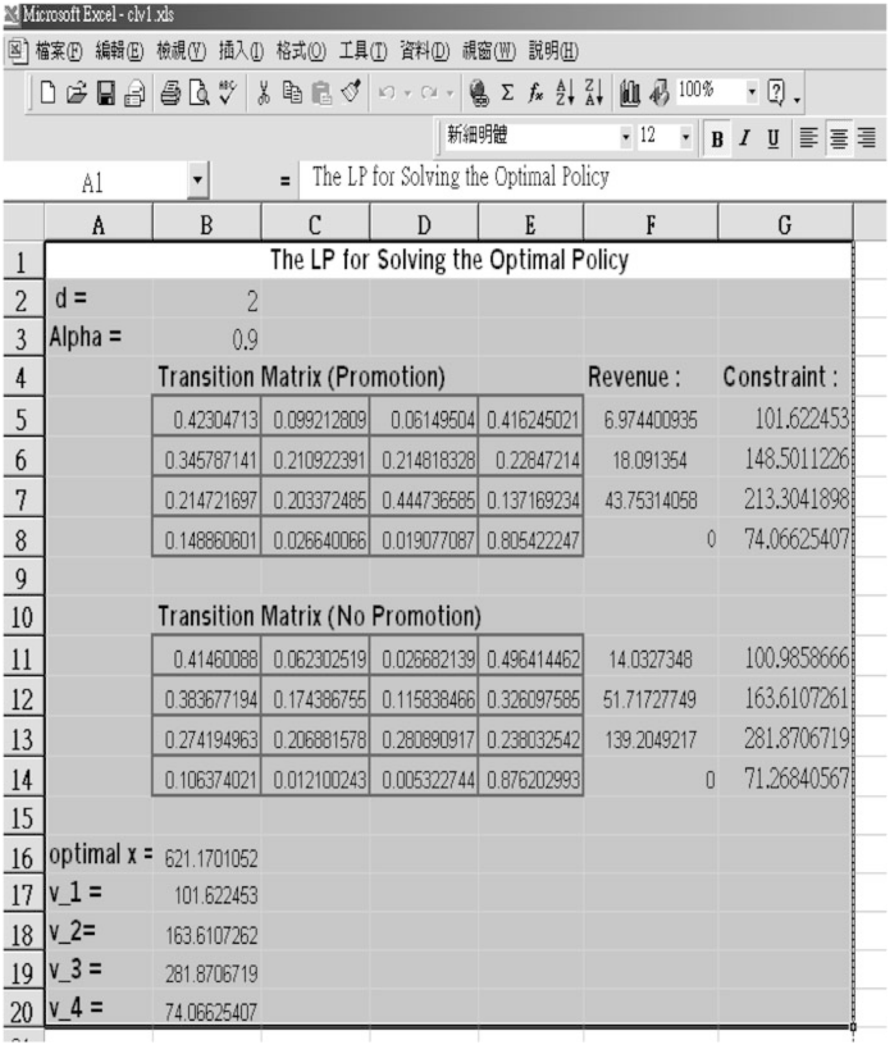
$$v_i \geq c_i^{(k)} - d_k + \alpha \sum_{j=0}^{N-1} p_{ji}^{(k)} v_j \tag{5.7}$$

for each i. In fact, the optimal values v_i are the smallest numbers (the least upper bound over all possible policy values) that satisfy these inequalities. This suggests that the problem of determining the v_i value can be transformed into the following linear programming problem [3, 207, 208]:

$$\begin{cases} \quad \min \quad x_0 = \sum_{i=0}^{N-1} v_i \\ \text{subject to} \\ \quad v_i \geq c_i^{(k)} - d_k + \alpha \sum_{j=0}^{N-1} p_{ji}^{(k)} v_j, \quad \text{for} \quad i = 0, \ldots, N-1; \\ \quad v_i \geq 0 \quad \text{for} \quad i = 0, \ldots, N-1. \end{cases} \tag{5.8}$$

The above linear programming problem can be solved easily by using speadsheet software such as EXCEL. In addition, a demonstration EXCEL file is available at the following site (http://hkumath.hku.hk/~wkc/clv1.zip), see also Fig. 5.1 (Taken from [64]). Returning to the model for the computer service company, there are 2 actions available (either (P) promotion or (NP) no-promotion) for all possible states. Thus $A_i = \{P, NP\}$ for all $i = 0, \ldots, N-1$. Moreover, customers are classified into 4 clusters, thus $N = 4$ (possible states of a customer are $0, 1, 2, 3$). Since no promotion cost is incurred for the action (NP), therefore $d_{NP} = 0$. For simplification, d is used to denote the only promotion cost instead of d_P in the application.

Table 5.4 presents the optimal stationary policies (i.e., to have promotion of $D_i = P$ or no-promotion $D_i = NP$ depends on the state i of the customer) and the corresponding revenues for different discount factors α and fixed promotion costs d. For instance, when the promotion cost is 0 and the discount factor is 0.99, then the optimal strategy is that when the current state is 0 or 1, the promotion should be done i.e. $D_0 = D_1 = P$, and when the current state is 2 or 3, no promotion


Microsoft Excel - clv1.xls
檔案(F) 編輯(E) 檢視(V) 插入(I) 格式(O) 工具(T) 資料(D) 視窗(W) 說明(H)
100%
新細明體
12
B I U
A1
=
The LP for Solving the Optimal Policy

	A	B	C	D	E	F	G
1	The LP for Solving the Optimal Policy						
2	d =	2					
3	Alpha =	0.9					
4		Transition Matrix (Promotion)				Revenue :	Constraint :
5		0.42304713	0.099212809	0.06149504	0.416245021	6.974400935	101.622453
6		0.345787141	0.210922391	0.214818328	0.22847214	18.091354	148.5011226
7		0.214721697	0.203372485	0.444736585	0.137169234	43.75314058	213.3041898
8		0.148860601	0.026640066	0.019077087	0.805422247	0	74.06625407
9							
10		Transition Matrix (No Promotion)					
11		0.41460088	0.062302519	0.026682139	0.496414462	14.0327348	100.9858666
12		0.383677194	0.174386755	0.115838466	0.326097585	51.71727749	163.6107261
13		0.274194963	0.206881578	0.280890917	0.238032542	139.2049217	281.8706719
14		0.106374021	0.012100243	0.005322744	0.876202993	0	71.26840567
15							
16	optimal x =	621.1701052					
17	v_1 =	101.622453					
18	v_2=	163.6107262					
19	v_3 =	281.8706719					
20	v_4 =	74.06625407					

Fig. 5.1 For solving infinite horizon problem without constraint

is required, i.e. $D_2 = D_3 = NP$ (see the first column of the upper left hand box of Table 5.3). The other values can be interpreted similarly. From the numerical examples, the following conclusions are drawn:

- When the fixed promotion cost d is large, the optimal strategy is that the company should not conduct any promotion on the active customers and should only conduct the promotion scheme to inactive (purchase no service) customers and customers of the competitor company. However, when d is small, the company should take care of the low-volume customers to prevent this group of customers from switching to the competitor companies.

Table 5.3 Optimal stationary policies and their CLVs

	$d=0$			$d=1$			$d=2$		
α	0.99	0.95	0.90	0.99	0.95	0.90	0.99	0.95	0.90
x_0	4791	1149	687	4437	1080	654	4083	1012	621
v_0	1112	204	92	1023	186	83	934	168	74
v_1	1144	234	119	1054	216	110	965	198	101
v_2	1206	295	179	1118	278	171	1030	261	163
v_3	1328	415	296	1240	399	289	1153	382	281
D_0	P	P	P	P	P	P	P	P	P
D_1	P	P	P	P	P	P	P	P	P
D_2	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_3	NP	NP	NP	NP	NP	NP	NP	NP	NP
	$d=3$			$d=4$			$d=5$		
α	0.99	0.95	0.90	0.99	0.95	0.90	0.99	0.95	0.90
x_0	3729	943	590	3375	879	566	3056	827	541
v_0	845	151	65	755	134	58	675	119	51
v_1	877	181	94	788	164	88	707	151	82
v_2	942	245	156	854	230	151	775	217	145
v_3	1066	366	275	978	351	269	899	339	264
D_0	P	P	P	P	P	P	P	P	P
D_1	P	P	NP	P	NP	NP	NP	NP	NP
D_2	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_3	NP	NP	NP	NP	NP	NP	NP	NP	NP

- It is also clear that the CLV of a high-volume user is larger than the CLV of other groups.
- The CLVs of each group depend on the discount rate α significantly. Here the discount rate can be viewed as the technology depreciation of the computer services in the company. Therefore, in order to generate the revenue of the company, new technology and services should be provided.

5.3.2 Finite Horizon with Hard Constraints

In the computer service and telecommunication industry, the product life cycle is short, e.g., it is usually one year. Therefore, the case of finite horizon with limited budget constraint is considered. This problem can also be solved efficiently by using stochastic dynamic programming and the optimal revenues obtained in the previous section are used as the boundary conditions. The model's parameters are defined as follows:

n = number of weeks remaining
p = number of possible promotions remaining

The recursive relation for the problem is given as follows:

$$v_i(n,p) = \max\ \{c_i^{(P)} - d_P + \alpha \textstyle\sum_{j=0}^{N-1} p_{ji}^{(P)} v_j(n-1,p-1), \\ c_i^{(NP)} - d_{NP} + \alpha \textstyle\sum_{j=0}^{N-1} p_{ji}^{(NP)} v_j(n-1,p)\} \tag{5.9}$$

for $n = 1,\ldots,n_{max}$ and $p = 1,\ldots,p_{max}$ and

$$v_i(n,0) = c_i^{(NP)} - d_{NP} + \alpha \sum_{j=0}^{N-1} p_{ji}^{(NP)} v_j(n-1,0) \tag{5.10}$$

for $n = 1,\ldots,n_{max}$. The above dynamic programming problem can be solved easily by using spreadsheet software such as EXCEL. A demonstration EXCEL file can be found at the following site (http://hkumath.hku.hk/~wkc/clv2.zip), see also Fig. 5.2 (Taken from [64]). In the numerical experiment of the computer service company, the length of the planning period is set to be $n_{max} = 52$ and the maximum number of promotions is $p_{max} = 4$. By solving the dynamic programming problem, the optimal values and promotion strategies are listed in Table 5.4. The optimal solution in the table is presented as follows:

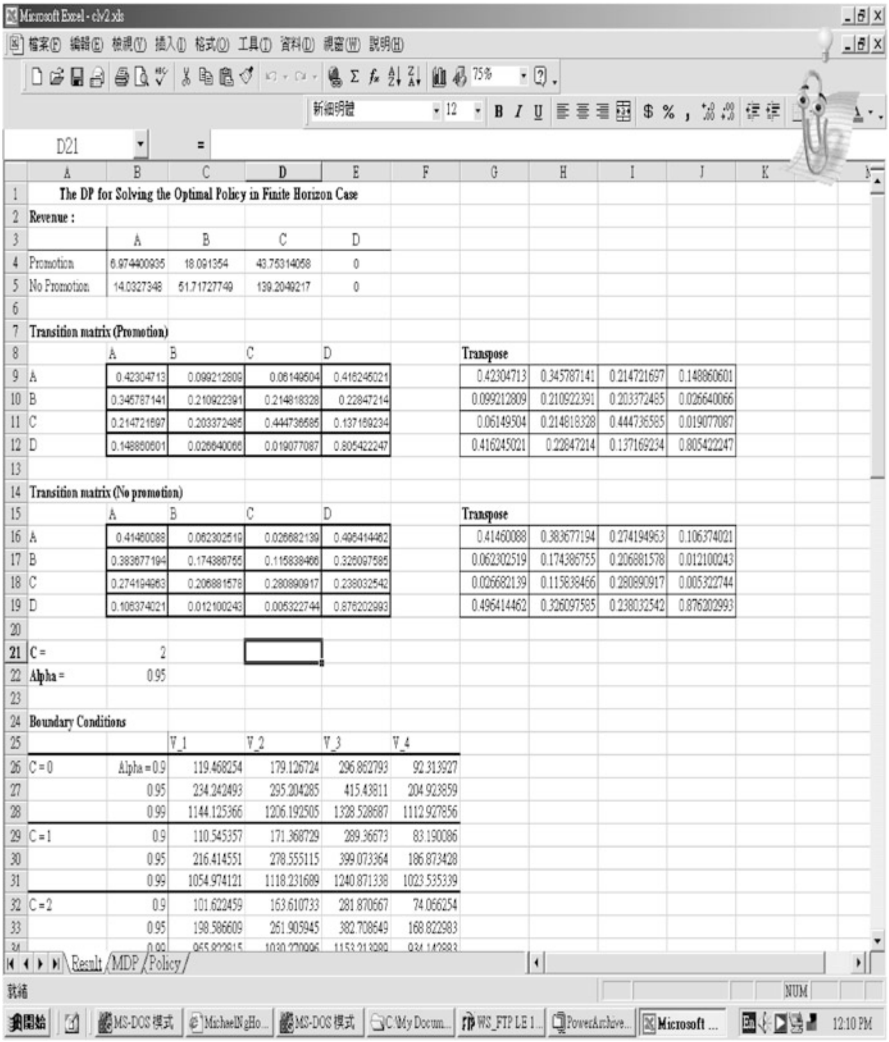
$$(t_1, t_2, t_3, t_4, r^*),$$

where r^* is the optimal expected revenue, and t_i is the promotion week of the optimal promotion strategy and "-" means no promotion. Findings are summarized as follows:

- For different values of the fixed promotion cost d, the optimal strategy for the customers in states 2 and 3 is to conduct no promotion.
- For those in state 0, the optimal strategy is to conduct all four promotions as early as possible.
- In state 1, the optimal strategy depends on the value of d. If d is large, then no promotion will be conducted. However, if d is small, promotions are carried out and the strategy is to conduct the promotions as late as possible.

5.3.3 Infinite Horizon with Constraints

For the purpose of comparison, the model in Sect. 5.3.2 is extended to the infinite horizon case. Similar to the previous model, the finite number of promotions available is denoted by $p_{\max}$. The value function $v_i(p)$, which represents the optimal discounted utility starting at state i when there are p promotions remaining, is the unique fixed point of the equations:



	A	B	C	D	E	F	G	H	I	J
1	**The DP for Solving the Optimal Policy in Finite Horizon Case**									
2	**Revenue :**									
3		A	B	C	D					
4	Promotion	6.974400935	18.091354	43.75314058	0					
5	No Promotion	14.0327348	51.71727749	139.2049217	0					
6										
7	**Transition matrix (Promotion)**									
8		A	B	C	D		**Transpose**			
9	A	0.42304713	0.099212809	0.06149504	0.416245021		0.42304713	0.345787141	0.214721697	0.148860601
10	B	0.345787141	0.210922391	0.214818328	0.22847214		0.099212809	0.210922391	0.203372485	0.026640066
11	C	0.214721697	0.203372485	0.444736585	0.137169234		0.06149504	0.214818328	0.444736585	0.019077087
12	D	0.148860601	0.026640066	0.019077087	0.805422247		0.416245021	0.22847214	0.137169234	0.805422247
13										
14	**Transition matrix (No promotion)**									
15		A	B	C	D		**Transpose**			
16	A	0.41460088	0.062302519	0.026682139	0.496414462		0.41460088	0.383677194	0.274194963	0.106374021
17	B	0.383677194	0.174386755	0.115838466	0.326097585		0.062302519	0.174386755	0.206881578	0.012100243
18	C	0.274194963	0.206881578	0.280890917	0.238032542		0.026682139	0.115838466	0.280890917	0.005322744
19	D	0.106374021	0.012100243	0.005322744	0.876202993		0.496414462	0.326097585	0.238032542	0.876202993
20										
21	C =	2								
22	**Alpha =**	0.95								
23										
24	**Boundary Conditions**									
25			V_1	V_2	V_3	V_4				
26	C = 0	Alpha = 0.9	119.468254	179.126724	296.862793	92.313927				
27		0.95	234.242493	295.204285	415.43811	204.923859				
28		0.99	1144.125366	1206.192505	1328.528687	1112.927856				
29	C = 1	0.9	110.545357	171.368729	289.36673	83.190086				
30		0.95	216.414551	278.555115	399.073364	186.873428				
31		0.99	1054.974121	1118.231689	1240.871338	1023.535339				
32	C = 2	0.9	101.622459	163.610733	281.870667	74.066254				
33		0.95	198.586609	251.905945	382.708649	168.822983				
34		0.99	965.822815	1030.270996	1153.213989	934.142883				

Fig. 5.2 EXCEL for solving finite horizon problem without constraint

$$v_i(p)$$

$$= \max\left\{c_i^{(P)} - d_P + \alpha\sum_{j=0}^{N-1} p_{ji}^{(P)} v_j(p-1), c_i^{(NP)} - d_{NP} + \alpha\sum_{j=0}^{N-1} p_{ji}^{(NP)} v_j(p)\right\}, \tag{5.11}$$

Table 5.4 Optimal promotion strategies and their CLVs

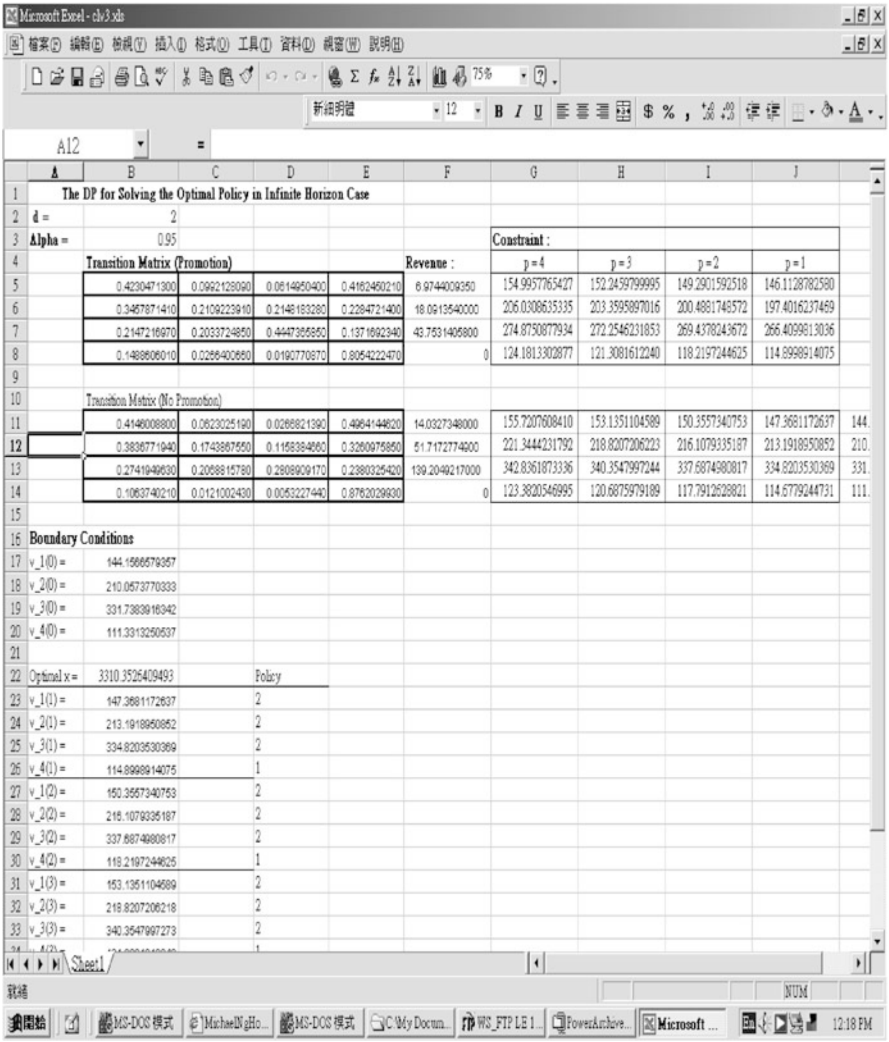
	α	State 0	State 1	State 2	State 3
$d = 0$	0.9	$(1, 2, 3, 4, 67)$	$(1, 45, 50, 52, 95)$	(-,-,-,-,158)	(-,-,-,-,276)
	0.95	$(1, 2, 3, 4, 138)$	$(45, 48, 50, 51, 169)$	(-,-,-,-,234)	(-,-,-,-,335)
	0.99	$(1, 2, 3, 4, 929)$	$(47, 49, 50, 51, 963)$	(-,-,-,-,1031)	(-,-,-,-,1155)
$d = 1$	0.9	$(1, 2, 3, 4, 64)$	$(47, 49, 51, 52, 92)$	(-,-,-,-,155)	(-,-,-,-,274)
	0.95	$(1, 2, 3, 4, 133)$	$(47, 49, 51, 52, 164)$	(-,-,-,-,230)	(-,-,-,-,351)
	0.99	$(1, 2, 3, 4, 872)$	$(47, 49, 51, 52, 906)$	(-,-,-,-,974)	(-,-,-,-,1098)
$d = 2$	0.9	$(1, 2, 3, 4, 60)$	$(49, 50, 51, 52, 89)$	(-,-,-,-,152)	(-,-,-,-,271)
	0.95	$(1, 2, 3, 4, 128)$	$(48, 50, 51, 52, 160)$	(-,-,-,-,225)	(-,-,-,-,347)
	0.99	$(1, 2, 3, 4, 815)$	$(48, 49, 51, 52, 849)$	(-,-,-,-,917)	(-,-,-,-,1041)
$d = 3$	0.9	$(1, 2, 3, 4, 60)$	$(-, -, -, -, 87)$	(-,-,-,-,150)	(-,-,-,-,269)
	0.95	$(1, 2, 3, 4, 123)$	$(49, 50, 51, 52, 155)$	(-,-,-,-,221)	(-,-,-,-,342)
	0.99	$(1, 2, 3, 4, 758)$	$(48, 50, 51, 52, 792)$	(-,-,-,-,860)	(-,-,-,-,984)
$d = 4$	0.9	$(1, 2, 3, 4, 54)$	$(-, -, -, -, 84)$	(-,-,-,-,147)	(-,-,-,-,266)
	0.95	$(1, 2, 3, 4, 119)$	$(-, -, -, -, 151)$	(-,-,-,-,217)	(-,-,-,-,338)
	0.99	$(1, 2, 3, 4, 701)$	$(49, 50, 51, 52, 736)$	(-,-,-,-,804)	(-,-,-,-,928)
$d = 5$	0.9	$(1, 2, 3, 4, 50)$	(-,-,-,-,81)	(-,-,-,-,144)	(-,-,-,-,264)
	0.95	$(1, 2, 3, 4, 114)$	(-,-,-,-,147)	(-,-,-,-,212)	(-,-,-,-,334)
	0.99	$(1, 2, 3, 4, 650)$	(-,-,-,-,684)	(-,-,-,-,752)	(-,-,-,-,876)

for $p = 1, \ldots, p_{\max}$, and

$$v_i(0) = c_i^{(NP)} - d_{NP} + \alpha \sum_{j=0}^{N-1} p_{ji}^{(NP)} v_j(0). \tag{5.12}$$

Since $[p_{ij}^{(k)}]$ is a transition probability matrix, the set of linear equations (5.12) with four unknowns has a unique solution. We note that (5.11) can be computed by the value iteration algorithm, i.e., as the limit of $v_i(n, p)$ (computed in Sect. 5.3.2), as n tends to infinity. Alternatively, it can be solved by using the linear programming approach [3]:

$$\left\{\begin{array}{l}
\qquad \min \quad x_0 = \displaystyle\sum_{i=0}^{N-1} \sum_{p=1}^{p_{\max}} v_i(p) \\
\text{subject to} \\
\qquad v_i(p) \geq c_i^{(P)} - d_1 + \alpha \displaystyle\sum_{j=0}^{N-1} p_{ji}^{(P)} v_j(p-1), \\
\qquad \text{for} \quad i = 0, \ldots, N-1, p = 1, \ldots, p_{\max}; \\
\qquad v_i(p) \geq c_i^{(NP)} - d_2 + \alpha \displaystyle\sum_{j=0}^{N-1} p_{ji}^{(NP)} v_j(p), \\
\qquad \text{for} \quad i = 0, \ldots, N-1, p = 1, \ldots, p_{\max}.
\end{array}\right.$$



	A	B	C	D	E	F	G	H	I	J	
1	The DP for Solving the Optimal Policy in Infinite Horizon Case										
2	d =	2									
3	Alpha =	0.95					Constraint :				
4		Transition Matrix (Promotion)				Revenue :	p = 4	p = 3	p = 2	p = 1	
5		0.4230471300	0.0992128090	0.0614950400	0.4162450210	6.9744009350	154.9957765427	152.2459799995	149.2901592518	146.1128782580	
6		0.3457871410	0.2109223910	0.2148183280	0.2284721400	18.0913540000	206.0308635335	203.3595897016	200.4881748572	197.4016237469	
7		0.2147216970	0.2033724850	0.4447365850	0.1371692340	43.7531405800	274.8750877934	272.2546231853	269.4378243672	266.4099813036	
8		0.1488606010	0.0266400660	0.0190770870	0.8054222470	0	124.1813302877	121.3081612240	118.2197244625	114.8998914075	
9											
10		Transition Matrix (No Promotion)									
11		0.4146008800	0.0623025190	0.0266821390	0.4964144620	14.0327348000	155.7207608410	153.1351104589	150.3557340753	147.3681172637	144.
12		0.3836771940	0.1743867550	0.1158384660	0.3260975850	51.7172774900	221.3444231792	218.8207206223	216.1079335187	213.1918950852	210.
13		0.2741949630	0.2068815780	0.2808909170	0.2380325420	139.2049217000	342.8361873336	340.3547997244	337.6874980817	334.8203530369	331.
14		0.1063740210	0.0121002430	0.0053227440	0.8762029930	0	123.3820546995	120.6875979189	117.7912628821	114.6779244731	111.
15											
16	Boundary Conditions										
17	v_1(0) =	144.1566579357									
18	v_2(0) =	210.0573770333									
19	v_3(0) =	331.7383916342									
20	v_4(0) =	111.3313250637									
21											
22	Optimal x =	3310.3526409493		Policy							
23	v_1(1) =	147.3681172637		2							
24	v_2(1) =	213.1918950852		2							
25	v_3(1) =	334.8203530369		2							
26	v_4(1) =	114.8998914075		1							
27	v_1(2) =	150.3557340753		2							
28	v_2(2) =	216.1079335187		2							
29	v_3(2) =	337.6874980817		2							
30	v_4(2) =	118.2197244625		1							
31	v_1(3) =	153.1351104589		2							
32	v_2(3) =	218.8207206218		2							
33	v_3(3) =	340.3547997273		2							
34				1							

Fig. 5.3 EXCEL for solving infinite horizon problem with constraints

We note that $v_i(0)$ is not included in the linear programming constraints and the objective function; $v_i(0)$ is solved beforehand using (5.12). A demonstration EXCEL file can be found at the following site (http://hkumath.hku.hk/~wkc/clv3.zip), see also Fig. 5.3 (Taken from [64]).

Tables 5.5 and 5.6 give the optimal values and promotion strategies of the computer service company. For instance, when the promotion cost is 0 and the discount factor is 0.99, then the optimal strategy is that when the current state is

Table 5.5 Optimal promotion strategies and their CLVs

	$d = 0$			$d = 1$			$d = 2$		
α	0.99	0.95	0.90	0.99	0.95	0.90	0.99	0.95	0.90
x_0	11355	3378	2306	11320	3344	2277	11277	3310	2248
$v_0(1)$	610	117	55	609	116	54	608	115	53
$v_1(1)$	645	149	85	644	148	84	643	147	84
$v_2(1)$	713	215	149	712	214	148	711	213	147
$v_3(1)$	837	337	267	836	336	267	845	335	266
$v_0(2)$	616	122	60	614	120	58	612	118	56
$v_1(2)$	650	154	89	648	152	87	647	150	86
$v_2(2)$	718	219	152	716	218	151	714	216	149
$v_3(2)$	842	341	271	840	339	269	839	338	268
$v_1(3)$	656	158	92	654	156	90	650	153	88
$v_2(3)$	724	224	155	722	221	153	718	219	151
$v_3(3)$	848	345	273	846	343	271	842	340	270
$v_0(4)$	628	131	67	624	128	63	620	124	60
$v_1(4)$	662	162	95	658	159	92	654	158	89
$v_2(4)$	730	228	157	726	225	155	722	221	152
$v_3(4)$	854	349	276	850	346	273	846	343	271
$D_0(1)$	P	P	P	P	P	P	P	P	P
$D_1(1)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_2(1)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_3(1)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_0(2)$	P	P	P	P	P	P	P	P	P
$D_1(2)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_2(2)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_3(2)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_0(3)$	P	P	P	P	P	P	P	P	P
$D_1(3)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_3(3)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_0(4)$	P	P	P	P	P	P	P	P	P
$D_1(4)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_2(4)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_3(4)$	NP	NP	NP	NP	NP	NP	NP	NP	NP

$1, 2$ or 3, the promotion should be done when there are some available promotions, i.e.,

$$D_1(p) = D_2(p) = D_3(p) = P \quad \text{for} \quad p = 1, 2, 3, 4$$

and when the current state is 0, no promotion is required, i.e. $D_0(p) = NP$ for $p = 1, 2, 3, 4$. Their corresponding CLVs $v_i(p)$ for different states and different numbers of remaining promotions are also listed (see the first column in the left hand side of Table 5.6.

From Tables 5.5 and 5.6, the optimal strategy for the customers in states 1, 2 and 3 is to conduct no promotion. Moreover, it is not affected by the promotion cost and the discount factor. These results are slightly different from those for the finite horizon case. However, the optimal strategy is to conduct all the four promotions to customers in state 0 as early as possible.

Table 5.6 Optimal promotion strategies and their CLVs

	$d = 3$			$d = 4$			$d = 5$		
α	0.99	0.95	0.90	0.99	0.95	0.90	0.99	0.95	0.90
x_0	11239	3276	2218	11200	3242	2189	11161	3208	2163
$v_0(1)$	607	114	52	606	113	51	605	112	50
$v_1(1)$	641	146	83	641	146	82	640	145	81
$v_2(1)$	710	212	146	709	211	145	708	211	145
$v_3(1)$	834	334	265	833	333	264	832	332	264
$v_0(2)$	610	116	54	608	114	52	606	112	50
$v_1(2)$	645	149	84	643	147	83	641	145	81
$v_2(2)$	713	214	148	711	213	146	709	211	145
$v_3(2)$	837	336	266	835	334	265	833	333	264
$v_0(3)$	613	119	56	610	116	53	607	113	50
$v_1(3)$	647	151	86	645	148	83	642	146	81
$v_2(3)$	715	216	149	713	214	147	710	211	145
$v_3(3)$	839	338	268	837	336	266	834	333	264
$v_0(4)$	616	121	57	612	117	54	608	113	50
$v_1(4)$	650	152	87	646	149	84	643	146	81
$v_2(4)$	718	218	150	714	215	147	711	212	145
$v_3(4)$	842	340	269	838	337	266	835	334	265
$D_0(1)$	P	P	P	P	P	P	P	P	P
$D_1(1)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_2(1)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_3(1)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_0(2)$	P	P	P	P	P	P	P	P	P
$D_1(2)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_2(2)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_3(2)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_0(3)$	P	P	P	P	P	P	P	P	P
$D_1(3)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_2(3)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_3(3)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_0(4)$	P	P	P	P	P	P	P	P	P
$D_1(4)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_2(4)$	NP	NP	NP	NP	NP	NP	NP	NP	NP
$D_3(4)$	NP	NP	NP	NP	NP	NP	NP	NP	NP

5.4 An Extension to Multi-period Promotions

In the previous discussions, the problem under consideration is to decide whether to offer the promotion at the start of each time unit with the assumption that the promotion only lasts for a single time unit. In this section, the analysis is extended to consider multi-period promotions proposed in [145, 150]. A multi-period promotion refers to a promotion that lasts for $2, 3, \ldots, R$ time units. This encourages more purchases or continuous subscriptions than a single-period promotion does. A useful application is the IDD service in a telecommunication company. Customers may think that it is convenient because they can have flexible IDD calls when the promotion period is long, hence a greater revenue can be expected.

The degree of increase in revenue for each period during the r-period of consecutive promotions can be described by the following formula:

$$1 + m(1 - e^{-ni}) \quad i = 1, 2, \ldots, r \tag{5.13}$$

where $m \geq 0$ denotes the ceiling level, $n > 0$ determines the growth rate of the exponential curve and r is an integer which stands for the number of periods of the consecutive promotion with $r \geq 2$. The increase applies in every single period and still applies if the customer changes to another consumption level (different state) under the consecutive promotion. Jain and Singh [124] suggest that the optimal acquisition and retention spending should follow the second term of the formula. We use this formula because we believe that a multi-period promotion has a significant advantage compared to a single-period one at the beginning, but then the revenue increase declines as the length of promotion extends and almost stops if the steady state of the market is reached.

We remark that other revenue adjustments can also be used. Take for example:

- Linear function: $1 + mr/n$, or $1 + m \min\{1, r/n\}$ which sets an upper bound
- S-shaped function: $m((nr + d_1)^{1/3} + d_2)$, where d_1, d_2 are shifts

In our revenue adjustment mechanism, the values of m and n have to be determined. Different values of the ceiling level of adjustment, m, will be tried in the calculations here. However for n, we propose that the increase slows down if it has reached the steady state. With regard to this, we calculate the sum of the absolute differences entry-wise between P^{100} and P^r. When it is smaller than 0.01, then we can say it reaches the "steady state" (roughly speaking) after r steps. Using this r, we want to find the value of n, such that e^{-nr} is smaller than 0.05 (hence the increase has almost ceased). We remark that given an ergodic irreducible Markov chain, the process will converge to its unique stationary distribution. It is well known that the convergence rate depends on the modulus of the second largest eigenvalue of the transition probability matrix being less than 1, i.e. $|\lambda_2| < 1$, see for instance Chap. 2. Roughly speaking, this means the error between P^n and P^∞ will decay at a rate proportional to $|\lambda_2|^n$ (exponential decay). Here $P^\infty = \mathbf{p}\mathbf{1}^t$ where $\mathbf{1}$ is a column vector of all ones and $\mathbf{p}$ is the steady-state probability distribution. Thus this suggests an effective method for determining m and n. Notice that the matrix P we are considering is indeed $P^{(1)}$. Here we show some cases of P^r:

$$P^2 = \begin{pmatrix} 0.7194 & 0.5424 & 0.4056 & 0.3073 \\ 0.1962 & 0.2884 & 0.2994 & 0.2771 \\ 0.0457 & 0.0866 & 0.1286 & 0.1583 \\ 0.0387 & 0.0826 & 0.1664 & 0.2573 \end{pmatrix} \quad P^3 = \begin{pmatrix} 0.6768 & 0.5881 & 0.5035 & 0.4343 \\ 0.2142 & 0.2504 & 0.2672 & 0.2730 \\ 0.0562 & 0.0781 & 0.1015 & 0.1214 \\ 0.0528 & 0.0834 & 0.1278 & 0.1713 \end{pmatrix}$$

$$P^4 = \begin{pmatrix} 0.6544 & 0.6072 & 0.5575 & 0.5147 \\ 0.2221 & 0.2384 & 0.2505 & 0.2589 \\ 0.0619 & 0.0739 & 0.0873 & 0.0991 \\ 0.0616 & 0.0805 & 0.1047 & 0.1273 \end{pmatrix} \quad P^{12} = \begin{pmatrix} 0.6267 & 0.6263 & 0.6257 & 0.6253 \\ 0.2305 & 0.2306 & 0.2308 & 0.2308 \\ 0.0691 & 0.0692 & 0.0694 & 0.0695 \\ 0.0737 & 0.0739 & 0.0741 & 0.0744 \end{pmatrix}.$$

In fact, the steady-state probability distribution is given by

$$(0.626, 0.2306, 0.0692, 0.0738)^T.$$

We note that the eigenvalues of P are $1.0000, 0.0519, 0.2664$ and 0.5656. The second largest eigenvalue is 0.5656. This means that the Markov chain process in this case will converge very fast to its steady-state distribution. Therefore we find that n should be greater than 0.2496. We take $n = 0.25$. In the next section we proceed to find the maximum expected profit using stochastic dynamic programming.

5.4.1 Stochastic Dynamic Programming Models

The optimization problem with multi-period promotion is an extension of our previous problem of solving for the optimal promotion strategy with only single-period promotions. It can also be tackled by applying the stochastic dynamic programming models because the basic features of the problem are preserved: customers may change their states with time and the company may take certain actions from a well-defined set with these actions affecting the transition probabilities of the next time unit, incurring immediate or subsequent gain or loss to the company. In the following subsections, two CLV models are under investigation. They are infinite horizon without constraints and finite horizon under the constraint that the number of promotions is fixed.

5.4.2 The Infinite Horizon Without Constraints

We solve this new optimization problem with multi-period promotions by introducing the revenue adjustment into the analysis of the basic case with single-period promotions. Recall that in the basic case, it has been demonstrated that the values of v_i which stands for the maximum expected profit starting from State i should satisfy (5.6), and that the problem of determining the v_i values can be transformed into the linear programming problem as shown in (5.8). In fact, (5.8) can be re-written in the following matrix form with $\mathbf{v} = [v_0\ v_1, \cdots, v_{n-1}]^T$ and $\mathbf{c}^{(k)} = [c_0^{(k)}\ c_1^{(k)}, \cdots, c_{n-1}^{(k)}]^T$:

$$\begin{cases} \qquad\quad \min \quad x_0 = \sum_{i=0}^{N-1} v_i \\ \text{subject to} \\ \qquad\quad \mathbf{v} \geq \mathbf{c}^{(k)} - d_j \mathbf{1} + \alpha \mathbf{v} P^{(k)}, \\ \qquad\quad \mathbf{v} \geq \mathbf{0}. \end{cases} \tag{5.14}$$

Here $\mathbf{1}$ is a matrix with all entries 1, $\mathbf{0}$ is a matrix with all entries 0, and both of them have the same size as $\mathbf{v}$.

To proceed with our revenue adjustment, one more notation is introduced as follows:

$$\mathbf{c}'^{(k)}(r) = (1 + m(1 - e^{-nr}))\mathbf{c}^{(k)} - d_j \mathbf{1},$$

where both $m \geq 0$ and $n > 0$ are fixed; $r = 2, \ldots, R$ with R being the maximum duration allowed for a consecutive promotion; and $k \in A_i$ with $k \neq A_{i0}$ where A_{i0} represents the alternative of taking no promotion when the customer is currently in State i.

Now we consider a policy using alternative k which lasts for 2 time units. Following from the one-step removal policy, its intermediate form after 1 time unit is:

$$\mathbf{v} = \mathbf{c}'^{(k)}(2) + \alpha \mathbf{v}' P^{(k)}$$

where $\mathbf{v}'$ denotes a vector of expected profit in infinite horizon given that no action can be made in the first step.

As we have no choice but to select other alternatives after 1 time unit since the promotion lasts for 2 time units, we have to use the matrix $P^{(k)}$ for transition and the same revenue $\mathbf{c}'^{(k)}(2)$ will be earned. Again, from the one-step removal policy, we have:

$$\mathbf{v}' = \mathbf{c}'^{(k)}(2) + \mathbf{v} \alpha P^{(k)}.$$

Combining the two, we have

$$\mathbf{v} = \mathbf{c}'^{(k)}(2) + \alpha P^{(k)} \Big(\mathbf{c}'^{(k)}(2) + \mathbf{v} \alpha P^{(k)} \Big)$$

or

$$\mathbf{v} = \Big(I + \alpha P^{(k)} \Big) \mathbf{c}'^{(k)}(2) + \mathbf{v} (\alpha P^{(k)})^2.$$

Applying the similar argument inductively and changing it into an inequality, it can be generalized as:

$$\mathbf{v} \geq \Big(I + \alpha P^{(k)} + \cdots + (\alpha P^{(k)})^{r-1} \Big) \mathbf{c}'^{(k)}(r) + \mathbf{v} (\alpha P^{(k)})^r.$$

Hence the linear programming problem we should ultimately consider is the following:

Table 5.7 Optimal stationary policies and their CLVs with $d = 0$

	m	v_0	v_1	v_2	v_3	D_0	D_1	D_2	D_3
$\alpha = 0.90$	0.0	92	119	179	297	1	1	0	0
	0.5	92	119	179	297	1	1	0	0
	1.0	93	121	180	298	1	2	0	0
	1.5	101	134	189	307	1	4	0	0
	2.0	111	150	209	320	1	4	4	0
$d = 0\,\alpha = 0.95$	0.0	205	234	295	415	1	1	0	0
	0.5	205	234	295	415	1	1	0	0
	1.0	207	237	297	418	1	2	0	0
	1.5	225	261	318	437	1	4	0	0
	2.0	250	292	354	466	1	4	4	0
$\alpha = 0.99$	0.0	1113	1144	1206	1329	1	1	0	0
	0.5	1113	1144	1206	1329	1	1	0	0
	1.0	1126	1158	1220	1342	1	2	0	0
	1.5	1231	1269	1327	1448	1	4	4	0
	2.0	1370	1414	1481	1592	1	4	4	0

$$\left\{\begin{array}{l} \qquad\qquad \min \quad x_0 = \displaystyle\sum_{i=0}^{N-1} v_i \\ \text{subject to} \\ \qquad\qquad \mathbf{v} \geq \mathbf{c}^{(k)} - d_k \mathbf{1} + \alpha \mathbf{v} P^{(k)}, \\ \qquad\qquad \mathbf{v} \geq (I + \alpha P^{(k)} + \cdots + (\alpha P^{(k)})^{r-1}) \mathbf{c}'^{(k)}(r) + \mathbf{v}(\alpha P^{(k)})^{r}, \\ \qquad\qquad\quad \text{for} \quad r = 2, \ldots, R, \\ \qquad\qquad \mathbf{v} \geq \mathbf{0}. \end{array}\right. \tag{5.15}$$

Some numerical examples are shown here with the value of R being confined to 4. The numerical results are computed using *Scilab* and the program code is available at the following site (http://hkumath.hku.hk/~wkc/MDP2.zip). We denote D_i to be the optimal plan taken when a customer is in State i. Here 0 means no promotion is conducted in the time period and 1 represents a single-period promotion execution, and so on. We have set $|A_i| = 2$ for $i = 0, 1, \ldots, N - 1$ (the only alternatives of actions are either to conduct a promotion or not to conduct a promotion) and $N = 4$ (all customers are classified into 4 states as in the basic case). We remark that when $m = 0$, the situation reduces to the case studied in [64]. Tables 5.7–5.9 present the optimal solutions.

5.4.3 Finite Horizon with Hard Constraints

Here we consider optimization in finite horizon to better model those industries with short product life cycles. In this case, we set a limit to the number of promotions

Table 5.8 Optimal stationary policies and their CLVs with $d = 2$

		m	v_0	v_1	v_2	v_3	D_0	D_1	D_2	D_3
$d = 2$	$\alpha = 0.90$	0.0	74	102	164	282	1	1	0	0
		0.5	74	102	164	282	1	1	0	0
		1.0	74	102	164	282	1	2	0	0
		1.5	82	115	173	291	1	4	0	0
		2.0	92	131	189	303	1	4	4	0
	$\alpha = 0.95$	0.0	169	199	262	383	1	1	0	0
		0.5	169	199	262	383	1	1	0	0
		1.0	170	200	263	384	1	2	0	0
		1.5	188	224	283	403	1	4	0	0
		2.0	211	253	315	429	1	4	4	0
	$\alpha = 0.99$	0.0	934	966	1030	1153	1	1	0	0
		0.5	934	966	1030	1153	1	1	0	0
		1.0	943	976	1040	1163	1	2	0	0
		1.5	1042	1080	1140	1263	1	4	0	0
		2.0	1175	1220	1286	1400	1	4	4	0

Table 5.9 Optimal stationary policies and their CLVs with $d = 4$

		m	v_0	v_1	v_2	v_3	D_0	D_1	D_2	D_3
$d = 4$	$\alpha = 0.90$	0.0	58	88	151	269	1	0	0	0
		0.5	58	88	151	269	1	0	0	0
		1.0	58	88	151	269	1	0	0	0
		1.5	63	96	157	275	1	4	0	0
		2.0	72	111	170	286	1	4	4	0
	$\alpha = 0.95$	0.0	134	164	230	351	1	0	0	0
		0.5	134	164	230	351	1	0	0	0
		1.0	134	164	230	351	1	0	0	0
		1.5	150	186	247	369	1	4	0	0
		2.0	172	214	276	393	1	4	4	0
	$\alpha = 0.99$	0.0	755	788	854	978	1	1	0	0
		0.5	755	788	854	978	1	1	0	0
		1.0	761	793	860	983	1	2	0	0
		1.5	855	893	955	1079	1	4	4	0
		2.0	981	1025	1091	1208	1	4	4	0

available for each customer in a period of planning. We will use the values obtained in the last subsection as the boundary conditions. Some additional notation is introduced here:

1. w, number of weeks remaining
2. p, number of promotions remaining
3. q, number of weeks remaining until the next decision echo
4. r, currently under a r-period promotion

As before, we consider the case $M = 2$ and $N = 4$. The recursive relation is given by:

$$v_i(w,p,q,r) = \begin{cases} \max\limits_{t=0,\ldots,\min\{w,p\}} \left\{ \widetilde{c}_i(t) + \alpha \sum\limits_{k=1}^{N} p_{ki}^{(m_1)} v_i(w-1, m_2, m_3, m_4) \right\} & \text{if } q = 0 \\ \widetilde{c}_i(r) + \alpha \sum\limits_{k=1}^{N} p_{ki}^{(1)} v_i(w-1, p-1, q-1, r) & \text{if } q \geq 2 \\ \widetilde{c}_i(r) + \alpha \sum\limits_{k=1}^{N} p_{ki}^{(1)} v_i(w-1, p-1, 0, 0) & \text{if } q = 1 \end{cases}$$

where

$$m_1 = \begin{cases} 2 \text{ if } t = 0 \\ 1 \text{ otherwise} \end{cases} \qquad m_2 = \begin{cases} p & \text{if } t = 0 \\ p-1 & \text{otherwise} \end{cases}$$

$$m_3 = \begin{cases} 0 & \text{if } t = 0 \\ t-1 & \text{otherwise} \end{cases} \qquad m_4 = \begin{cases} 0 \text{ if } t = 0,\ 1 \\ t \text{ otherwise} \end{cases}$$

and

$$\widetilde{c}_i(t) = \begin{cases} (1 + m(1 - e^{-nr}))c_i^{(1)} - d_1 & \text{if } t \geq 2 \\ c_i^{(1)} - d_1 & \text{if } t = 1 \\ c_i^{(2)} - d_2 & \text{if } t = 0 \end{cases}$$

where $m \geq 0$ and $n > 0$ are fixed.

In our promotion planning, we set $w_{\max} = 52$ and $p_{\max} = 4$. The solution is presented in the following form:

$$(t_1, t_2, t_3, t_4, v^*)$$

where v^* is the maximum expected profit, t_i is the week to employ a promotion, "-" represents as no promotion and being enclosed by square brackets represents a consecutive promotion. For example, $([1,2], 50, 52, 83)$ means we should have a consecutive promotion in weeks 1 and 2, i.e., a two-period promotion in week 1, and two single-period promotions in weeks 50 and 52, and that the maximum expected profit is 83 (Tables 5.10–5.12).

From the numerical results, we observe that there is no need to conduct promotion to customers in state 3. For customers in state 2, promotion is useful only when m is large. For customers in state 0, generally speaking, promotion should be conducted in the very beginning in order to keep the customer. If we let the value of m increase, when it is greater than a certain threshold, the promotion pattern differs from that with $m = 0$ (the situation where consecutive promotions have no beneficial effect). From that level onwards, we can say consecutive promotions have a significant benefit over single-period promotions. Also, when m is large,

Table 5.10 Optimal promotion strategies and their CLVs when $d = 0$

α	m	State 0	State 1	State 2	State 3
0.9	0	$(1, 2, 3, 4, 67)$	$(1, 47, 50, 52, 95)$	$(-, -, -, -, 158)$	$(-, -, -, -, 276)$
	0.5	$(1, 2, 3, 4, 67)$	$(1, 47, 50, 52, 95)$	$(-, -, -, -, 158)$	$(-, -, -, -, 276)$
	1	$(1, 2, 3, 4, 67)$	$([1, 2], [50, 51], 95)$	$(-, -, -, -, 158)$	$(-, -, -, -, 276)$
	1.5	$(1, 2, 3, 4, 68)$	$([1, 2, 3, 4], 101)$	$(-, -, -, -, 161)$	$(-, -, -, -, 279)$
	2	$([1, 2, 3, 4], 70)$	$([1, 2, 3, 4], 109)$	$([1, 2, 3, 4], 168)$	$(-, -, -, -, 284)$
0.95	0	$(1, 2, 3, 4, 138)$	$(45, 48, 50, 52, 169)$	$(-, -, -, -, 234)$	$(-, -, -, -, 355)$
	0.5	$(1, 2, 3, 4, 138)$	$(45, 48, 50, 52, 169)$	$(-, -, -, -, 234)$	$(-, -, -, -, 355)$
	1	$(1, 2, 3, 4, 138)$	$([45, 46], [50, 51], 169)$	$(-, -, -, -, 234)$	$(-, -, -, -, 356)$
	1.5	$(1, 2, 3, 4, 140)$	$([1, 2, 3, 4], 177)$	$(-, -, -, -, 239)$	$(-, -, -, -, 361)$
	2	$([1, 2, 3, 4], 145)$	$([1, 2, 3, 4], 187)$	$([1, 2, 3, 4], 249)$	$(-, -, -, -, 368)$
0.99	0	$(1, 2, 3, 4, 929)$	$(47, 49, 50, 51, 963)$	$(-, -, -, -, 1031)$	$(-, -, -, -, 1155)$
	0.5	$(1, 2, 3, 4, 929)$	$(47, 49, 50, 51, 963)$	$(-, -, -, -, 1031)$	$(-, -, -, -, 1155)$
	1	$(1, 2, 3, 4, 937)$	$([47, 48], [50, 51], 971)$	$(-, -, -, -, 1039)$	$(-, -, -, -, 1162)$
	1.5	$([48, 49, 50, 51], 1003)$	$([1, 2, 3, 4], 1039)$	$([49, 50, 51, 52], 1106)$	$(-, -, -, -, 1230)$
	2	$([49, 50, 51, 52], 1094)$	$([1, 2, 3, 4], 1132)$	$([20, 21, 22, 23], 1198)$	$(-, -, -, -, 1322)$

Table 5.11 Optimal promotion strategies and their CLVs when $d = 2$

α	m	State 0	State 1	State 2	State 3
0.9	0	(1, 2, 3, 4, 60)	(49, 50, 51, 52, 89)	(−, −, −, −, 152)	(−, −, −, −, 271)
	0.5	(1, 2, 3, 4, 60)	(49, 50, 51, 52, 89)	(−, −, −, −, 152)	(−, −, −, −, 271)
	1	(1, 2, 3, 4, 60)	([49, 50], [51, 52], 89)	(−, −, −, −, 152)	(−, −, −, −, 271)
	1.5	(1, 2, 3, 4, 61)	([1, 2, 3, 4], 95)	(−, −, −, −, 155)	(−, −, −, −, 274)
	2	([1, 2, 3, 4], 63)	([1, 2, 3, 4], 102)	([1, 2, 3, 4], 161)	(−, −, −, −, 278)
0.95	0	(1, 2, 3, 4, 128)	(48, 50, 51, 52, 160)	(−, −, −, −, 225)	(−, −, −, −, 347)
	0.5	(1, 2, 3, 4, 128)	(48, 50, 51, 52, 160)	(−, −, −, −, 225)	(−, −, −, −, 347)
	1	(1, 2, 3, 4, 128)	([48, 49], [51, 52], 160)	(−, −, −, −, 225)	(−, −, −, −, 347)
	1.5	(1, 2, 3, 4, 130)	([1, 2, 3, 4], 167)	(−, −, −, −, 230)	(−, −, −, −, 351)
	2	([49, 50, 51, 52], 136))	([1, 2, 3, 4], 177)	([1, 2, 3, 4], 239)	(−, −, −, −, 359)
0.99	0	(1, 2, 3, 4, 815)	(48, 49, 51, 52, 849)	(−, −, −, −, 917)	(−, −, −, −, 1041)
	0.5	(1, 2, 3, 4, 815)	(48, 49, 51, 52, 849)	(−, −, −, −, 917)	(−, −, −, −, 1041)
	1	(1, 2, 3, 4, 821)	([48, 49], [51, 52], 855)	(−, −, −, −, 923)	(−, −, −, −, 1047)
	1.5	([49, 50, 51, 52], 883)	([1, 2, 3, 4], 920)	(−, −, −, −, 987)	(−, −, −, −, 1110)
	2	([49, 50, 51, 52], 971)	([1, 2, 3, 4], 1008)	([34, 35, 36, 37], 1075)	(−, −, −, −, 1199)

Table 5.12 Optimal promotion strategies and their CLVs when $d = 4$

α	m	State 0	State 1	State 2	State 3
0.9	0	$(1, 2, 3, 4, 54)$	$(-, -, -, -, 84)$	$(-, -, -, -, 147)$	$(-, -, -, -, 266)$
	0.5	$(1, 2, 3, 4, 54)$	$(-, -, -, -, 84)$	$(-, -, -, -, 147)$	$(-, -, -, -, 266)$
	1	$(1, 2, 3, 4, 54)$	$(-, -, -, -, 84)$	$(-, -, -, -, 147)$	$(-, -, -, -, 266)$
	1.5	$(1, 2, 3, 4, 54)$	$([1, 2, 3, 4], 88)$	$(-, -, -, -, 149)$	$(-, -, -, -, 268)$
	2	$([49, 50, 51, 52], 57)$	$([1, 2, 3, 4], 95)$	$([41, 42, 43, 44], 154)$	$(-, -, -, -, 273)$
0.95	0	$(1, 2, 3, 4, 119)$	$(-, -, -, -, 151)$	$(-, -, -, -, 217)$	$(-, -, -, -, 338)$
	0.5	$(1, 2, 3, 4, 119)$	$(-, -, -, -, 151)$	$(-, -, -, -, 217)$	$(-, -, -, -, 338)$
	1	$(1, 2, 3, 4, 119)$	$(-, -, -, -, 151)$	$(-, -, -, -, 217)$	$(-, -, -, -, 338)$
	1.5	$(48, 49, 50, 51, 121)$	$([1, 2, 3, 4], 157)$	$(-, -, -, -, 221)$	$(-, -, -, -, 342)$
	2	$([49, 50, 51, 52], 128)$	$([1, 2, 3, 4], 167)$	$([1, 2, 3, 4], 229)$	$(-, -, -, -, 350)$
0.99	0	$(1, 2, 3, 4, 701)$	$(49, 50, 51, 52, 736)$	$(-, -, -, -, 804)$	$(-, -, -, -, 928)$
	0.5	$(1, 2, 3, 4, 701)$	$(49, 50, 51, 52, 736)$	$(-, -, -, -, 804)$	$(-, -, -, -, 928)$
	1	$(1, 2, 3, 4, 705)$	$([49, 50], [51, 52], 739)$	$(-, -, -, -, 807)$	$(-, -, -, -, 931)$
	1.5	$([49, 50, 51, 52], 765)$	$([1, 2, 3, 4], 801)$	$(-, -, -, -, 868)$	$(-, -, -, -, 992)$
	2	$([49, 50, 51, 52], 849)$	$([1, 2, 3, 4], 885)$	$([39, 40, 41, 42], 952)$	$(-, -, -, -, 1076)$

either no promotion is used or 4-period promotions are employed. In other words, although consecutive promotion gives up the opportunity to observe the states after a one period transition and set strategy accordingly, the benefit derived exceeds significantly the cost of the opportunity we lost.

Finally, we remark that the new model actually provides better objective values when compared to the previous model.

5.5 Higher-Order Markov Decision Process

The MDP presented in the previous section is a first-order type, i.e., the transition probabilities depend on the current state only. A brief introduction was given in Chap. 1. For the Higher-order Markov Decision Process (HMDP), the transition probabilities depend on the current state and a number of previous states. For instance, the probabilities of a second-order MDP moving from state s_i to state s_j depends only on the latest two states, the present state s_i and the previous state s_h. The transition probability is denoted by p_{hij}. In this section, we are interested in studying a HMDP with applications to the CLV problems.

In the infinite horizon case, there are an infinite number of policies with the initial state s_i and the previous state s_h. The policy D prescribes an alternative, say k^*, for the transition out of states s_h and state s_i. The probability of being in state s_j after one transition is $p_{hij}^{(k*)}$ and this probability is re-written as $p(1, j)$. Now using the alternatives directed by D, one can calculate the probabilities of being in the various states after two transitions; these probabilities can be denoted by

$$p(2,l) \quad \text{for} \quad l = 0, 1, \ldots, N-1.$$

Similarly one can calculate the probability $p(n, j)$ of being in state s_i and state s_h after n transitions. Denote $D(n, h, i)$ the alternative that D prescribes for use after n transitions if the system is in state s_j, the expected reward to be earned by D on the $(n + 1)$-transition would be

$$\sum_{j=0}^{N-1} p(n, j) q_j^{D(n,h,i)}, \tag{5.16}$$

and the present value of this sum is

$$\alpha^n \sum_{j=0}^{N-1} p(n, j) q_j^{D(n,h,i)}, \tag{5.17}$$

thus the total expected reward of D is given by

$$q_i^{(k*)} + \sum_{n=1}^{\infty} \alpha^n \sum_{j=0}^{N-1} p(n, j) q_j^{D(n,h,i)}, \tag{5.18}$$

choosing Q such that

$$|q_l^{(k)}| \le Q \quad \text{for all} \quad l = 0, 1, \ldots, N-1, \tag{5.19}$$

and $k \in A_i$, so that the sum is absolutely convergent. This sum is called the value of the policy D, and it is denoted by $w_{hi}(D)$. It is clear that

$$|w_{hi}(D)| \le Q(1-\alpha)^{-1}. \tag{5.20}$$

5.5.1 Stationary Policy

A stationary policy is a policy where the choice of an alternative depends only on the state the system is in and is independent of n. $D(h,i)$ is defined to be the stationary policy with the current state s_i and the previous s_h. For a Markov decision process with infinite horizon and discount factor α, $0 < \alpha < 1$, the value of an optimal policy is defined as follows:

$$v_{hi} = \text{ lub } \{w_{hi}(D) | D \text{ a policy with initial state } s_i \text{ and previous state } s_h\} \tag{5.21}$$

where lub is the standard abbreviation for least upper bound.

Proposition 5.1. *For a Markov decision process with infinite horizon, discount factor α, where $0 < \alpha < 1$,*

$$u_{hi} = \max_{k \in A_i} \left\{ q_i^{(k)} + \alpha \sum_{j=0}^{N-1} p_{hij}^{(k)} v_{ij} \right\}, \quad h, i = 0, 1, \ldots, N-1. \tag{5.22}$$

Then, for each h, i, $u_{hi} = v_{hi}$.

Proof. Fixing $h, i = 0, 1, \ldots, N-1$, let D be any policy with initial state s_i and previous state s_h. Suppose that D prescribes the alternative k^* on the first transition out of s_h, s_i, and $\bar{D}_{ij}$ denotes the associated one-step-removed policy. Thus

$$\begin{aligned} w_{hi}(D) &= q_i^{(k^*)} + \alpha \sum_{j=0}^{N-1} p_{hij}^{(k^*)} w_{ij}(\bar{D}_{ij}) \\ &\le q_i^{(k^*)} + \alpha \sum_{j=0}^{N-1} p_{hij}^{(k^*)} v_{ij} \\ &\le \max_{k \in A_i} \{ q_i^{(k)} + \alpha \sum_{j=0}^{N-1} p_{hij}^{(k)} v_{ij} \} = u_{hi}. \end{aligned}$$

Therefore u_{hi} is an upper bound for the set

$$\{w_{hi}(D)|D \text{ a policy with initial state } s_i \text{ previous state } s_h\}$$

and

$$v_{hi} = \text{ lub } \{w_{hi}(D)\} \leq u_{hi}$$

where lub stands for least upper bound. Consider an alternative k_{hi} such that

$$u_{hi} = \max_{k \in A_i}\{q_i^{(k)} + \alpha \sum_{j=0}^{N-1} p_{hij}^{(k)} v_{ij}\} = q_i^{(k_{hi})} + \alpha \sum_{j=0}^{N-1} p_{hij}^{(k_{hi})} v_{ij}.$$

For any given $\epsilon > 0$ and for each j, a policy D_{hi}^* is chosen with initial state s_i and previous state s_h such that

$$v_{hi} - \epsilon < w_{hi}(D_{hi}^*).$$

Define a policy D with initial state s_i and previous state s_h as follows: use alternative k_{hi} out of states s_h and state s_i, then for each h, i if the system moves to state s_j on the first transition, policy D_{ij}^* is used thereafter. We have

$$\begin{aligned}
u_{hi} &= q_i^{(k_{hi})} + \alpha \sum_{j=0}^{N-1} p_{hij}^{(k_{hi})} v_{ij} \\
&\leq q_i^{(k_{hi})} + \alpha \sum_{j=0}^{N-1} p_{hij}^{(k_{hi})} (w_{ij}(D_{ij}^*) + \epsilon) \\
&= q_i^{(k_{hi})} + \alpha \sum_{j=0}^{N-1} p_{hij}^{(k_{hi})} w_{ij}(D_{ij}^*) + \alpha\epsilon \sum_{j=0}^{N-1} p_{hij}^{(k_{hi})} \\
&= w_{hi}(D) + \alpha\epsilon \\
&< v_{hi} + \epsilon.
\end{aligned}$$

Since ϵ is arbitrary, $u_{hi} \leq v_{hi}$. The result follows.

Proposition 5.2. *(Stationary Policy Theorem) Given a Markov decision process with infinite horizon and discount factor α, $0 < \alpha < 1$, choose, for each h, i, an alternative k_{hi} such that*

$$\max_{k \in A_i}\{q_i^{(k)} + \alpha \sum_{j=0}^{N-1} p_{hij}^{(k)} v_{ij}\} = q_i^{(k_{hi})} + \alpha \sum_{j=0}^{N-1} p_{hij}^{(k_{hi})} v_{ij}.$$

Define the stationary policy D by $D(h, i) = k_{hi}$. Then for each h, i, $w_{hi}(D) = v_{hi}$.

Proof. Since

$$v_{hi} = q_i^{(k_{hi})} + \alpha \sum_{j=0}^{N-1} p_{hij}^{(k_{hi})} v_{ij},$$

we have

$$\mathbf{v} = \mathbf{q} + \alpha P \mathbf{v}$$

where

$$\mathbf{v} = [v_{0,0}, v_{0,1}, \ldots v_{0,N-1}, v_{1,0}, \ldots v_{N-1,N-1}]^T,$$
$$\mathbf{q} = [q_0, q_1, \ldots, q_{N-1}, q_0, \ldots, q_{N-1}]^T,$$

and

$$P = [p_{hij}^{(k_{hi})}].$$

The superscripts are omitted in the above vectors. For $0 < \alpha < 1$, the matrix $(I - \alpha P)$ is non-singular and the result follows.

According to the above two propositions, the optimal stationary policy can be obtained by solving the following LP problem:

$$\begin{cases} \quad \min \{x_{0,0} + x_{0,1} + \cdots + x_{0,N-1} + x_{1,0} + \cdots + x_{N-1,N-1}\} \\ \text{subject to} \\ \quad x_{hi} \geq q_i^{(k)} + \alpha \displaystyle\sum_{j=0}^{N-1} p_{hij}^{(k)} x_{ij}, \quad h, i = 0, 1, \ldots, N-1, \\ \quad k \in A_i. \end{cases} \tag{5.23}$$

5.5.2 Application to the Calculation of CLV

In previous sections, a first-order MDP is applied to a computer service company. In this section, the same set of customer data is used with the HMDP. A comparison of the two models can be found in Ching et al. [62].

The one-step transition probabilities are given in Sect. 5.3. Similarly, one can estimate the second-order (two-step) transition probabilities. Given the current state i and previous state h, the number of customers switching to state j is recorded. Then, divide it by the total number of customers in the current state i and previous state h. The values obtained are the second-order transition probabilities. The transition probabilities under the promotion and no-promotion period are given respectively in Table 5.13.

Table 5.13 The second-order transition probabilities

	Promotion				No-promotion			
States	0	1	2	3	0	1	2	3
(0,0)	0.8521	0.1225	0.0166	0.0088	0.8957	0.0904	0.0098	0.0041
(0,1)	0.5873	0.3258	0.0549	0.0320	0.6484	0.3051	0.0329	0.0136
(0,2)	0.4471	0.3033	0.1324	0.1172	0.5199	0.3069	0.0980	0.0753
(0,3)	0.3295	0.2919	0.1482	0.2304	0.4771	0.2298	0.1343	0.1587
(1,0)	0.6739	0.2662	0.0394	0.0205	0.7287	0.2400	0.0227	0.0086
(1,1)	0.3012	0.4952	0.1661	0.0375	0.3584	0.5117	0.1064	0.0234
(1,2)	0.1915	0.4353	0.2169	0.1563	0.2505	0.4763	0.1860	0.0872
(1,3)	0.1368	0.3158	0.2271	0.3203	0.1727	0.3750	0.2624	0.1900
(2,0)	0.5752	0.2371	0.1043	0.0834	0.6551	0.2253	0.0847	0.0349
(2,1)	0.2451	0.4323	0.2043	0.1183	0.3048	0.4783	0.1411	0.0757
(2,2)	0.1235	0.3757	0.2704	0.2304	0.2032	0.3992	0.2531	0.1445
(2,3)	0.1030	0.2500	0.2630	0.3840	0.1785	0.2928	0.2385	0.2901
(3,0)	0.4822	0.2189	0.1496	0.1494	0.6493	0.2114	0.0575	0.0818
(3,1)	0.2263	0.3343	0.2086	0.2308	0.2678	0.4392	0.1493	0.1437
(3,2)	0.1286	0.2562	0.2481	0.3671	0.2040	0.3224	0.2434	0.2302
(3,3)	0.0587	0.1399	0.1855	0.6159	0.1251	0.1968	0.1933	0.4848

The transition probability from state 0 to state 0 is very high in the first-order model for both promotion and no-promotion periods. However, in the second-order model, the transition probabilities

$$(0,0) \rightarrow 0, (1,0) \rightarrow 0, (2,0) \rightarrow 0 \quad \text{and} \quad (3,0) \rightarrow 0$$

are very different. It is clear that the second-order Markov chain model can better capture the customers' behavior than the first-order Markov chain model.

In Tables 5.14–5.16, the optimal stationary policy is given for the first-order and the second-order MDP respectively for different values of the discount factor α and promotion cost d. Once again, (P) represents conducting a promotion and (NP) represents to conduct no promotion. It is found that the optimal stationary policies for both models are consistent in the sense that $D_i = D_{ii}$ for $i = 0, 1, 2, 3$ in all the tested cases. For the second-order case, the optimal stationary policy D_{ii} depends not only on states (the optimal policy depends on the current state only in the first-order model) but also on the values of α and d. It is observed that the second-order Markov decision process always gives better objective value.

5.6 Summary

In this chapter, stochastic dynamic programming models are proposed for the optimization of CLV. Both infinite horizon and finite horizon with budget constraints are discussed. The former can be solved by using linear programming techniques,

Table 5.14 Optimal strategies when the first-order MDP is used

	$d = 0$			$d = 1$			$d = 2$		
α	0.99	0.95	0.90	0.99	0.95	0.90	0.99	0.95	0.90
x_0	4791	1149	687	4437	1080	654	4083	1012	621
v_0	1112	204	92	1023	186	83	934	168	74
v_1	1144	234	119	1054	216	110	965	198	101
v_2	1206	295	179	1118	278	171	1030	261	163
v_3	1328	415	296	1240	399	289	1153	382	281
D_0	P	P	P	P	P	P	P	P	P
D_1	P	P	P	P	P	P	P	P	P
D_2	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_3	NP	NP	NP	NP	NP	NP	NP	NP	NP
	$d = 3$			$d = 4$			$d = 5$		
α	0.99	0.95	0.90	0.99	0.95	0.90	0.99	0.95	0.90
x_0	3729	943	590	3375	879	566	3056	827	541
v_0	845	151	65	755	134	58	675	119	51
v_1	877	181	94	788	164	88	707	151	82
v_2	942	245	156	854	230	151	775	217	145
v_3	1066	366	275	978	351	269	899	339	264
D_0	P	P	P	P	P	P	P	P	P
D_1	P	P	NP	P	NP	NP	NP	NP	NP
D_2	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_3	NP	NP	NP	NP	NP	NP	NP	NP	NP

the later problem can be solved by using a dynamic programming approach. In both cases, they can be implemented easily in an EXCEL spreadsheet. The models are then applied to practical data of a computer service company. The company makes use of the proposed CLV model to make and maintain value-laden relationships with the customers. We also extend the idea of MDP to a higher-order setting. Optimal stationary policy is also obtained in this case.

Further research can be done in promotion strategy through advertising. Advertising is an important tool in modern marketing. The purpose of advertising is to enhance potential users' responses to the company. This is achieved by providing information for choosing a particular product or service. A number of marketing models can be found in Lilien et al. [148] and the references therein. It has been shown that a pulsation advertising policy is effective, see Mesak et al. [159–162] and Ching et al. [73]. It will be interesting to incorporate the pulsation advertising policy in the CLV model.

Table 5.15 Optimal strategies when the second-order MDP is used

	$d = 0$			$d = 1$			$d = 2$		
α	0.99	0.95	0.90	0.99	0.95	0.90	0.99	0.95	0.90
x_0	19001	5055	3187	17578	4785	3066	16154	4520	2950
v_{00}	1034	177	74	943	158	65	853	140	56
v_{01}	1081	217	108	991	200	100	901	182	93
v_{02}	1168	299	184	1080	282	177	991	266	170
v_{03}	1309	433	312	1220	417	305	1132	401	298
v_{10}	1047	188	83	956	169	74	866	152	66
v_{11}	1110	242	129	1020	224	120	930	207	112
v_{12}	1195	322	204	1107	306	196	1019	290	190
v_{13}	1347	466	339	1259	450	333	1171	434	326
v_{20}	1071	209	102	981	191	93	891	174	85
v_{21}	1135	265	149	1046	247	141	957	230	133
v_{22}	1217	341	221	1129	325	214	1041	310	207
v_{23}	1370	487	358	1283	471	352	1195	456	345
v_{30}	1094	230	120	1004	212	112	915	195	104
v_{31}	1163	290	171	1074	273	163	985	256	156
v_{32}	1239	359	236	1151	343	229	1062	327	223
v_{33}	1420	531	398	1333	516	391	1245	501	385
D_{00}	P	P	P	P	P	P	P	P	P
D_{01}	P	P	P	P	P	NP	P	NP	NP
D_{02}	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_{03}	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_{10}	P	P	P	P	P	P	P	P	P
D_{11}	P	P	P	P	P	P	P	P	P
D_{12}	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_{13}	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_{20}	P	P	P	P	P	P	P	P	P
D_{21}	P	P	P	P	P	P	P	P	P
D_{22}	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_{23}	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_{30}	P	P	P	P	P	P	P	P	P
D_{31}	P	P	P	P	P	P	P	P	P
D_{32}	P	NP	NP	P	NP	NP	P	NP	NP
D_{33}	NP	NP	NP	NP	NP	NP	NP	NP	NP

5.7 Exercises

1. Consider an online gaming company that plans to stay in business for 4 more years and then it will be shut down without any salvage value. Each year, the volume of players depends only on the volume in the last year, and it is classified as either high or low. If a high volume of players occurs, the expected profit for

Table 5.16 Optimal strategies when the second-order MDP is used

	$d = 3$			$d = 4$			$d = 5$		
α	0.99	0.95	0.90	0.99	0.95	0.90	0.99	0.95	0.90
x_0	14731	4277	2858	13572	4148	2825	13224	4093	2791
v_{00}	763	124	50	690	117	49	670	115	48
v_{01}	811	167	87	739	159	86	717	156	84
v_{02}	902	251	164	830	243	162	809	240	160
v_{03}	1044	386	293	972	378	290	951	375	288
v_{10}	776	135	59	703	127	57	682	124	55
v_{11}	841	191	107	768	182	105	745	179	103
v_{12}	930	275	184	858	267	182	836	263	180
v_{13}	1083	420	321	1012	412	319	990	409	317
v_{20}	801	158	79	728	150	77	707	146	74
v_{21}	867	214	127	794	206	124	771	201	121
v_{22}	953	295	202	881	287	200	859	284	198
v_{23}	1107	442	340	1035	434	338	1014	430	336
v_{30}	825	179	97	752	171	95	731	167	93
v_{31}	896	240	149	823	231	147	800	227	144
v_{32}	973	313	218	901	305	216	879	301	213
v_{33}	1158	487	381	1087	480	379	1065	476	377
D_{00}	P	P	NP	NP	NP	NP	NP	NP	NP
D_{01}	P	NP	NP	NP	NP	NP	NP	NP	NP
D_{02}	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_{03}	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_{10}	P	P	P	P	P	P	P	P	P
D_{11}	P	P	NP	P	NP	NP	P	NP	NP
D_{12}	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_{13}	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_{20}	P	P	P	P	P	P	P	P	P
D_{21}	P	P	P	P	P	P	P	P	P
D_{22}	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_{23}	NP	NP	NP	NP	NP	NP	NP	NP	NP
D_{30}	P	P	P	P	P	P	P	P	P
D_{31}	P	P	P	P	P	P	P	P	P
D_{32}	P	NP	NP	P	NP	NP	NP	NP	NP
D_{33}	NP	NP	NP	NP	NP	NP	NP	NP	NP

the company will be 8 million dollars; but the profit drops to 4 million dollars when a low volume of players is encountered. At the end of every year, the profit of the year is collected, and then the company has the option to take certain actions that will influence the performance of their service and hence the volume of players in the future. But some of these actions are costly so they reduce instant profit. To be more specific, the company can choose to: (1) take no action, which costs nothing; (2) perform only regular maintenance to the service system, which costs 1 million; (3) or fully upgrade the service system, which costs 3 million. If the volume of players in the last year was high, it stays in the high state in the coming year with probability 0.4 if no action is taken; this probability is 0.8 if only regular maintenance is performed; and the probability rises to 1 if the system is fully upgraded. If the volume of players in the last year was low, then the probability that the player volume stays low is 0.9 with no action taken, 0.6 with regular maintenance, and 0.2 when the service system is fully upgraded. Assuming the discount factor is 0.9, and that the company experienced a low volume of players last year, determine the optimal policy for the company.

2. Determine the optimal policy and the values for the Markov decision process of the example in the previous problem, assuming that the process has an infinite horizon and the discount factor remains equal to 0.9.

Chapter 6
Higher-Order Markov Chains

6.1 Introduction

Data sequences or time series occur frequently in many real world applications. One of the most important steps in analyzing a data sequence (or time series) is the selection of an appropriate mathematical model for the data. This is because it helps in predictions, hypothesis testing and rule discovery. A data sequence $\{X^{(n)}\}$ can be logically represented as a vector

$$(X^{(1)}, X^{(2)}, \cdots, X^{(T)}),$$

where T is the length of the sequence, and $X^{(i)} \in \mathrm{DOM}(A)$ $(1 \leq i \leq T)$ is associated with a defined semantic and data type. In our context, we consider and assume other types used can be mapped to one of these two types. The domains of attributes associated with these two types are called numeric and categorical respectively. A numeric domain consists of real numbers. A domain $DOM(A)$ is defined as categorical if it is finite and unordered, e.g., for any $a, b \in DOM(A)$, either $a = b$ or $a \neq b$, see for instance [109]. Numerical data sequences have been studied in detail, see for instance [25]. Mathematical tools such as the Fourier transform and spectral analysis are employed frequently in the analysis of numerical data sequences. Many different time sequence models have been proposed and developed in the literature, see for instance [25] and the references therein.

For categorical data sequence, there are many situations that one would like to employ a higher-order Markov chain model as a mathematical tool, see for instance [1, 146, 151, 155, 176]. A number of applications can be found in the literatures [120, 155, 177]. For example, in sales demand prediction, products are classified into several states: very high sales volume, high sales volume, standard, low sales volume and very low sales volume (categorical type: ordinal data). A higher-order

W.-K. Ching et al., *Markov Chains*, International Series in Operations Research & Management Science 189, DOI 10.1007/978-1-4614-6312-2_6,

Markov chain model has been used in fitting observed data and then applied to wind turbine design. In these applications and many others, one would like to

1. Characterize categorical data sequences for the purpose of comparison and classification; or
2. Model categorical data sequences and hence make predictions in the control and planning process.

It has been shown that higher-order Markov chain models can be a promising approach for these purposes [53, 57, 120, 176, 177].

The remainder of this chapter is organized as follows. In Sect. 6.2, we present the higher-order Markov chain model. Estimation methods for the model parameters are also discussed. In Sect. 6.3, the higher-order Markov chain model is applied to a number of applications such as sales demand predictions and web page predictions. Further extension of the model is then discussed in Sect. 6.4. In Sect. 6.5, we apply the model to the Newsboy problem, a classical problem in management sciences. Section 6.6 discusses a higher-order Markov-switching model for risk measurement. Finally a summary is given in Sect. 6.7.

6.2 Higher-Order Markov Chains

In the following, we assume that each data point $X^{(n)}$ in a categorical data sequence takes values in the set

$$M \equiv \{1, 2, \ldots, m\}$$

where m is finite, i.e., a sequence has m possible categories or states. The conventional model for a k-th order Markov chain has $(m-1)m^k$ model parameters. The major problem in using this kind of model is that the number of parameters (the transition probabilities) increases exponentially with respect to the order of the model. This large number of parameters discourages people from using a higher-order Markov chain directly.

In [176], Raftery proposed a higher-order Markov chain model which involves only one additional parameter for each extra lag. The model can be written as follows:

$$P(X^{(n)} = j_0 \mid X^{(n-1)} = j_1, \ldots, X^{(n-k)} = j_k) = \sum_{i=1}^{k} \lambda_i q_{j_0 j_i} \tag{6.1}$$

where

$$\sum_{i=1}^{k} \lambda_i = 1$$

and $Q = [q_{ij}]$ is a transition matrix with column sums equal to one, such that

$$0 \le \sum_{i=1}^{k} \lambda_i q_{j_0 j_i} \le 1, \quad j_0, j_i \in M. \tag{6.2}$$

The constraint in (6.2) is to guarantee that the right hand side of (6.1) is a probability distribution. The total number of independent parameters in this model is of size $(k + m^2)$. Raftery proved that (6.1) is analogous to the standard AR(n) model in the sense that each additional lag, after the first, is specified by a single parameter and that the autocorrelations satisfy a system of linear equations similar to the Yule-Walker equations. Moreover, the parameters $q_{j_0 j_i}$ and λ_i can be estimated numerically by maximizing the log-likelihood of (6.1) subject to the constraints (6.2). However, this approach involves solving a highly non-linear optimization problem. The proposed numerical method guarantees neither convergence nor a global maximum even if it converges.

6.2.1 The New Model

In this subsection, we extend Raftery's model [176] to a more general higher-order Markov chain model by allowing Q to vary with different lags. Here we assume that the weight λ_i is non-negative such that

$$\sum_{i=1}^{k} \lambda_i = 1. \tag{6.3}$$

It should be noted that (6.1) can be re-written as

$$\mathbf{X}^{(n+k+1)} = \sum_{i=1}^{k} \lambda_i Q \mathbf{X}^{(n+k+1-i)}, \tag{6.4}$$

where $\mathbf{X}^{(n+k+1-i)}$ is the probability distribution of the states at time $(n+k+1-i)$. Using (6.3) and the fact that Q is a transition probability matrix, we note that each entry of $\mathbf{X}^{(n+k+1)}$ is between 0 and 1, and that the sum of all entries is also equal to 1. In Raftery's model, it does not assume λ to be non-negative and therefore the additional constraints (6.2) should be added to guarantee that $\mathbf{X}^{(n+k+1)}$ is the probability distribution of the states.

Raftery's model in (6.4) can be generalized as follows:

$$\mathbf{X}^{(n+k+1)} = \sum_{i=1}^{k} \lambda_i Q_i \mathbf{X}^{(n+k+1-i)}. \tag{6.5}$$

The total number of independent parameters in the new model is $(k + km^2)$. We note that if

$$Q_1 = Q_2 = \ldots = Q_k$$

then (6.5) is just the Raftery's model in (6.4).

In the model we assume that $\mathbf{X}^{(n+k+1)}$ depends on $\mathbf{X}^{(n+i)}$ $(i = 1, 2, \ldots, k)$ via the matrix Q_i and weight λ_i. One may relate Q_i to the i-step transition matrix of the process and we will use this idea to estimate Q_i. Here we assume that each Q_i is a non-negative stochastic matrix with column sums equal to one. Before we present our estimation method for the model parameters we first discuss some properties of our proposed model in the following proposition.

Proposition 6.1. *Suppose Q_i is irreducible and $\lambda_i > 0$ such that*

$$0 \le \lambda_i \le 1 \quad \text{and} \quad \sum_{i=1}^{k} \lambda_i = 1,$$

then the model in (6.5) has a stationary distribution $\bar{\mathbf{X}}$ when $n \to \infty$ independent of the initial state vectors $\mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(k-1)}$. The stationary distribution $\bar{\mathbf{X}}$ is also the unique solution of the following linear system of equations:

$$(I - \sum_{i=1}^{n} \lambda_i Q_i)\bar{\mathbf{X}} = \mathbf{0} \quad \text{and} \quad \mathbf{1}^T \bar{\mathbf{X}} = 1.$$

Here I is the m-by-m identity matrix (m is the number of possible states taken by each data point) and $\mathbf{1}$ is an $m \times 1$ vector of ones.

Proof. Let

$$\mathbf{Y}^{(n+k+1)} = (\mathbf{X}^{(n+k+1)}, \mathbf{X}^{(n+k)}, \ldots, \mathbf{X}^{(n+2)})^T$$

be an $nm \times 1$ vector. Then one may write

$$\mathbf{Y}^{(n+1)} = R\mathbf{Y}^{(n)}$$

where

$$R = \begin{pmatrix} \lambda_1 Q_1 & \lambda_2 Q_2 & \cdots & \lambda_{n-1} Q_{n-1} & \lambda_n Q_n \\ I & 0 & \cdots & 0 & 0 \\ 0 & I & 0 & & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & I & 0 \end{pmatrix} \tag{6.6}$$

is an $nm \times nm$ square matrix. We then define

$$\tilde{R} = \begin{pmatrix} \lambda_1 Q_1 & I & 0 & 0 & \cdots & \cdots & 0 \\ \vdots & 0 & I & 0 & & & \vdots \\ \vdots & 0 & 0 & \ddots & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \lambda_{n-1} Q_{n-1} & \vdots & & & \ddots & \ddots & I \\ \lambda_n Q_n & 0 & \cdots & & \cdots & \cdots & 0 \end{pmatrix}. \tag{6.7}$$

We note that R and $\tilde{R}$ have the same characteristic polynomial in τ:

$$det[(-1)^{k-1}((\lambda_1 Q_1 - \tau I)\tau^{k-1} + \sum_{i=2}^{k} \lambda_i Q_i \tau^{k-i})].$$

Thus R and $\tilde{R}$ have the same set of eigenvalues.

It is clear that $\tilde{R}$ is an irreducible stochastic matrix with column sums equal to one. Then from the Perron-Frobenius Theorem [8, p. 134], all the eigenvalues of $\tilde{R}$ (or equivalently R) lie in the interval (0, 1] and there is exactly one eigenvalue equal to one. This implies that

$$\lim_{n\to\infty} \overbrace{R \ldots R}^{n\ terms} = \lim_{n\to\infty} (R)^n = \mathbf{V}\mathbf{U}^T$$

is a positive rank one matrix as R is irreducible. Therefore we have

$$\begin{aligned}\lim_{n\to\infty} \mathbf{Y}^{(n+k+1)} &= \lim_{n\to\infty} (R)^n \mathbf{Y}^{(k+1)} \\ &= \mathbf{V}(\mathbf{U}^t \mathbf{Y}^{(k+1)}) \\ &= \alpha \mathbf{V}.\end{aligned}$$

Here α is a positive number because $\mathbf{Y}_{k+1} \neq \mathbf{0}$ and is non-negative. This implies that $X^{(n)}$ also tends to a stationary distribution as t goes to infinity. Hence we have

$$\lim_{n\to\infty} \mathbf{X}^{(n+k+1)} = \lim_{n\to\infty} \sum_{i=1}^{k} \lambda_i Q_i \mathbf{X}^{(n+k+1-i)}$$

and therefore we have

$$\bar{\mathbf{X}} = \sum_{i=1}^{k} \lambda_i Q_i \bar{\mathbf{X}}.$$

The stationary distribution vector $\bar{\mathbf{X}}$ satisfies

$$(I - \sum_{i=1}^{k} \lambda_i Q_i)\bar{\mathbf{X}} = \mathbf{0} \quad \text{with} \quad \mathbf{1}^T \bar{\mathbf{X}} = 1. \tag{6.8}$$

The normalization constraint is necessary as the matrix

$$(I - \sum_{i=1}^{k} \lambda_i Q_i)$$

has a one-dimensional null space. The result is then proved.

We remark that if some λ_i are equal to zero, one can rewrite the vector $\mathbf{Y}_{n+k+1}$ in terms of $\mathbf{X}_i$ where the λ_i are nonzero. Then the model in (7.13) still has a stationary distribution $\bar{\mathbf{X}}$ when n goes to infinity independent of the initial state vectors. Moreover, the stationary distribution $\bar{\mathbf{X}}$ can be obtained by solving the corresponding linear system of equations with the normalization constraint.

It has been shown in [216] that the condition can be further relaxed and the result is presented in the following proposition.

Proposition 6.2. *If Q_k is irreducible and aperiodic, $\lambda_1, \lambda_k > 0$ and $\sum_{i=1}^{k} \lambda_i = 1$, then the model has a stationary distribution $\mathbf{X}$ satisfying (6.8) and*

$$\lim_{n \to \infty} \mathbf{X}^{(n)} = \mathbf{X}.$$

6.2.2 Parameter Estimation

In this subsection, we present efficient methods to estimate the parameters Q_i and λ_i for $i = 1, 2, \ldots, k$. To estimate Q_i, one may regard Q_i as the i-step transition matrix of the categorical data sequence $\{X^{(n)}\}$. Given the categorical data sequence $\{X^{(n)}\}$, one can count the transition frequency $f_{jl}^{(i)}$ in the sequence from State l to State j in the i-step. Hence one can construct the i-step transition matrix for the sequence $\{X^{(n)}\}$ as follows:

$$F^{(i)} = \begin{pmatrix} f_{11}^{(i)} & \cdots\cdots & f_{m1}^{(i)} \\ f_{12}^{(i)} & \cdots\cdots & f_{m2}^{(i)} \\ \vdots & \vdots \;\; \vdots & \vdots \\ f_{1m}^{(i)} & \cdots\cdots & f_{mm}^{(i)} \end{pmatrix}. \tag{6.9}$$

From $F^{(i)}$, we get the estimates for $Q_i = [q_{lj}^{(i)}]$ as follows:

$$\hat{Q}_i = \begin{pmatrix} \hat{q}_{11}^{(i)} & \cdots\cdots & \hat{q}_{m1}^{(i)} \\ \hat{q}_{12}^{(i)} & \cdots\cdots & \hat{q}_{m2}^{(i)} \\ \vdots & \vdots \;\; \vdots & \vdots \\ \hat{q}_{1m}^{(i)} & \cdots\cdots & \hat{q}_{mm}^{(i)} \end{pmatrix}, \tag{6.10}$$

where

$$\hat{q}_{lj}^{(i)} = \begin{cases} \dfrac{f_{lj}^{(i)}}{\displaystyle\sum_{l=1}^{m} f_{lj}^{(i)}} & \text{if } \displaystyle\sum_{l=1}^{m} f_{lj}^{(i)} \neq 0 \\ 0 & \text{otherwise.} \end{cases} \tag{6.11}$$

We note that the computational complexity of the construction of $F^{(i)}$ is of $O(L^2)$ operations, where L is the length of the given data sequence. Hence the total computational complexity of the construction of $\{F^{(i)}\}_{i=1}^k$ is of $O(kL^2)$ operations. Here k is the number of lags.

The following proposition shows that these estimators are unbiased.

Proposition 6.3. *The estimators in (6.11) satisfy*

$$E(f_{lj}^{(i)}) = q_{lj}^{(i)} E\left(\sum_{j=1}^{m} f_{lj}^{(i)}\right).$$

Proof. Let T be the length of the sequence, $[q_{lj}^{(i)}]$ be the i-step transition probability matrix and $\bar{X}_l$ be the steady-state probability that the process is in state l. Then we have

$$E(f_{lj}^{(i)}) = T \cdot \bar{X}_l \cdot q_{lj}^{(i)}$$

and

$$E(\sum_{j=1}^{m} f_{lj}^{(i)}) = T \cdot \bar{X}_l \cdot (\sum_{j=1}^{m} q_{lj}^{(i)}) = T \cdot \bar{X}_l.$$

Therefore we have

$$E(f_{lj}^{(i)}) = q_{lj}^{(i)} \cdot E(\sum_{j=1}^{m} f_{lj}^{(i)}).$$

In some situations, if the sequence is too short then $\hat{Q}_i$ (especially $\hat{Q}_k$) contains a lot of zeros (therefore $\hat{Q}_n$ may not be irreducible). However, this did not occur in the tested examples. Here we propose the second method for the parameter estimation. Let $\mathbf{W}^{(i)}$ be the probability distribution of the i-step transition sequence, then another possible estimation for Q_i can be $\mathbf{W}^{(i)}\mathbf{1}^t$. We note that if $\mathbf{W}^{(i)}$ is a positive vector, then $\mathbf{W}^{(i)}\mathbf{1}^t$ will be a positive matrix and hence an irreducible matrix.

Proposition 6.1 gives a sufficient condition for the sequence $\mathbf{X}^{(n)}$ to converge to a stationary distribution $\mathbf{X}$. Suppose $\mathbf{X}^{(n)} \to \bar{\mathbf{X}}$ as n goes to infinity, then $\bar{\mathbf{X}}$ can be estimated from the sequence $\{X^{(n)}\}$ by computing the proportion of the occurrence of each state in the sequence, denoted by $\hat{\mathbf{X}}$. From (6.8) one would expect that

$$\sum_{i=1}^{k} \lambda_i \hat{Q}_i \hat{\mathbf{X}} \approx \hat{\mathbf{X}}. \tag{6.12}$$

This suggests one possible way to estimate the parameters

$$\lambda = (\lambda_1, \ldots, \lambda_k)$$

as follows. One may consider the following minimization problem:

$$\min_{\lambda} \left\{ \left\| \sum_{i=1}^{k} \lambda_i \hat{Q}_i \hat{\mathbf{X}} - \hat{\mathbf{X}} \right\| \right\}$$

subject to

$$\sum_{i=1}^{k} \lambda_i = 1, \quad \text{and} \quad \lambda_i \geq 0, \quad \forall i.$$

Here $||.||$ is a certain vector norm. In particular, if $||.||_\infty$ is chosen, we have the following minimization problem:

$$\min_{\lambda} \left\{ \max_{l} \left| \left[\sum_{i=1}^{k} \lambda_i \hat{Q}_i \hat{\mathbf{X}} - \hat{\mathbf{X}} \right]_l \right| \right\}$$

subject to

$$\sum_{i=1}^{k} \lambda_i = 1, \quad \text{and} \quad \lambda_i \geq 0, \quad \forall i.$$

Here $[\cdot]_l$ denotes the lth entry of the vector. The constraints in the optimization problem guarantee the existence of the stationary distribution $\mathbf{X}$. Next we see that the above minimization problem can be formulated as a linear programming problem:

$$\min_{\lambda} \; w$$

subject to

$$\begin{pmatrix} w \\ w \\ \vdots \\ w \end{pmatrix} \geq \hat{\mathbf{X}} - \left[\hat{Q}_1 \hat{\mathbf{X}} \mid \hat{Q}_2 \hat{\mathbf{X}} \mid \cdots \mid \hat{Q}_n \hat{\mathbf{X}} \right] \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{pmatrix},$$

$$\begin{pmatrix} w \\ w \\ \vdots \\ w \end{pmatrix} \geq -\hat{\mathbf{X}} + \left[\hat{Q}_1 \hat{\mathbf{X}} \mid \hat{Q}_2 \hat{\mathbf{X}} \mid \cdots \mid \hat{Q}_n \hat{\mathbf{X}} \right] \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{pmatrix},$$

$$w \geq 0, \quad \sum_{i=1}^{k} \lambda_i = 1, \quad \text{and} \quad \lambda_i \geq 0, \quad \forall i.$$

We can solve the above linear programming problem efficiently and obtain the parameters λ_i. In the next subsection, we will demonstrate the estimation method by a simple example.

Instead of solving a min-max problem, one can also choose to use $||.||_1$ and formulate the following minimization problem:

$$\min_{\lambda} \left\{ \sum_{l=1}^{m} \left| \left[\sum_{i=1}^{k} \lambda_i \hat{Q}_i \hat{\mathbf{X}} - \hat{\mathbf{X}} \right]_l \right| \right\}$$

subject to

$$\sum_{i=1}^{k} \lambda_i = 1, \quad \text{and} \quad \lambda_i \geq 0, \quad \forall i.$$

The corresponding linear programming problem is given as follows:

$$\min_{\lambda} \sum_{l=1}^{m} w_l$$

subject to

$$\begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{pmatrix} \geq \hat{\mathbf{X}} - \left[\hat{Q}_1 \hat{\mathbf{X}} \mid \hat{Q}_2 \hat{\mathbf{X}} \mid \cdots \mid \hat{Q}_k \hat{\mathbf{X}} \right] \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_k \end{pmatrix},$$

$$\begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{pmatrix} \geq -\hat{\mathbf{X}} + \left[\hat{Q}_1 \hat{\mathbf{X}} \mid \hat{Q}_2 \hat{\mathbf{X}} \mid \cdots \mid \hat{Q}_k \hat{\mathbf{X}} \right] \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_k \end{pmatrix},$$

$$w_i \geq 0, \quad \forall i, \quad \sum_{i=1}^{k} \lambda_i = 1, \quad \text{and} \quad \lambda_i \geq 0, \quad \forall i.$$

In the above linear programming formulation, the number of variables is equal to k and the number of constraints is equal to $(2m + 1)$. The complexity of solving the linear programming problem is $O(k^3 L)$, where k is the number of variables and L is the number of binary bits needed to store all the data (the constraints and the objective function) of the problem [98].

We remark that other norms such as $||.||_2$ can also be considered. In this case, it will result in a quadratic programming problem. It is known that in approximating data by a linear function [76, p. 220], $||.||_1$ gives the most robust answer, $||.||_\infty$ avoids gross discrepancies with the data as much as possible and if the errors are known to be normally distributed then $||.||_2$ is the best choice. In the tested examples, we only consider the norms leading to solutions of the linear programming problems.

6.2.3 An Example

We consider a sequence $\{X^{(n)}\}$ of the three states ($m = 3$) given by

$$\{1, 1, 2, 2, 1, 3, 2, 1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 1, 2\}. \tag{6.13}$$

The sequence $\{X^{(n)}\}$ can be written in vector form

$$X^{(1)} = (1, 0, 0)^T,\ X^{(2)} = (1, 0, 0)^T,\ X^{(3)} = (0, 1, 0)^T,\ \ldots,\ X^{(20)} = (0, 1, 0)^T.$$

We consider $k = 2$, then from (6.13) we have the transition frequency matrices

$$F^{(1)} = \begin{pmatrix} 1 & 3 & 3 \\ 6 & 1 & 1 \\ 1 & 3 & 0 \end{pmatrix} \quad \text{and} \quad F^{(2)} = \begin{pmatrix} 1 & 4 & 1 \\ 3 & 2 & 3 \\ 3 & 1 & 0 \end{pmatrix}. \tag{6.14}$$

Therefore from (6.14) we have the i-step transition probability matrices ($i = 1, 2$) as follows:

$$\hat{Q}_1 = \begin{pmatrix} 1/8 & 3/7 & 3/4 \\ 3/4 & 1/7 & 1/4 \\ 1/8 & 3/7 & 0 \end{pmatrix} \quad \text{and} \quad \hat{Q}_2 = \begin{pmatrix} 1/7 & 4/7 & 1/4 \\ 3/7 & 2/7 & 3/4 \\ 3/7 & 1/7 & 0 \end{pmatrix} \tag{6.15}$$

and

$$\hat{\mathbf{X}} = (\frac{2}{5}, \frac{2}{5}, \frac{1}{5})^T.$$

Hence we have

$$\hat{Q}_1\hat{\mathbf{X}} = (\frac{13}{35}, \frac{57}{140}, \frac{31}{140})^T,$$

and

$$\hat{Q}_2\hat{\mathbf{X}} = (\frac{47}{140}, \frac{61}{140}, \frac{8}{35})^T.$$

To estimate λ_i one can consider the optimization problem:

$$\min_{\lambda_1, \lambda_2} w$$

subject to

$$\begin{cases} w \geq \frac{2}{5} - \frac{13}{35}\lambda_1 - \frac{47}{140}\lambda_2 \\ w \geq -\frac{2}{5} + \frac{13}{35}\lambda_1 + \frac{47}{140}\lambda_2 \\ w \geq \frac{2}{5} - \frac{57}{140}\lambda_1 - \frac{61}{140}\lambda_2 \\ w \geq -\frac{2}{5} + \frac{57}{140}\lambda_1 + \frac{61}{140}\lambda_2 \\ w \geq \frac{1}{5} - \frac{31}{140}\lambda_1 - \frac{8}{35}\lambda_2 \\ w \geq -\frac{1}{5} + \frac{31}{140}\lambda_1 + \frac{8}{35}\lambda_2 \\ w \geq 0, \\ \lambda_1 + \lambda_2 = 1, \\ \lambda_1, \lambda_2 \geq 0. \end{cases}$$

The optimal solution is

$$(\lambda_1^*, \lambda_2^*, w^*) = (1, 0, 0.0286),$$

and we have the model

$$\mathbf{X}^{(n+1)} = \hat{Q}_1 \mathbf{X}^{(n)}. \tag{6.16}$$

We remark that if we do not specify the non-negativity of λ_1 and λ_2, the optimal solution becomes

$$(\lambda_1^{**}, \lambda_2^{**}, w^{**}) = (1.80, -0.80, 0.0157),$$

and the corresponding model is

$$\mathbf{X}^{(n+1)} = 1.80\hat{Q}_1 \mathbf{X}^{(n)} - 0.80\hat{Q}_2 \mathbf{X}^{(n-1)}. \tag{6.17}$$

Although w^{**} is less than w^*, the model (6.17) is not suitable. It is easy to check that

$$1.80\hat{Q}_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} - 0.80\hat{Q}_2 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} -0.2321 \\ 1.1214 \\ 0.1107 \end{pmatrix},$$

and therefore λ_1^{**} and λ_2^{**} are not valid parameters.

We note that if we consider the minimization problem:

$$\min_{\lambda_1, \lambda_2} \; (w_1 + w_2 + w_3)$$

subject to

$$\begin{cases} w_1 \geq \frac{2}{5} - \frac{13}{35}\lambda_1 - \frac{47}{140}\lambda_2 \\ w_1 \geq -\frac{2}{5} + \frac{13}{35}\lambda_1 + \frac{47}{140}\lambda_2 \\ w_2 \geq \frac{2}{5} - \frac{57}{140}\lambda_1 - \frac{61}{140}\lambda_2 \\ w_2 \geq -\frac{2}{5} + \frac{57}{140}\lambda_1 + \frac{61}{140}\lambda_2 \\ w_3 \geq \frac{1}{5} - \frac{31}{140}\lambda_1 - \frac{9}{35}\lambda_2 \\ w_3 \geq -\frac{1}{5} + \frac{31}{140}\lambda_1 + \frac{9}{35}\lambda_2 \\ w_1, w_2, w_3 \geq 0, \quad \lambda_1 + \lambda_2 = 1, \quad \lambda_1, \lambda_2 \geq 0, \end{cases}$$

then the optimal solution is the same as the previous min-max formulation, and is equal to

$$(\lambda_1^*, \lambda_2^*, w_1^*, w_2^*, w_3^*) = (1, 0, 0.0286, 0.0071, 0.0214).$$

6.3 Some Applications

In this section we apply our model to some data sequences. They include sales demand data and webpage data. Given the state vectors $\mathbf{X}^{(i)}$, $i = n-k, n-k+1, \ldots, k-1$, the state probability distribution at time n can be estimated as follows:

$$\hat{\mathbf{X}}^{(n)} = \sum_{i=1}^{k} \lambda_i \hat{Q}_i \mathbf{X}^{(n-i)}.$$

In many applications, one would like to make use of the higher-order Markov chain models for the purpose of prediction. According to this state probability distribution, the prediction of the next state $\hat{X}^{(n)}$ at time n can be taken as the state with the maximum probability, i.e.,

$$\hat{X}^{(n)} = j, \quad \text{if } [\hat{\mathbf{X}}^{(n)}]_i \leq [\hat{\mathbf{X}}^{(n)}]_j, \quad \forall 1 \leq i \leq m.$$

To evaluate the performance and effectiveness of the higher-order Markov chain model, a prediction accuracy r is defined as

$$r = \frac{1}{T} \sum_{t=k+1}^{T} \delta_t,$$

where T is the length of the data sequence and

$$\delta_t = \begin{cases} 1, & \text{if } \hat{X}^{(t)} = X^{(t)} \\ 0, & \text{otherwise.} \end{cases}$$

Using the example in the previous section, two possible prediction rules can be drawn as follows:

$$\begin{cases} \hat{X}^{(n+1)} = 2, & \text{if } X^{(n)} = 1, \\ \hat{X}^{(n+1)} = 1, & \text{if } X^{(n)} = 2, \\ \hat{X}^{(n+1)} = 1, & \text{if } X^{(n)} = 3 \end{cases}$$

or

$$\begin{cases} \hat{X}^{(n+1)} = 2, & \text{if } X^{(n)} = 1, \\ \hat{X}^{(n+1)} = 3, & \text{if } X^{(n)} = 2, \\ \hat{X}^{(n+1)} = 1, & \text{if } X^{(n)} = 3. \end{cases}$$

The prediction accuracy r for the sequence in (6.13) is equal to $12/19$ for both prediction rules. While the prediction accuracies of other rules for the sequence in (6.13) are less than the value $12/19$.

Next other numerical results on different data sequences are discussed. In the following tests, we solve min-max optimization problems to determine the parameters λ_i of higher-order Markov chain models. However, we remark that the results of using the $||.||_1$ optimization problem as discussed in the previous section are about the same as that of using the min-max formulation.

6.3.1 The Sales Demand Data

A large soft-drink company in Hong Kong faces an in-house problem of production planning and inventory control. A pressing issue that stands out is the storage space of its central warehouse, which often finds itself in a state of overflow or near capacity. The company thus faces urgent need to study the interplay between their storage space requirement and its growing sales demand. There are product states defined by the level of sales volume. The states include:

state 1: very slow-moving (very low sales volume)
state 2: slow-moving
state 3: standard
state 4: fast-moving
state 5: very fast-moving (very high sales volume)

Such labelling is useful from both marketing and production planning points of view. For instance, when it comes to production planning, the company can develop a Dynamic Programming (DP) model to recommend better production strategies to minimize its inventory build-up, and to maximize the demand satisfaction. Since the number of alternatives at each stage (each day in the planning horizon) is very large

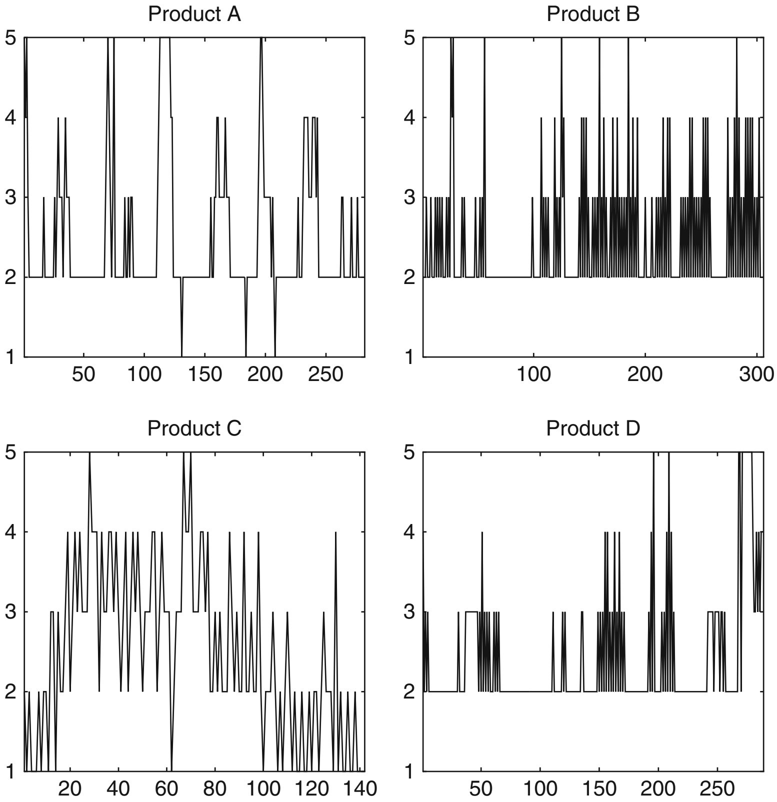


Fig. 6.1 The states of four products A, B, C and D

(the number of products raised to the power of the number of production lines), the computational complexity of the DP model is enormous. A priority scheme based on the state (the level of sales volume) of the product is introduced to tackle this combinatorial problem, and therefore an effective and efficient production plan can be obtained. It is obvious that the accurate prediction of the state (the level of sales volume) of the product is important in the production planning model.

In Fig. 6.1 (Taken from [52]), we show the states of four of the products of the soft-drink company for some sales periods. Here we employ higher-order Markov chain models to predict categories of these four products separately. For the new model, we consider a second-order ($n = 2$) model and use the data to estimate $\hat{Q}_i$ and λ_i ($i = 1, 2$). The results are reported in Table 6.1. For comparison, we also study the first-order and the second-order full Markov chain models. Results show the effectiveness of our new model. We also see from Fig. 6.1 that the change of the states of the Products A, B and D is more regular than that of the Product C. We find in Table 6.1 that the prediction results for the Products A, B and D are better than that of C (Taken from [52]).

Table 6.1 Prediction accuracy in the sales demand data

	Product A	Product B	Product C	Product D
First-order Markov chain model	0.76	0.70	0.39	0.74
Second-order Markov chain model	0.79	0.78	0.51	0.83
New model ($n = 2$)	0.78	0.76	0.43	0.78
Randomly chosen	0.20	0.20	0.20	0.20

6.3.2 Webpage Prediction

The Internet provides a rich source for users to retrieve useful information. However, it is easy for a user to get lost in an ocean of information. One way to assist the user with their informational needs is to predict a user's future request and to use the prediction for making recommendations. Recommendation systems rely on a prediction model to make inferences about users' interests, upon which it will make recommendations. Examples are the WebWatcher [126] system and Letzia [147] system. An accurate prediction can potentially shorten the users' access times and reduce network traffic when the recommendation is handled correctly. In this subsection, we use a higher-order Markov chain model to exploit the information from web server logs for predicting users' actions on the web.

The higher-order Markov chain model is built on a web server log file. We consider the web server log file to be preprocessed into a collection of user sessions. Each session is indexed by a unique user ID and starting time [183]. Each session is a sequence of requests where each request corresponds to a visit to a web page. We represent each request as a state. Then each session is just a categorical data sequence. Moreover, we denote each Web page (state) by an integer.

6.3.2.1 Web Log Files and Preprocessing

Experiments were conducted on a real Web log file taken from the Internet. We first implemented a data preprocessing program to extract sessions from the log file. We downloaded two web log files from the Internet. The data set was a web log file from the EPA WWW server located at Research Triangle Park, NC. This log contained 47748 transactions generated in 24 h from 23:53:25 EDT, August 29, to 23:53:07, August 30, 1995. In preprocessing, we removed all the invalid requests and the requests for images. We used Host ID to identify visitors and a 30 min time threshold to identify sessions. 428 sessions of lengths between 16 and 20 min were identified from the EPA log file. The total number of web pages (states) involved is 3753.

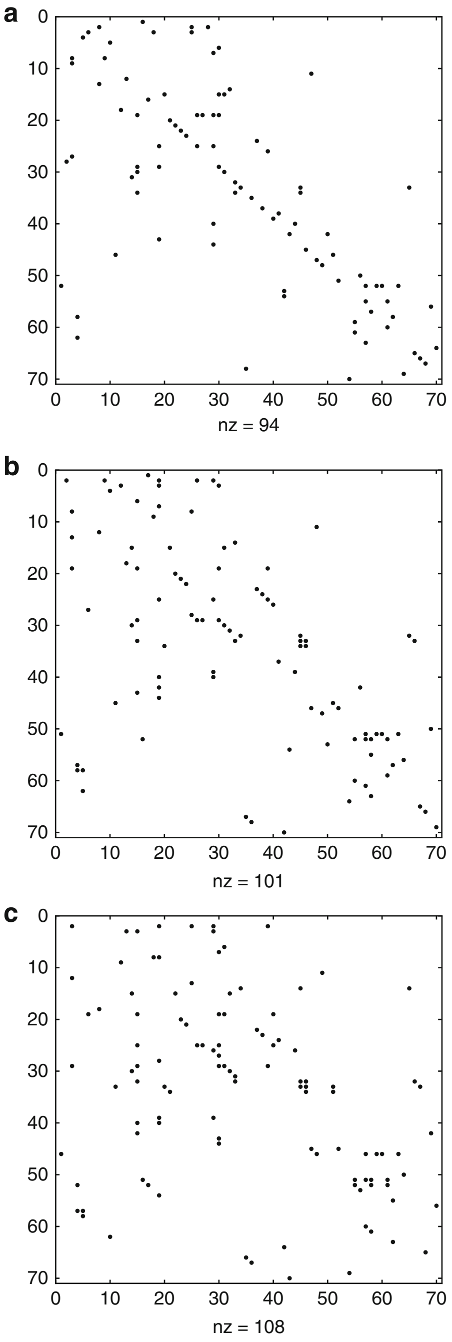
6.3.2.2 Prediction Models

By exploring the session data from the web log file, we observed that a large number of similar sessions rarely exist. This is because in a complex web site with a variety of pages, and many paths and links, one should not expect that in a given time period, a large number of visitors follow only a few paths. If this is true, it would mean that the structure and contents of the web site had a serious problem, because only a few pages and paths were interesting to the visitors. In fact, most web site designers expect that the majority of their pages, if not every one, are visited and paths are followed (equally) frequently. The first and the second step transition matrices of all sessions are very sparse in our case. In fact, there are $3,900$ and $4,747$ entries in the first and the second step transition matrices respectively. Nonzero entries only make up about 0.033 % of the total elements of the first and the second step transition matrices.

Based on these observations, if we directly use these transition matrices to build prediction models, they may not be effective. Since the number of pages (states) are very large, the prediction probability for each page may be very low. Moreover, the computational work for solving the linear programming problem in the estimation of the λ_i are also high, since the number of constraints in the linear programming problem depend on the number of pages (states). Here we propose to use clustering algorithms [120] to cluster the sessions. The idea is to form a transition probability matrix for each session, to construct the distance between two sessions based on the Frobenius norm (See Definition 1.42 of Chap. 1) of the difference of their transition probability matrices, and then use a k-means algorithm to cluster the sessions. As a result of the cluster analysis, the web page cluster can be used to construct a higher-order Markov chain model. Then we prefetch those web documents that are close to a user-requested document in a Markov chain model.

We find that there is a clear similarity among these sessions in each cluster for the EPA log file. As an example, we show in Fig. 6.2 (Taken from [52]) that the first, the second and the third step transition probability matrices of a cluster in EPA log file. There are 70 pages involved in this cluster. Non-zero entries make up about 1.92 %, 2.06 % and 2.20 % respectively of the total elements of the first, the second and the third step transition matrices. Usually, the prediction of the next web page is based on the current page and the previous few pages [2]. Therefore, we use a third-order model ($n = 3$) and consider the first, the second and the third transition matrices in the construction of the Markov chain model. After we find the transition matrices, we determine λ_i and build our new higher-order Markov chain model for each cluster. For the above mentioned cluster, the corresponding λ_1, λ_2 and λ_3 are 0.4984, 0.4531 and 0.0485 respectively. The parameters show that the prediction of the next web page strongly depends on the current and the previous pages.

Fig. 6.2 The first (**a**), second (**b**), third (**c**) step transition matrices



6.3.2.3 Prediction Results

We performed clustering based on the transition matrices and parameters. Sixteen clusters were found experimentally based on the average within-cluster distance. Therefore sixteen third-order Markov chain models for these clusters are determined for the prediction of user-request documents. For comparison, we also compute the first-order Markov chain model for each cluster. In total there are $6,255$ web documents for the prediction test. We find the prediction accuracy of our method is about 77 %, but the prediction accuracy of using the first-order full Markov chain model is 75 %. Results show an improvement in the prediction. We have applied these prediction results to the problem of integrated web caching and prefetching [210]. The slight increase of the prediction accuracy can enhance a prefetching engine. Experimental results in [210] show that the resultant system outperforms web systems that are based on caching alone.

6.4 Extension of the Model

In this section, we consider an extension of the higher-order Markov chain model [68]. The higher-order Markov chain model (6.5)

$$\mathbf{X}_{n+k+1} = \sum_{i=1}^{k} \lambda_i Q_i \mathbf{X}_{n+k+1-i}$$

can be further generalized by replacing the constraints

$$0 \le \lambda_i \le 1, \quad i = 1, 2, \ldots, k \quad \text{and} \quad \sum_{i=1}^{k} \lambda_i = 1$$

by

$$0 \le \sum_{i=1}^{k} \lambda_i q_{j_0 j_i}^{(i)} \le 1, \quad j_0, j_i \in \mathcal{M} \quad \text{and} \quad \sum_{i=1}^{k} \lambda_i = 1.$$

We expect this new model will have better prediction accuracy when an appropriate order of model is used.

Next we give the sufficient condition for the proposed model to be stationary. Similar to the proof in [176], it can be shown that:

Proposition 6.4. *If we suppose that $\{X^{(n)}, n \in N\}$ is defined by (7.13), where the constraints $0 \le \lambda \le 1$ are replaced by*

$$0 < \sum_{i=1}^{k} \lambda_i q_{j_0 j_i}^{(i)} \le 1,$$

then the model (7.13) has a stationary distribution $\bar{\mathbf{X}}$ *when* $n \to \infty$ *independent of the initial state vectors*

$$(\mathbf{X}^{(0)}, \mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(k-1)}).$$

The stationary distribution $\bar{\mathbf{X}}$ *is also the unique solution of the linear system of equations:*

$$(I - \sum_{i=1}^{k} \lambda_i Q_i)\bar{\mathbf{X}} = 0 \quad \text{and} \quad \mathbf{1}^T \bar{\mathbf{X}} = 1.$$

We can use the method in Sect. 6.2.2 to estimate the parameters Q_i. For λ_i, the linear programming formulation can be considered as follows. In view of Proposition 6.4, if we suppose the model is stationary, then we have a stationary distribution $\bar{\mathbf{X}}$. Then $\bar{\mathbf{X}}$ can be estimated from the observed sequence $\{X^{(s)}\}$ by computing the proportion of the occurrence of each state in the sequence. In Sect. 6.2.2, it suggests one possible way to estimate the parameters

$$\lambda = (\lambda_1, \ldots, \lambda_k)$$

as follows. In view of (6.12) one can consider the following optimization problem:

$$\min_{\lambda} \left\{ \left|\left| \sum_{i=1}^{k} \lambda_i \hat{Q}_i \hat{\mathbf{X}} - \hat{\mathbf{X}} \right|\right|_{\infty} \right\} = \min_{\lambda} \left\{ \max_{j} \left[\left| \sum_{i=1}^{k} \lambda_i \hat{Q}_i \hat{\mathbf{X}} - \hat{\mathbf{X}} \right| \right]_j \right\}$$

subject to

$$\sum_{i=1}^{k} \lambda_i = 1,$$

and

$$0 \le \sum_{i=1}^{k} \lambda_i q_{j_0 j_i}^{(i)} \le 1, \quad j_0, j_i \in \mathcal{M}.$$

Here $[\cdot]_j$ denotes the jth entry of the vector. We see that the above optimization problem can be re-formulated as a linear programming problem as stated in the previous section. Instead of solving a min-max problem, one can also formulate the l_1-norm optimization problem. In these linear programming problems, we note that the number of variables is equal to k and the number of constraints is equal to $(2m^{k+1} + 2m + 1)$. With the following proposition (see also [177]), we can reduce the number of constraints to $(4m + 1)$ if we formulate the estimation problem as a nonlinear programming problem.

Proposition 6.5. *The constraints*

$$0 \le \sum_{i=1}^{k} \lambda_i q_{j_0 j_i}^{(i)} \le 1, \quad j_0, j_i \in \mathcal{M}$$

are equivalent to

$$\sum_{i=1}^{k} \left(\max\{\lambda_i, 0\} \min_{j_i}\{q_{j_0 j_i}^{(i)}\} - \max\{-\lambda_i, 0\} \max_{j_i}\{q_{j_0 j_i}^{(i)}\} \right) \ge 0 \tag{6.18}$$

and

$$\sum_{i=1}^{k} \left(\max\{\lambda_i, 0\} \max_{j_i}\{q_{j_0 j_i}^{(i)}\} - \max\{-\lambda_i, 0\} \min_{j_i}\{q_{j_0 j_i}^{(i)}\} \right) \le 1 \tag{6.19}$$

Proof. We prove the first part of the inequality. If inequality (6.18) holds, then

$$\begin{aligned}\sum_{i=1}^{k} \lambda_i q_{j_0 j_i}^{(i)} &= \sum_{\lambda_i \ge 0} \lambda_i q_{j_0 j_i}^{(i)} + \sum_{\lambda_i < 0} \lambda_i q_{j_0 j_i}^{(i)} \\ &\ge \sum_{\lambda_i \ge 0} \lambda_i \min_{j_i}\{q_{j_0 j_i}^{(i)}\} + \sum_{\lambda_i < 0} \lambda_i \max_{j_i}\{q_{j_0 j_i}^{(i)}\} \\ &\ge 0.\end{aligned}$$

Conversely, we assume that

$$\forall j_0, j_i \in \mathcal{M}, \quad \sum_{i=1}^{k} \lambda_i q_{j_0 j_i}^{(i)} \ge 0.$$

Suppose

$$\min_{j_i}\{q_{j_0 j_i}^{(i)}\} = q_{j_0 j_{i_0}}^{(i)}$$

and

$$\max_{j_i}\{q_{j_0 j_i}^{(i)}\} = q_{j_0 j_{i_1}}^{(i)}$$

then

$$\sum_{\lambda_i \ge 0} \lambda_i \min_{j_i}\{q_{j_0 j_i}^{(i)}\} + \sum_{\lambda_i < 0} \lambda_i \max_{j_i}\{q_{j_0 j_i}^{(i)}\} = \sum_{\lambda_i \ge 0} \lambda_i q_{j_0 j_{i_0}}^{(i)} + \sum_{\lambda_i < 0} \lambda_i q_{j_0 j_{i_1}}^{(i)} \ge 0.$$

This is equivalent to (6.18). One can use a similar method to prove the second part and hence we complete the proof.

In the following, we give a simple example to demonstrate our estimation methods. We consider a sequence $\{X^{(t)}\}$ of two states ($m = 2$) given by

$$\{1, 1, 2, 2, 1, 2, 2, 1, 2, 2, 1, 2, 2, 1, 2, 2, 1, 2, 2, 2\}. \tag{6.20}$$

The sequence $\{X^{(t)}\}$ can be written in vector form

$$X^{(1)} = (1, 0)^T, \quad X^{(2)} = (1, 0)^T, \quad X^{(3)} = (0, 1)^T, \quad \ldots \quad , X^{(20)} = (0, 1)^T.$$

If we consider $k = 2, 3, 4$, then from (6.20) we have the transition frequency matrices

$$F^{(1)} = \begin{pmatrix} 1 & 5 \\ 6 & 7 \end{pmatrix}, \quad F^{(2)} = \begin{pmatrix} 0 & 5 \\ 7 & 6 \end{pmatrix}, \tag{6.21}$$

$$F^{(3)} = \begin{pmatrix} 5 & 0 \\ 2 & 10 \end{pmatrix}, \quad F^{(4)} = \begin{pmatrix} 1 & 4 \\ 5 & 6 \end{pmatrix}. \tag{6.22}$$

Therefore from (6.21) we have the i-step transition matrices ($i = 1, 2, 3, 4$) as follows:

$$\hat{Q}_1 = \begin{pmatrix} 1/7 & 5/12 \\ 6/7 & 7/12 \end{pmatrix}, \hat{Q}_2 = \begin{pmatrix} 0 & 5/11 \\ 1 & 6/11 \end{pmatrix}, \tag{6.23}$$

$$\hat{Q}_3 = \begin{pmatrix} 5/7 & 0 \\ 2/7 & 1 \end{pmatrix}, \hat{Q}_4 = \begin{pmatrix} 1/6 & 4/10 \\ 5/6 & 6/10 \end{pmatrix} \tag{6.24}$$

and $\hat{\mathbf{X}} = (0.35, 0.65)^T$. In this example, the model parameters can be obtained by solving a linear programming problem. It turns out that the parameters obtained are exactly the same for both $\|\cdot\|_1$ and $\|\cdot\|_\infty$. We report the parameters for the case of $k = 2, 3, 4$. For $k = 2$, we have

$$(\lambda_1^*, \lambda_2^*) = (1.4583, -0.4583).$$

For $k = 3$, we have

$$(\lambda_1^*, \lambda_2^*, \lambda_3^*) = (1.25, 0, -0.25).$$

For $k = 4$, we have

$$(\lambda_1^*, \lambda_2^*, \lambda_3^*, \lambda_4^*) = (0, 0, -0.3043, 1.3043).$$

We remark that to compare different models, one can also adopt the χ^2 statistics method. From the observed data sequence, one can obtain the distribution of states

$$(O_1, O_2, \ldots, O_m).$$

From the model parameters Q_i and λ_i, by solving

$$\mathbf{X} = \sum_{i=1}^{n} \lambda_i \hat{Q}_i \mathbf{X} \quad \text{with} \quad \mathbf{1}^T \mathbf{X} = 1,$$

one can obtain the theoretical probability distribution of the states

$$(E_1, E_2, \ldots, E_m).$$

Then the χ^2 statistic is defined as

$$\chi^2 = L \sum_{i=1}^{m} \frac{(E_i - O_i)^2}{E_i}.$$

The smaller this value is, the better the model will be.

6.5 The Newsboy Problem

The Newsboy problem is a well-known classical problem in management science [165] and it can be described as follows. A newsboy starts selling newspaper every morning. The cost of each newspapers remaining unsold at the end of the day is C_o (overage cost) and the cost of each unit of unsatisfied demand is C_s (shortage cost). Suppose that the probability distribution function of the demand D is given by

$$\text{Prob}\,(D = d) = p_d \geq 0, \quad d = 1, 2, \ldots, m. \tag{6.25}$$

The objective here is to determine the best amount r^* of newspaper to be ordered such that the expected cost is minimized. To write down the expected long-run cost for a given amount of order size r we have the following two cases:

1. If the demand $d < r$, then the cost will be $(r - d)C_o$, and
2. if the demand $d > r$, then the cost will be $(d - r)C_s$.

Therefore the expected cost when the order size is r is given by

$$\mathcal{E}(r) = \underbrace{C_o \sum_{d=1}^{r} (r - d) p_i}_{\text{Expected Overage Cost}} + \underbrace{C_s \sum_{d=r+1}^{m} (d - r) p_i}_{\text{Expected Shortage Cost}}. \tag{6.26}$$

Let us define the cumulative probability function of the demand D as follows:

$$F(d) = \sum_{i=1}^{d} p_i = \text{Prob}\,(D \leq d) \quad \text{for} \quad d = 1, 2, \ldots, m. \tag{6.27}$$

We have the following results.

Proposition 6.6.

$$\mathcal{E}(r) - \mathcal{E}(r+1) = C_s - (C_o + C_s)F(r) \tag{6.28}$$

and

$$\mathcal{E}(r) - \mathcal{E}(r-1) = -C_s + (C_o + C_s)F(r-1). \tag{6.29}$$

By using the above proposition and making use of the fact that $F(r)$ is monotonically increasing in r, we have the following proposition.

Proposition 6.7. *The optimal order size r^* is the one which satisfies*

$$F(r^* - 1) < \frac{C_s}{C_s + C_o} \leq F(r^*). \tag{6.30}$$

6.5.1 A Markov Chain Model for the Newsboy Problem

One can further generalize the Newsboy problem as follows. Suppose that the demand is governed by a Markov chain, i.e., the demand tomorrow depends on the demand today. Again the demand has m possible states. We shall order the states in increasing order. The demand at time t is said to be in state i if the demand is i and is denoted by the vector

$$X_t = (0, \ldots, 0, \underbrace{1}_{i\text{th entry}}, 0 \ldots, 0)^T.$$

We let Q (an $m \times m$ matrix) be the transition probability matrix of the Markov process of the demand. Therefore we have

$$X_{t+1} = QX_t.$$

Here we assume that Q is irreducible and hence the stationary probability distribution S exists, i.e.

$$\lim_{t \to \infty} X_t = S = (s_1, s_2, \ldots, s_m)^T.$$

Now we let $r_j \in \{1, 2, \ldots, m\}$ be the size of the next order given that the current demand is j, and $C(r_j, i)$ be the cost of the situation that the size of order is r_j and the actual next demand is i. We note that $C(r_j, i)$ is a more general cost than the one in (6.26). Clearly the optimal ordering policy depends on the state of the current demand because the demand probability distribution in the next period depends on the state of the current demand. The expected cost is then given by

$$\mathcal{E}(\{r_1, r_2, \ldots, r_m\}) = \sum_{j=1}^{m} s_j \times \left(\sum_{i=1}^{m} C(r_j, i) q_{ij} \right), \tag{6.31}$$

where $q_{ij} = [Q]_{ij}$ is the transition probability of the demand from the state j to the state i. In other words, q_{ij} is the probability that the next demand will be in state i given that the current demand is in state j. The optimal ordering policy

$$(r_1^*, r_2^*, \ldots, r_m^*)$$

is the one which minimizes (6.31). We observe that if the current demand is j, then we only need to choose the ordering size r_j to minimize the expected cost. Since

$$\min_{r_j} \mathcal{E}(\{r_1, r_2, \ldots, r_m\}) = \sum_{j=1}^{m} s_j \times \left(\min_{r_j} \sum_{i=1}^{m} C(r_j, i) q_{ij} \right), \tag{6.32}$$

the optimal ordering size r_j^* can be obtained by solving

$$\min_{r_j} \left\{ \sum_{i=1}^{m} C(r_j, i) q_{ij} \right\}. \tag{6.33}$$

By using Proposition 6.7, we have

Proposition 6.8. *If*

$$C(r_j, i) = \begin{cases} C_o(r_j - i) \text{ if } r_j \geq i \\ C_s(i - r_j) \text{ if } r_j < i \end{cases} \tag{6.34}$$

and let

$$F_j(k) = \sum_{i=1}^{k} q_{ij}$$

then the optimal ordering size r_j^* satisfies

$$F_j(r_j^* - 1) < \frac{C_s}{C_s + C_o} \leq F_j(r_j^*).$$

We remark that one has to estimate q_{ij} before one can apply the Markov chain model. We will propose an estimation method for q_{ij} as discussed in the previous section. We note that when $q_{ij} = q_i$ for $i, j = 1, 2, \ldots, m$, (the demand distribution is stationary and independent of the current demand state) then the Markov Newsboy model described above reduces to the classical Newsboy problem. Let us consider an example to demonstrate that the extension to a Markov chain model is useful and important.

Example 6.9. Suppose that the demand $(1, 2, \ldots, 2k)$ $(m = 2k)$ follows a Markov process, with the transition probability matrix Q of size $2k \times 2k$ given by

$$Q = \begin{pmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \ddots & & 0 \\ 0 & 1 & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & 0 \end{pmatrix}, \tag{6.35}$$

and the cost is given in (6.34) with $C_o = C_s$. Clearly the next demand can be determined precisely by the state of the current demand, and hence the optimal expected cost is equal to zero when the Markov chain model is used. When the classical Newsboy model is used, we note that the stationary distribution of Q is given by

$$\frac{1}{2k}(1, 1, \ldots, 1)^T.$$

The optimal ordering size is equal to k by Proposition 6.7 and therefore the optimal expected cost is $C_o k$.

According to this example, it is obvious that the more "information" one can extract from the demand sequence, the better the model will be and hence the better the optimal ordering policy one can obtain. Therefore it is natural for one to consider a higher-order Markov chain model. The only obstacle here is the huge number of states and parameters. We employ a higher-order Markov chain model that can cope with the difficulty.

Let us study the optimal ordering policy for this higher-order Markov chain model. Define the set

$$\Phi = \{G = (j_1, j_2, \ldots, j_n)^T \mid j_k \in \{1, 2, \ldots, m\} \text{ for } k = 1, 2, \ldots, n\}.$$

let

$$p_{i,G} = P(X_{t+n+1} = E_i \mid X_{t+1} = E_{j_1}, X_{t+2} = E_{j_2}, \ldots, X_{t+n} = E_{j_n}\}$$

$(G = (j_1, j_2, \ldots, j_n)^T)$ to be the probability that the demand at time $(t+n+1)$ is i given that the demand at the time $t + k$ is $j_k \in \{1, 2, \ldots, m\}$ for $k = 1, 2, \ldots, n$. Here E_i is a unit vector representing the state of demand. This means that the demand distribution at time $(t + n + 1)$ depends only on the states of the demand at the time $t + 1, t + 2, \ldots, t + n$, and this is also true for the optimal ordering policy. In the higher-order Markov chain model (7.12), we have

$$p_{i,G} = \sum_{i=1}^{n} \lambda_i Q_i E_{j_i}.$$

Under some practical conditions as described in previous sections, one can show that

$$\lim_{t\to\infty} P(X_{t+1} = E_{j_1}, X_{t+2} = E_{j_2}, \ldots, X_{t+n} = E_{j_n}) = s_G,$$

where s_G is independent of t. Let

$$r_G, \quad (G = (j_1, j_2, \ldots, j_n)^T)$$

be the ordering policy when the demands of the previous n periods are $j_1, j_2, \ldots, j_n$. The expected cost for all ordering policies $G \in \Phi$ is then given by

$$\mathcal{E}(\Phi) = \sum_{G\in\Phi} s_G \left(\sum_{i=1}^{m} C(r_G, i) p_{i,G} \right). \tag{6.36}$$

The optimal ordering policy $\{r_G^* \mid G \in \Phi\}$ is the one which minimizes (6.36). We remark that the computational complexity for computing all the optimal ordering policies r_G^* is of $O(m^n)$ operations because $|\Phi| = m^n$. However, we observe that if the demands of the previous n periods are $j_1, j_2, \ldots, j_n$, then we only need to solve the ordering size r_G which minimizes the expected cost. Since

$$\min_{r_G} \mathcal{E}(\Phi) = \sum_{j=1}^{m} s_G \times \left(\min_{r_G} \sum_{i=1}^{m} C(r_G, i) p_{i,G} \right), \tag{6.37}$$

the optimal ordering size r_G^* can be obtained by solving

$$\min_{r_G} \sum_{i=1}^{m} C(r_G, i) p_{i,G}, \quad r_G \in \{1, 2, \ldots, m\}.$$

By Proposition 6.7 again, if

$$C(r_G, i) = \begin{cases} C_o(r_G - i) \text{ if } r_G \geq i \\ C_s(i - r_G) \text{ if } r_G < i, \end{cases}$$

and let

$$F_G(k) = \sum_{i=1}^{k} p_{i,G},$$

then the optimal ordering size r_G^* satisfies the inequalities

$$F_G(r_G^* - 1) < \frac{C_s}{C_s + C_o} \leq F_G(r_G^*).$$

Therefore, in order to compute the optimal ordering size, the main task here is to estimate the probabilities $p_{i,G}$ or equivalently to estimate the parameters λ_i and Q_i based on the observed data sequence.

Table 6.2 The optimal costs of the three different models

	Product A	Product B	Product C
Third-order Markov model	11,200	9,300	10,800
First-order Markov model	27,600	18,900	11,100
Stationary model	31,900	18,900	16,300

6.5.2 A Numerical Example

In this subsection, we present an application of the higher-order Markov model to a generalized Newsboy problem [51]. The background is that a large soft-drink company faces an in-house problem of production planning and inventory control. There are three types of products A, B and C having five different possible sales volume $(1, 2, 3, 4$ and $5)$. Such labelling is useful from both marketing and production planning points of view. The categorical data sequences for the demand for three products of the soft-drink company for some sales periods can be found in [51]. Based on the sales demand data, we build the higher-order Markov models of different orders. These models are then applied to the problem of long-run production planning and the following cost matrix is assumed

$$C = \begin{pmatrix} 0 & 100 & 300 & 700 & 1500 \\ 100 & 0 & 100 & 300 & 700 \\ 300 & 100 & 0 & 100 & 300 \\ 700 & 300 & 100 & 0 & 100 \\ 1500 & 700 & 300 & 100 & 0 \end{pmatrix}. \tag{6.38}$$

Here $[C]_{ij}$ is the cost when the production plan is for sales volume of state i and the actual sales volume is state j. We note that the costs here are non-linear, i.e. $[C]_{ij} \neq c|i - j|$, where c is a positive constant. When the unsatisfied demand is higher, the shortage cost is larger. Similarly, when the holding product is more, the overage cost is larger. For the higher-order Markov model, we find that the third-order model gives the best optimal cost. Here we also report the results on the first-order model and the stationary model for the three product demand sequences. The results are given in Table 6.2 (Taken from [51]).

6.6 Higher-Order Markov Regime-Switching Model for Risk Measurement

In this section, we discuss an application of the higher-order Markov model in risk measurement. The focus here is on the use of the higher-order Markov chain as the modulating Markov chain for a Markov regime-switching model. Firstly, we give a brief introduction to Markov regime-switching models. Then we establish the modeling framework based on a higher-order Markov regime-switching (HMRS) model, and evaluate risk measures using the HMRS model. This section is based on some material in [189].

6.6.1 A Snapshot for Markov Regime-Switching Models

Markov regime-switching models are one of the major classes of models in (financial) econometrics. The basic idea of Markov regime-switching models is to allow the model parameters, (or coefficients), to switch over time according to the state of an underlying process described by a Markov chain. This Markov chain is called a modulating Markov chain. In economic and financial applications, the states of the modulating Markov chain are usually interpreted as the underlying states of an economy. Consequently, Markov regime-switching models provide flexibility in incorporating the impact of the states of the economy in modeling economic and financial dynamics, such as asset price dynamics. In physical and engineering sciences, the states of the modulating Markov chain may be interpreted as different states of physical and engineering systems such as energy modes in high energy physics, modes of manufacturing machines and modes of electricity generators.

Indeed, Markov regime-switching models have a long history in engineering science; in statistics and economics, their history can be tracked back to works as early as [107, 174]. The idea of Markov regime switching has also appeared in one of the pioneering works on parametric nonlinear time series analysis by Tong [197, 200], (see [198, 199]), where one of the oldest nonlinear time series models, namely, the class of Threshold AutoRegressive (TAR) time series models, was first introduced. This class of parametric nonlinear time series models is general and flexible enough to accommodate a number of interesting cases, including the Self-exciting Threshold AutoRegressive (SETAR) time series models and Markov regime-switching time series models. Hamilton [112] pioneered the econometric applications of Markov regime-switching models. Since then, the Markov regime-switching models become increasingly popular in economics and finance. Recently there has been phenomenal growth in the applications of Markov regime-switching models in discussing various practical problems in finance. Some examples include [86, 94, 96, 211] for optimal portfolio allocation, [90, 93, 171, 185, 186, 215] for modeling stochastic interest rate, [29, 30, 85, 95, 110, 166, 184] for option valuation, [88, 89, 91, 97] for volatility estimation, [92, 189] for risk measurement. A recent monograph [156] presents the cutting edge applications of regime-switching models to various practical problems in economics, finance and actuarial science.

In what follows, we discuss the basic idea of a Markov regime-switching model using a simple discrete-time two-regime model as an example. Here we suppose that the modulating Markov chain is a standard Markov chain, (i.e., a first-order Markov chain). To make our discussion more concrete, we adopt the two-regime model to describe the logarithmic return of a risky asset.

Consider a first-order two-state Markov chain $X := \{X^{(i)}, 1 \leq i \leq T\}$ with length T and the following transition probabilities:

$$P(X^{(n)} = j_0 | X^{(n-1)} = j_1) = q_{j_0, j_1}, \quad j_0, j_1 \in \{1, 2\}. \tag{6.39}$$

Here $\{1, 2\}$ is the state space of the Markov chain X. If $X^{(n)} = 1$, we say that the economy is in a "Good" state. Whereas, if $X^{(n)} = 2$, we say that the economy in a "Bad" state.

Let $\{\xi_t, 1 \leq t \leq T\}$ be a sequence of independent and identically distributed, (i.i.d.), random variables with zero mean and unit variance and suppose $\{Y_t, 1 \leq t \leq T\}$ is the sequence of logarithmic returns of a risky asset. For each $t = 1, 2, \ldots, T$, let μ_t be the expected rate of return, (i.e., the appreciation rate), of the risky asset in the t^{th} period. Similarly, let σ_t be the volatility of the risky asset in the t^{th} period. We suppose that the expected rate of return μ_t and the volatility σ_t are modulated by the Markov chain X as:

$$\mu_t := \mu(X^{(t)}), \quad \sigma_t := \sigma(X^{(t)}).$$

So if $X^{(t)} = 1$, (i.e., the economy in the t^{th} period is good), then

$$\mu_t = \mu(1), \quad \sigma_t = \sigma(1).$$

Otherwise,

$$\mu_t = \mu(2), \quad \sigma_t = \sigma(2).$$

It is reasonable and intuitive to assume that $\mu(1) > \mu(2)$ and $\sigma(1) < \sigma(2)$.

Now we describe the evolution of the logarithmic return of the risky asset over time as the following discrete-time, two-regime, Markov, regime-switching, model:

$$Y_t = \mu(X^{(t)}) + \sigma(X^{(t)})\xi_t.$$

So if the economy is good, the dynamics of the logarithmic return are given by:

$$Y_t = \mu(1) + \sigma(1)\xi_t.$$

Similarly when the economy is bad, the dynamics of the logarithmic return are given by:

$$Y_t = \mu(2) + \sigma(2)\xi_t.$$

The modulating Markov chain can be assumed as either observable or hidden. If we assume that the Markov chain is observable, the model for logarithmic return is completely observed, and we can interpret the states of the Markov chain as different levels of proxies of some macro-economic factors, such as gross domestic product and retail price index. If we suppose that the Markov chain is hidden, then the model for logarithmic return is partially observed, and the states of the Markov chain can be interpreted as hidden states of the economy. The model based on a hidden Markov chain appears to be more general than its counterpart based on an observable Markov chain. Furthermore, in the situation of a hidden Markov chain, filtering is often involved in the implementation of the model. In this chapter and Chap. 8, we consider a general situation where the modulating Markov chain is hidden.

6.6.2 A Risk Measurement Framework Based on a HMRS Model

The main motivation for considering a HMRS model for risk measurement is that many real-world economic and financial time series possess memories and these memories in economic and financial time series may have significant economic consequences. The empirical phenomenon of (long-term) memories, is coined as the Joseph effect, see for example [78, 153]. It is known that Markov chains serve as reasonable approximations to (continuous state) time series models. In the same vein, the higher-order Markov chain, which is also called the weak Markov chain (see for instance [201, 202, 205]), provides a feasible and convenient way to approximate time series models with memories.

In this subsection we present a HMRS model which was introduced in [189]. We only highlight the main results here, and interested readers may refer to [189] for details. The central tenet of the HMRS model is that the expected rate of return and the volatility of a risky portfolio are modulated by a discrete-time, finite-state, higher-order Markov chain. Note that instead of modeling returns of individual risky assets as in the last subsection, we consider the model at a portfolio level and describe the dynamics of the portfolio's return. The rationale of the HMRS model is to incorporate a regime-switching effect with long-term memory in modeling the dynamics of the portfolio's returns. This is different from some existing time series models with long-term memories, where the effect of long-term memories is incorporated in the innovations terms, or the error terms. In what follows, we describe the mathematical set up of the HMRS.

Let $\mathcal{T}$ be the time parameter set $\{1, 2, \cdots, \}$ of the economy. Unlike the previous subsection, we consider an infinite-horizon, discrete-time situation for the sake of generality. To describe uncertainty, we consider a complete probability space $(\Omega, \mathcal{F}, P)$, where P is a real-world probability measure. Note that for the purpose of risk measurement, a real-world probability measure should be used.

Let $V := \{V_t, t \in \mathcal{T}\}$ be a discrete-time, higher-order Markov chain defined on $(\Omega, \mathcal{F}, P)$ with state space

$$\mathcal{V} := \{v_1, v_2, \ldots, v_M\}.$$

Here we suppose that the Markov chain is hidden and that the states of the chain represent different states of a hidden economy. We may interpret v_1 as the "best" economic state, v_2 as the second "best" economic state and v_M as the "worst" economic state, etc.

We consider here the situation that V is an l^{th}-order Markov chain. For each $l = 1, 2, \cdots$ and each $t \geq l - 1$, let

$$\mathbf{i}(t, l) := (i_t, i_{t-1}, \ldots, i_{t-l+1}),$$

where $i_t, i_{t-1}, \ldots, i_{t-l+1} \in \{1, 2, \ldots, M\}$.

We note that $\mathbf{i}(t, l)$ represents the indices of the states of the Markov chain from time $t - l + 1$ to t inclusively. In other words, given that

$$\mathbf{i}(t, l) := (i_t, i_{t-1}, \ldots, i_{t-l+1}),$$

we have

$$V_t = v_{i_t}, V_{t-1} = v_{i_{t-1}} \ldots, V_{t-l+1} = v_{i_{t-l+1}}.$$

To specify the probability laws of the l^{th}-order Markov chain, we define a set of state transition probabilities by putting:

$$\begin{aligned} &P(i_{t+1}|\mathbf{i}(t, l)) := P[V_{t+1} = v_{i_{t+1}}|V_t = v_{i_t}, \ldots, V_{t-l+1} = v_{i_{t-l+1}}], \\ &i_{t+1} = 1, 2, \cdots, M. \end{aligned} \tag{6.40}$$

The order l represents the degree of the long-term memory of the states of the economy.

To completely determine the probability laws of the chain V, we must specify its initial distributions as follows:

$$P(i_{l+1}|i(l)) := \pi_{i_{l+1}|\mathbf{i}(l)}, \quad \text{for} \quad 0 \leq t < l, i_{l+1} = 1, 2, \ldots, M \tag{6.41}$$

where $\pi_{i_{l+1}|\mathbf{i}(l)}$ is the probability that $V_{l+1} = v_{i_{l+1}}$ given that

$$V_l := v_{i_l}, \;\; V_{l-1} = v_{i_{l-1}}, \;\; \cdots, \;\; V_1 = v_{i_1}$$

and $\mathbf{i}(l) = (i_1, i_2, \ldots, i_l)$.

We now specify the HMRS model modulated by the l^{th}-order hidden Markov chain V. Let $\{Y_t, t \in \mathcal{T}\}$ be a sequence of logarithmic returns of a risky portfolio, where Y_t denotes the logarithmic return of the portfolio in the t^{th} period. To simplify our notation, we write $\mathbf{V}_{t,l}$ for $(V_t, V_{t-1}, \ldots, V_{t-l+1})$, for each $t \geq l - 1$, $l = 1, 2, \ldots$.

Let μ_t and σ_t be the expected rate of return and the volatility of the portfolio in the t^{th} period, respectively. We suppose that both the expected rate of return and the volatility are modulated by the l^{th}-order hidden Markov chain V as follows:

$$\mu_t := \mu(\mathbf{V}_{t,l}), \quad \sigma_t := \sigma(\mathbf{V}_{t,l}).$$

In other words, both the expected rate of return and volatility of the portfolio in the t^{th} period depend on the current and past values of the chain V up to lag l.

Let $\{\xi_t, t \in \mathcal{T}\}$ be a sequence of independent and identically distributed (i.i.d.) random variables defined on $(\Omega, \mathcal{F}, P)$, with common distribution $N(0,1)$, a standard normal distribution with zero mean and unit variance. We assume that ξ and V are stochastically independent under P. Then we suppose that the evolution of the logarithmic returns of the portfolio over time is governed by the following HMRS model:

$$Y_t = \mu(\mathbf{V}_{t,l}) + \sigma(\mathbf{V}_{t,l})\xi_t. \tag{6.42}$$

Note that the structure of the HMRS model resembles that of the continuous-state observation process in [87] and that the HMRS is a generalization of the simple Markov, regime-switching, model discussed in the last subsection. In particular, when $l = 1$ and the chain V has two states, the above HMRS model reduces to the Markov, regime-switching, model in the last subsection.

To simplify our discussion, we focus on the situation where $l = 2$, (i.e., a second-order hidden Markov chain). The method presented below can be extended to a general order l. However, the notation in the general case is tedious. When $l = 2$, the dynamics of the logarithmic returns of the portfolio become:

$$Y_t = \mu(V_t, V_{t-1}) + \sigma(V_t, V_{t-1})\xi_t,\ t \in \mathcal{T}. \tag{6.43}$$

Instead of handling the second-order hidden Markov chain directly, we consider a two-dimensional first-order hidden Markov chain X which embeds the second-order chain. By doing so, we can adopt the filtering method for the first-order hidden Markov chains to derive filters for the second-order hidden Markov chains.

Consider now the following two-dimensional hidden Markov chain X defined on $(\Omega, \mathcal{F}, P)$, which embeds V:

$$X_t := (V_t, V_{t-1}). \tag{6.44}$$

Let $\mathcal{X}$ be an $(M \times M)$-matrix with the (i,j)-element

$$x_{ij} := (v_i, v_j), \quad i, j = 1, 2, \ldots, M,$$

so that $\mathcal{X}$ is the state space of the two-dimensional first-order hidden Markov chain X.

Define $\tilde{\mathcal{X}} := vec(\mathcal{X})$ where $vec(\cdot)$ is the column-by-column vectorization function. Then $\tilde{\mathcal{X}}$ is an M^2-dimensional column vector. In particular, the $((j-1)M + i)^{th}$-element $\tilde{x}_{(j-1)M+i}$ of $\tilde{\mathcal{X}}$ is given by $x_{ij} := (v_i, v_j)$. Consequently, we can define a one-dimensional first-order hidden Markov chain $\tilde{X}$, induced by the two-dimensional first-order hidden Markov chain X, such that

$$\tilde{X}_t = \tilde{x}_{(j-1)M+i}$$

whenever $X_t = x_{ij}$.

Following the treatment in [83], without loss of generality, we identify the state space of the chain $\tilde{X}$ with a set of standard unit vectors in $\mathcal{R}^{M^2}$:

$$\mathcal{E} := \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_{M^2}\}$$

with the i^{th} component of e_j being the Kronecker delta function δ_{ij}, for each $i, j = 1, 2, \cdots, \mathcal{R}^{M^2}$.

The use of $\mathcal{E}$ to be the state space of the chain $\tilde{X}$ facilitates the use of mathematics, and this state space is called the canonical state space of the chain $\tilde{X}$.

Again, to specify the probability laws of the chain $\tilde{X}$, we define an $(M^2 \times M^2)$-matrix representing the time-independent, (homogeneous), transition probability matrix of the first-order Markov chain $\tilde{X}$. The (j, k)-element a_{jk} of A, $(j, k = 1, 2, \ldots, M^2)$, is given by

$$a_{jk} := P(\tilde{X}_t = \mathbf{e}_j | \tilde{X}_{t-1} = \mathbf{e}_k). \tag{6.45}$$

Let $F^{\tilde{X}} := \{\mathcal{F}_t^{\tilde{X}} | t \in \mathcal{T}\}$ be the right-continuous, P-completed, natural filtration generated by the chain $\tilde{X}$, where $\mathcal{F}_t^{\tilde{X}}$ is the minimal σ-field generated by the information about $\tilde{X}$ up to and including time t and all P-null sets in $\mathcal{F}$. Then with the canonical state space $\mathcal{E}$ of the chain $\tilde{X}$, the following semimartingale dynamics for the chain $\tilde{X}$ are obtained in [84]:

$$\tilde{X}_t := A\tilde{X}_{t-1} + L_t. \tag{6.46}$$

Here L is an $\mathcal{R}^{M^2}$-valued, $(F^{\tilde{X}}, P)$-martingale difference process.

We now specify the structure of information in our model. For each $t \in \mathcal{T}$, let $\mathcal{F}_t^Y$ and $\mathcal{F}_t^V$ be the σ-fields generated by the return process Y and the hidden Markov chain V up to and including time t, respectively. Note that $\mathcal{F}_t^Y$ represents observable information at time t.

For each $i, j = 1, 2, \ldots, M$, let

$$\phi_{ij}(x) := \frac{1}{\sqrt{2\pi\sigma_{ij}^2}} \exp\left(- \frac{1}{2\sigma_{ij}^2} x^2 \right).$$

This is the probability density function of a normal distribution $N(0, \sigma_{ij}^2)$ with mean zero and variance σ_{ij}^2.

Then it has been shown in [189] that the predictive distribution of $F_{Y_{t+1}}(y|\mathcal{F}_t^Y)$ of Y_{t+1} given $\mathcal{F}_t^Y$ under P is given by:

$$F_{Y_{t+1}}(y|\mathcal{F}_t^Y) = \sum_{i=1}^{M} \sum_{j=1}^{M} P(\tilde{X}_t = \tilde{x}_{(j-1)M+i} | \mathcal{F}_t^Y) \int_{-\infty}^{y-\mu_{ij}} \phi_{ij}(x) dx,$$

and so

$$f_{Y_{t+1}}(y|\mathcal{F}_t^Y) = \sum_{i=1}^{M} \sum_{j=1}^{M} P(\tilde{X}_t = \tilde{x}_{(j-1)M+i} | \mathcal{F}_t^Y) \phi_{ij}(y - \mu_{ij}).$$

For each $t \in \mathcal{T}$, let $\tilde{X}_t^Y := E(\tilde{X}_t|\mathcal{F}_t^Y)$, where E is the expectation under the measure P. Then

$$F_{Y_{t+1}}(y|\mathcal{F}_t^Y) = \sum_{i=1}^{M}\sum_{j=1}^{M}\big\langle \tilde{X}_t^Y, \mathbf{e}_{(j-1)M+i}\big\rangle \int_{-\infty}^{y-\mu_{ij}} \phi_{ij}(x)dx,$$

and

$$f_{Y_{t+1}}(y|\mathcal{F}_t^Y) = \sum_{i=1}^{M}\sum_{j=1}^{M}\big\langle \tilde{X}_t^Y, \mathbf{e}_{(j-1)M+i}\big\rangle \phi_{ij}(y-\mu_{ij}),$$

where $\langle\cdot,\cdot\rangle$ denotes a scalar product in $\mathcal{R}^{M^2}$.

Using a version of the Bayes' rule, a recursive filter for $\tilde{X}_t^Y$ can be obtained as follows:

$$\begin{aligned}\tilde{X}_{t+1}^Y &:= E(\tilde{X}_{t+1}|\mathcal{F}_{t+1}^Y)\\ &= \frac{\sum_{i=1}^{M}\sum_{j=1}^{M}\big\langle \tilde{X}_t^Y, \mathbf{e}_{(j-1)M+i}\big\rangle \phi_{ij}(y_{t+1}-\mu_{ij})A\mathbf{e}_{(j-1)M+i}}{\sum_{i=1}^{M}\sum_{j=1}^{M}\big\langle \tilde{X}_t^Y, \mathbf{e}_{(j-1)M+i}\big\rangle \phi_{ij}(y_{t+1}-\mu_{ij})}. \quad (6.47)\end{aligned}$$

This filtered estimate $\tilde{X}_{t+1}^Y$ is optimal among all linear estimates in the sense of mean-square loss. This is left as an exercise.

6.6.3 Value at Risk Forecasts

Value-at-Risk (VaR) has emerged as one of the most prominent tools in the finance and insurance industries. Many regulatory bodies, financial institutions and insurance companies adopt VaR as a measure for risk. Technically speaking, VaR is a statistical estimation of a portfolio's loss, where the owner of the portfolio expects to incur that loss or more with a given probability level over a specified time horizon for risk measurement (see for example, [17, 81, 127]). For example, if the daily VaR at a 95 % confidence level is US$ 1 million, then there is a 0.95 chance that the actual loss will not exceed the amount of US$ 1 million in the next day. For the practical implementation of VaR, refer to J.P. Morgan's RiskMetrics—Technical Document and the monograph by McNeil et al. [157]. Basically, there are two common approaches to the VaR implementation, namely, (a) the historical simulation and (b) the model-based approach. The historical simulation is to calculate VaR based on the empirical distribution of historical data by bootstrapping. This method is non-parametric in the sense that stringent assumptions for the profit/loss distribution are not required. The model-based method assumes a parametric form of the profit/loss distribution and estimates the unknown parameters using historical data.

Although VaR is a popular tool in the practice of risk measurement and management in the finance and insurance industries, [5, 6] have pointed out the theoretical shortcomings of VaR. In particular, VaR does not satisfy the sub-additive property saying that the merge of two risky positions does not reduce risk. This is counter-intuitive from the perspective of diversification, which is a key theme in modern finance. Furthermore, VaR remains silent about the severity of losses when the losses exceed a certain threshold level. Despite these theoretical shortcomings, VaR still remains a popular measure of risk in practice due to its simplistic interpretation. It provides both an easy-to-understand risk measure in financial reporting and computational tractability under some specific parametric assumptions, such as the multivariate normality assumption. It is also worth noting that VaR satisfies the sub-additive property when the profit/loss distribution is in the elliptical class (i.e., symmetrical), which includes normal distributions as a special case. In what follows, we discuss the evaluation of VaR forecasts using the HMRS model presented in the last subsection.

Suppose $q_{t+1|t}(\alpha)$ is the α-quantile of the predictive distribution of Y_{t+1} given $\mathcal{F}_t^Y$ under P. By definition,

$$F_{Y_{t+1}}(q_{t+1|t}(\alpha)|\mathcal{F}_t^Y) = \alpha, \tag{6.48}$$

where $F_{Y_{t+1}}(y|\mathcal{F}_t^Y)$ is the predictive distribution of Y_{t+1} given $\mathcal{F}_t^Y$ under P in the second-order HMRS model derived in the last subsection.

Let PV_t be the market value of the portfolio at time t. The VaR, denoted by $VaR_{t+1|t}(\alpha)$, for the long position of the portfolio with probability level α is defined as the α-percentile, (in practice, α can be $1\,\%$ or $5\,\%$), of the loss distribution. It is easy to see that

$$VaR_{t+1|t}(\alpha) = PV_t[1 - \exp(q_{t+1|t}(\alpha) - r)]. \tag{6.49}$$

In [189], the performance of VaR forecasts implied by the second-order HMRS model were evaluated using backtesting, which is a standard procedure to evaluate the performance of VaR forecasts used in the finance and insurance industries. The numerical results of the backtesting reveal that the degree of long-term memory described by the order of the HMRS model has a significant impact on the accuracy of VaR forecasts. For details, please refer to Sect. 3 of [189].

6.7 Summary

In this chapter, a higher-order Markov chain model is proposed with efficient estimation methods for the model parameters. A further extension of the model is also discussed. The higher-order Markov chain model and its extension are then applied to diverse fields including sales demand predictions, web page predictions, the Newsboy problem and financial risk management.

6.8 Exercise

1. Consider the following categorical sequence of three states:

$$1, 1, 2, 3, 2, 1, 1, 1, 2, 2, 3, 3, 2, 1, 2, 3, 2, 1, 2, 2, 2, 1, 2, 3, 1.$$

   Build a third-order Markov chain model, as discussed in Sect. 6.2, by using the Euclidean norm $||.||_2$ in solving the parameters.
2. Prove Proposition 6.7.
3. Suppose the demand sequence of the product discussed in Sect. 6.5 is given as follows:

$$1, 2, 5, 2, 3, 4, 4, 3, 4, 5, 1, 2, 2, 2, 4, 4, 4, 5, 4, 5, 3, 2, 2, 2, 3, 4, 4, 5, 5, 5, 4, 4, 3, 2.$$

   (a) Construct a second-order Markov chain model by using the Euclidean norm $||.||_2$ in solving the parameters.
   (b) Suppose the following non-symmetric cost matrix (compare with (6.38)) is employed:

$$C = \begin{pmatrix} 0 & 200 & 400 & 800 & 2000 \\ 100 & 0 & 200 & 400 & 800 \\ 300 & 100 & 0 & 200 & 400 \\ 700 & 300 & 100 & 0 & 200 \\ 1500 & 700 & 300 & 100 & 0 \end{pmatrix}.$$

   Find the optimal production policy when the second-order Markov chain model is adopted.
4. Derive the semi-martingale dynamics for the chain $\tilde{X}$ in (6.46).
5. Derive the recursive filter $\tilde{X}_t^Y$ in (6.47).
6. Show that the filter estimate $\tilde{X}_t^Y$ in (6.47) is optimal over the space of all linear estimates in the mean-square-loss sense.
7. Write a Matlab program to compute the filter $\tilde{X}_t^Y$ in (6.47).

Chapter 7
Multivariate Markov Chains

7.1 Introduction

By making use of the transition probability matrix in Chap. 6, a categorical data sequence of m states can be modeled by an m-state Markov chain model. In this chapter, we extend this idea to model multiple categorical data sequences. One would expect categorical data sequences generated by similar sources or the same source to be correlated to each other. Therefore, by exploring these relationships, one can develop better models for the categorical data sequences and hence better prediction rules.

The outline of this chapter is as follows. In Sect. 7.1, we present the multivariate Markov chain model with estimation methods for the model parameters. In Sect. 7.3, we apply the model to the multi-product demand estimation problem. In Sect. 7.4, an application to credit rating is discussed. In Sect. 7.5, we extend the model to a higher-order multivariate Markov chain model. Section 7.6 discusses an improved model with application to dependency rating transition. Finally, a summary is given in Sect. 7.7 to conclude the chapter.

7.2 Construction of Multivariate Markov Chain Models

In this section, we propose a multivariate Markov chain model to represent the behavior of multiple categorical sequences generated by similar sources or the same source. Here we assume that there are s categorical sequences and each has m possible states in the set

$$M = \{1, 2, \ldots, m\}.$$

Let $\mathbf{X}_n^{(j)}$ be the state vector of the jth sequence at time n. If the jth sequence is in state l at time n then we write

W.-K. Ching et al., *Markov Chains*, International Series in Operations Research & Management Science 189, DOI 10.1007/978-1-4614-6312-2_7,

$$\mathbf{X}_n^{(j)} = \mathbf{e}_l = (0, \ldots, 0, \underbrace{1}_{l\text{th entry}}, 0 \ldots, 0)^t.$$

In the proposed multivariate Markov chain model, we assume the following relationship:

$$\mathbf{X}_{n+1}^{(j)} = \sum_{k=1}^{s} \lambda_{jk} P^{(jk)} \mathbf{X}_n^{(k)}, \quad \text{for} \quad j = 1, 2, \ldots, s \tag{7.1}$$

where

$$\lambda_{jk} \geq 0, \quad 1 \leq j, k \leq s \tag{7.2}$$

and

$$\sum_{k=1}^{s} \lambda_{jk} = 1, \quad \text{for} \quad j = 1, 2, \ldots, s. \tag{7.3}$$

The state probability distribution of the kth sequence at time $(n+1)$ depends on the weighted average of $P^{(jk)}\mathbf{X}_n^{(k)}$. Here $P^{(jk)}$ is a transition probability matrix from the states in the kth sequence to the states in the jth sequence, and $\mathbf{X}_n^{(k)}$ is the state probability distribution of the kth sequences at time n. In matrix form, we write

$$\mathbf{X}_{n+1} \equiv \begin{pmatrix} \mathbf{X}_{n+1}^{(1)} \\ \mathbf{X}_{n+1}^{(2)} \\ \vdots \\ \mathbf{X}_{n+1}^{(s)} \end{pmatrix} = \begin{pmatrix} \lambda_{11} P^{(11)} & \lambda_{12} P^{(12)} & \cdots & \lambda_{1s} P^{(1s)} \\ \lambda_{21} P^{(21)} & \lambda_{22} P^{(22)} & \cdots & \lambda_{2s} P^{(2s)} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{s1} P^{(s1)} & \lambda_{s2} P^{(s2)} & \cdots & \lambda_{ss} P^{(ss)} \end{pmatrix} \begin{pmatrix} \mathbf{X}_n^{(1)} \\ \mathbf{X}_n^{(2)} \\ \vdots \\ \mathbf{X}_n^{(s)} \end{pmatrix}$$
$$\equiv Q\mathbf{X}_n$$

or

$$\mathbf{X}_{n+1} = Q\mathbf{X}_n.$$

Although the column sum of Q is not equal to one (the column sum of $P^{(jk)}$ is equal to one), we still have the following proposition.

Proposition 7.1. *If the parameters $\lambda_{jk} > 0$ for $1 \leq j, k \leq s$, then the matrix Q has an eigenvalue equal to one and the eigenvalues of Q have moduli less than or equal to one.*

Proof. From (7.2) and (7.3), the column sum of the following matrix

$$\Lambda = \begin{pmatrix} \lambda_{1,1} & \lambda_{2,1} & \cdots & \lambda_{s,1} \\ \lambda_{1,2} & \lambda_{2,2} & \cdots & \lambda_{s,2} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{1,s} & \lambda_{2,s} & \cdots & \lambda_{s,s} \end{pmatrix}$$

is equal to one. Since $\lambda_{jk} > 0$, Λ is nonnegative and irreducible. By Perron-Frobenius Theorem, there exists a vector

$$\mathbf{y} = (y_1, y_2, \ldots, y_s)^T$$

such that

$$\mathbf{y}^T \Lambda = \mathbf{y}^T.$$

We note that

$$\mathbf{1}_m P^{(ij)} = \mathbf{1}_m, \quad 1 \le i, j \le s,$$

where $\mathbf{1}_m$ is the $1 \times m$ vector of all ones, i.e.,

$$\mathbf{1}_m = (1, 1, \ldots, 1).$$

Then it is easy to show that we have

$$(y_1\mathbf{1}_m, y_2\mathbf{1}_m, \ldots, y_s\mathbf{1}_m)Q = (y_1\mathbf{1}_m, y_2\mathbf{1}_m, \ldots, y_s\mathbf{1}_m).$$

and hence 1 must be an eigenvalue of Q.

We then show that all the eigenvalues of Q are less than or equal to one. Let us define the following vector-norm

$$||\mathbf{z}||_V = \max_{1 \le i \le s} \left\{||\mathbf{z}_i||_1 : \mathbf{z} = (\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_s),\ \mathbf{z}_j \in \mathbf{R}^m,\ 1 \le j \le s\right\}.$$

It is straightforward to show that $||\cdot||_V$ is a vector-norm on $\mathbf{R}^{ms}$. It follows that we can define the following matrix norm

$$||Q||_M \equiv \sup\{||Q\mathbf{z}||_V : ||\mathbf{z}||_V = 1\}.$$

Since $P^{(ij)}$ is a transition matrix, each element of $P^{(ij)}$ are less than or equal to 1. We have

$$\|P^{(ij)}\mathbf{z}_j\|_1 \le \|\mathbf{z}_j\|_1 \le 1, \quad 1 \le i, j \le s.$$

Here $||.||_1$ is the 1-norm for a vector. It follows that

$$\|\lambda_{i1}P^{(i1)}\mathbf{z}_1 + \lambda_{i2}P^{(i2)}\mathbf{z}_2 + \cdots + \lambda_{is}P^{(is)}\mathbf{z}_s\|_1 \le ||\mathbf{z}||_V \cdot \sum_{j=1}^{s} \lambda_{ij} = 1,\ 1 \le i \le s$$

and hence $||Q||_M \le 1$. Since the spectral radius of Q is always less than or equal to any matrix norm of Q, the result follows.

Proposition 7.2. *Suppose that the matrices $P^{(jk)}$ $(1 \le j, k \le s)$ are irreducible and $\lambda_{jk} > 0$ for $1 \le j, k \le s$. Then there is a unique vector*

$$\mathbf{x} = (\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(s)})^T$$

such that $\mathbf{x} = Q\mathbf{x}$ *and*

$$\sum_{i=1}^{m}[\mathbf{x}^{(j)}]_i = 1,\ 1 \le j \le s.$$

Proof. By Proposition 7.1, there is exactly one eigenvalue of Q equal to one. This implies that

$$\lim_{n\to\infty} Q^n = \mathbf{v}\mathbf{u}^T$$

is a positive rank one matrix as Q is irreducible. Therefore we have

$$\lim_{n\to\infty} \mathbf{x}_{n+1} = \lim_{n\to\infty} Q\mathbf{x}_n = \lim_{n\to\infty} Q^n\mathbf{x}_0 = \mathbf{v}\mathbf{u}^T\mathbf{x}_0 = \alpha\mathbf{v}.$$

Here α is a positive number since $\mathbf{x} \neq 0$ and is nonnegative. This implies that $\mathbf{x}_n$ tends to a stationary vector as n goes to infinity. Finally, we note that if $\mathbf{x}_0$ is a vector such that

$$\sum_{i=1}^{m}[\mathbf{x}_0^{(j)}]_i = 1,\ 1 \le j \le s,$$

then $Q\mathbf{x}_0$ and $\mathbf{x}$ are also vectors having this property.

Now suppose that there exists $\mathbf{y}$ such that $\mathbf{y} \neq \mathbf{x}$ and

$$\mathbf{y} = \lim_{n\to\infty} \mathbf{x}_n.$$

Then we have

$$||\mathbf{x}-\mathbf{y}|| = ||\mathbf{x} - Q\mathbf{x}|| = 0.$$

This is a contradiction and therefore the vector $\mathbf{x}$ must be unique. Hence the result follows.

We note that $\mathbf{x}$ is not a probability distribution vector, but $\mathbf{x}^{(j)}$ is a probability distribution vector. The above proposition suggests one possible way to estimate the model parameters λ_{ij}. The idea is to find λ_{ij} which minimizes $||Q\hat{\mathbf{x}} - \hat{\mathbf{x}}||$ under a certain vector norm $||\cdot||$.

We remark that the result can be further extended to the following [216].

Proposition 7.3. *If* $\lambda_{ii} > 0$*,* P_{ii} *is irreducible (for* $1 \le i \le s$*), the matrix* $[\lambda_{ij}]$ *is irreducible and at least one of* P_{ii} *is aperiodic, then the model has a stationary joint probability distribution*

$$\mathbf{x} = (\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(s)})^T$$

satisfying $\mathbf{x} = Q\mathbf{x}$*. Moreover, we have*

$$\lim_{n\to\infty} \mathbf{x}_n = \mathbf{x}.$$

7.2.1 Estimations of Model Parameters

In this subsection we propose some methods to estimate $P^{(jk)}$ and λ_{jk}. For each data sequence, we estimate the transition probability matrix by the following method. Given the data sequence, we count the transition frequency from the states in the kth sequence to the states in the jth sequence. Hence one can construct the transition frequency matrix for the data sequence. After making a normalization, the estimates of the transition probability matrices can also be obtained. We note that one has to estimate s^2 $m \times m$ transition frequency matrices for the multivariate Markov chain model. More precisely, we count the transition frequency $f_{i_j i_k}^{(jk)}$ from the state i_k in the sequence $\{x_n^{(k)}\}$ to the state i_j in the sequence $\{x_n^{(j)}\}$, and therefore the transition frequency matrix for the sequences can be constructed as follows:

$$F^{(jk)} = \begin{pmatrix} f_{11}^{(jk)} & \cdots\cdots & f_{m1}^{(jk)} \\ f_{12}^{(jk)} & \cdots\cdots & f_{m2}^{(jk)} \\ \vdots & \vdots \quad \vdots & \vdots \\ f_{1m}^{(jk)} & \cdots\cdots & f_{mm}^{(jk)} \end{pmatrix}.$$

From $F^{(jk)}$, we get the estimates for $P^{(jk)}$ as follows:

$$\hat{P}^{(jk)} = \begin{pmatrix} \hat{p}_{11}^{(jk)} & \cdots\cdots & \hat{p}_{m1}^{(jk)} \\ \hat{p}_{12}^{(jk)} & \cdots\cdots & \hat{p}_{m2}^{(jk)} \\ \vdots & \vdots \quad \vdots & \vdots \\ \hat{p}_{1m}^{(jk)} & \cdots\cdots & \hat{p}_{mm}^{(jk)} \end{pmatrix}$$

where

$$\hat{p}_{i_j i_k}^{(jk)} = \begin{cases} \dfrac{f_{i_j i_k}^{(jk)}}{\displaystyle\sum_{i_k=1}^{m} f_{i_j i_k}^{(jk)}} & \text{if } \displaystyle\sum_{i_k=1}^{m} f_{i_j i_k}^{(jk)} \neq 0 \\ \\ 0 & \text{otherwise.} \end{cases}$$

In addition to the estimates of $P^{(jk)}$, one needs to estimate the parameters λ_{jk}. We have seen that the multivariate Markov chain model has a stationary vector $\mathbf{x}$ in Proposition 7.2. The vector $\mathbf{x}$ can be estimated from the sequences by computing the proportion of the occurrence of each state in each of the sequences, denoted by

$$\hat{\mathbf{x}} = (\hat{\mathbf{x}}^{(1)}, \hat{\mathbf{x}}^{(2)}, \ldots, \hat{\mathbf{x}}^{(s)})^T.$$

One would expect that

$$\begin{pmatrix} \lambda_{11}P^{(11)} & \lambda_{12}P^{(12)} & \cdots & \lambda_{1s}P^{(1s)} \\ \lambda_{21}P^{(21)} & \lambda_{22}P^{(22)} & \cdots & \lambda_{2s}P^{(2s)} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{s1}P^{(s1)} & \lambda_{s2}P^{(s2)} & \cdots & \lambda_{ss}P^{(ss)} \end{pmatrix} \hat{\mathbf{x}} \approx \hat{\mathbf{x}}. \tag{7.4}$$

From (7.4), it suggests one possible way to estimate the parameters $\lambda = \{\lambda_{jk}\}$ as follows. In fact, by using $||.||_\infty$ as the vector norm for measuring the difference in (7.4), one may consider solving the following minimization problem:

$$\begin{cases} \qquad \min_{\lambda} \left\{ \max_{i} \left| \left[\sum_{k=1}^{m} \lambda_{jk} \hat{P}^{(jk)} \hat{\mathbf{x}}^{(k)} - \hat{\mathbf{x}}^{(j)} \right]_i \right| \right\} \\ \text{subject to} \\ \qquad \sum_{k=1}^{s} \lambda_{jk} = 1, \\ \text{and} \\ \qquad \lambda_{jk} \geq 0, \quad \forall k. \end{cases} \tag{7.5}$$

Problem (7.5) can be formulated as s linear programming problems as follows, see for instance [76].

For each j:

$$\begin{cases} \qquad \min_{\lambda} w_j \\ \text{subject to} \\ \qquad \begin{pmatrix} w_j \\ w_j \\ \vdots \\ w_j \end{pmatrix} \geq \hat{\mathbf{x}}^{(j)} - B \begin{pmatrix} \lambda_{j1} \\ \lambda_{j2} \\ \vdots \\ \lambda_{js} \end{pmatrix}, \\ \qquad \begin{pmatrix} w_j \\ w_j \\ \vdots \\ w_j \end{pmatrix} \geq -\hat{\mathbf{x}}^{(j)} + B \begin{pmatrix} \lambda_{j1} \\ \lambda_{j2} \\ \vdots \\ \lambda_{js} \end{pmatrix}, \\ \qquad w_j \geq 0, \\ \qquad \sum_{k=1}^{s} \lambda_{jk} = 1, \quad \lambda_{jk} \geq 0, \quad \forall k, \end{cases}$$

where

$$B = [\hat{P}^{(j1)}\hat{\mathbf{x}}^{(1)} \mid \hat{P}^{(j2)}\hat{\mathbf{x}}^{(2)} \mid \cdots \mid \hat{P}^{(js)}\hat{\mathbf{x}}^{(s)}].$$

In the next subsection, we give an example to demonstrate the construction of a multivariate Markov chain model from two data sequences.

7.2.2 An Example

Consider the following two categorical data sequences:

$$S_1 = \{4, 3, 1, 3, 4, 4, 3, 3, 1, 2, 3, 4\}$$

and

$$S_2 = \{1, 2, 3, 4, 1, 4, 4, 3, 3, 1, 3, 1\}.$$

By counting the transition frequencies

$$S_1 : 4 \to 3 \to 1 \to 3 \to 4 \to 4 \to 3 \to 3 \to 1 \to 2 \to 3 \to 4$$

and

$$S_2 : 1 \to 2 \to 3 \to 4 \to 1 \to 4 \to 4 \to 3 \to 3 \to 1 \to 3 \to 1$$

we have

$$F^{(11)} = \begin{pmatrix} 0&0&2&0 \\ 1&0&0&0 \\ 1&1&1&2 \\ 0&0&2&1 \end{pmatrix} \quad \text{and} \quad F^{(22)} = \begin{pmatrix} 0&0&2&1 \\ 1&0&0&0 \\ 1&1&1&1 \\ 1&0&1&1 \end{pmatrix}.$$

Moreover by counting the inter-transition frequencies

$$\begin{array}{l*{23}{c}}
S_1: & 4 & & 3 & & 1 & & 3 & & 4 & & 4 & & 3 & & 3 & & 1 & & 2 & & 3 & & 4 \\
& & \nearrow & & \nearrow & & \nearrow & & \nearrow & & \nearrow & & \nearrow & & \nearrow & & \nearrow & & \nearrow & & \nearrow & & \nearrow & \\
S_2: & 1 & & 2 & & 3 & & 4 & & 1 & & 4 & & 4 & & 3 & & 3 & & 1 & & 3 & & 1
\end{array}$$

and

$$\begin{array}{l*{23}{c}}
S_1: & 4 & & 3 & & 1 & & 3 & & 4 & & 4 & & 3 & & 3 & & 1 & & 2 & & 3 & & 4 \\
& & \searrow & & \searrow & & \searrow & & \searrow & & \searrow & & \searrow & & \searrow & & \searrow & & \searrow & & \searrow & & \searrow & \\
S_2: & 1 & & 2 & & 3 & & 4 & & 1 & & 4 & & 4 & & 3 & & 3 & & 1 & & 3 & & 1
\end{array}$$

we have

$$F^{(21)} = \begin{pmatrix} 1&0&2&0 \\ 0&0&0&1 \\ 0&1&3&0 \\ 1&0&0&2 \end{pmatrix}, \qquad F^{(12)} = \begin{pmatrix} 0&1&1&0 \\ 0&0&1&0 \\ 2&0&1&2 \\ 1&0&1&1 \end{pmatrix}.$$

After making a normalization, we have the transition probability matrices:

$$\hat{P}^{(11)} = \begin{pmatrix} 0 & 0 & \frac{2}{5} & 0 \\ \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 1 & \frac{1}{5} & \frac{2}{3} \\ 0 & 0 & \frac{2}{5} & \frac{1}{3} \end{pmatrix}, \qquad \hat{P}^{(12)} = \begin{pmatrix} 0 & 1 & \frac{1}{4} & 0 \\ 0 & 0 & \frac{1}{4} & 0 \\ \frac{2}{3} & 0 & \frac{1}{4} & \frac{2}{3} \\ \frac{1}{3} & 0 & \frac{1}{4} & \frac{1}{3} \end{pmatrix},$$

$$\hat{P}^{(21)} = \begin{pmatrix} \frac{1}{2} & 0 & \frac{2}{5} & 0 \\ 0 & 0 & 0 & \frac{1}{3} \\ 0 & 1 & \frac{3}{5} & 0 \\ \frac{1}{2} & 0 & 0 & \frac{2}{3} \end{pmatrix}, \qquad \hat{P}^{(22)} = \begin{pmatrix} 0 & 0 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & 1 & \frac{1}{4} & \frac{1}{3} \\ \frac{1}{3} & 0 & \frac{1}{4} & \frac{1}{3} \end{pmatrix}.$$

Moreover we also have

$$\hat{\mathbf{x}}_1 = (\frac{1}{6}, \frac{1}{12}, \frac{5}{12}, \frac{1}{3})^T \quad \text{and} \quad \hat{\mathbf{x}}_2 = (\frac{1}{3}, \frac{1}{12}, \frac{1}{3}, \frac{1}{4})^T$$

By solving the corresponding linear programming problems (the optimal solution is not unique in this example), one of the possible multivariate Markov chain models for the two categorical data sequences S_1 and S_2 is given by

$$\begin{cases} \mathbf{x}_{n+1}^{(1)} = 0.5000\hat{P}^{(11)}\mathbf{x}_n^{(1)} + 0.5000\hat{P}^{(12)}\mathbf{x}_n^{(2)} \\ \mathbf{x}_{n+1}^{(2)} = 0.8858\hat{P}^{(21)}\mathbf{x}_n^{(1)} + 0.1142\hat{P}^{(22)}\mathbf{x}_n^{(2)}. \end{cases}$$

7.3 Applications to Multi-product Demand Estimation

Let us consider demand estimation problems as discussed in Sect. 6.3.1. We study customer's sales demand of five important products of a company in a particular year. The sales demand sequences are generated by the same customer and therefore we expect that they should be correlated to each other. Therefore by exploring these relationships, one can develop the multivariate Markov chain model for such demand sequences, and hence obtain better prediction rules.

We first estimate all the transition probability matrices $P^{(ij)}$ by using the method proposed in Sect. 7.2 and we also have the estimates of the state distribution of the five products:

$$\begin{cases} \hat{\mathbf{x}}_1 = (0.0818, 0.4052, 0.0483, 0.0335, 0.0037, 0.4275)^T \\ \hat{\mathbf{x}}_2 = (0.3680, 0.1970, 0.0335, 0.0000, 0.0037, 0.3978)^T \\ \hat{\mathbf{x}}_3 = (0.1450, 0.2045, 0.0186, 0.0000, 0.0037, 0.6283)^T \\ \hat{\mathbf{x}}_4 = (0.0000, 0.3569, 0.1338, 0.1896, 0.0632, 0.2565)^T \\ \hat{\mathbf{x}}_5 = (0.0000, 0.3569, 0.1227, 0.2268, 0.0520, 0.2416)^T \end{cases}$$

By solving the corresponding minimization problems through linear programming we obtain the optimal solution:

$$\Lambda = [\lambda_{jk}] = \begin{pmatrix} 0.0000 & 1.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 1.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 1.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.4741 & 0.5259 \\ 0.0000 & 0.0000 & 0.0000 & 1.0000 & 0.0000 \end{pmatrix}$$

and the multivariate Markov chain model for these five sequences is as follows:

$$\begin{cases} \mathbf{x_{n+1}^{(1)}} = P^{(12)}\mathbf{x_n^{(2)}} \\ \mathbf{x_{n+1}^{(2)}} = P^{(22)}\mathbf{x_n^{(2)}} \\ \mathbf{x_{n+1}^{(3)}} = P^{(35)}\mathbf{x_n^{(5)}} \\ \mathbf{x_{n+1}^{(4)}} = 0.4741P^{(44)}\mathbf{x_n^{(4)}} + 0.5259P^{(45)}\mathbf{x_n^{(5)}} \\ \mathbf{x_{n+1}^{(5)}} = P^{(54)}\mathbf{x_n^{(4)}} \end{cases}$$

where

$$P^{(12)} = \begin{pmatrix} 0.0707 & 0.1509 & 0.0000 & 0.2000 & 0.0000 & 0.0660 \\ 0.4343 & 0.4528 & 0.4444 & 0.2000 & 1.0000 & 0.3491 \\ 0.0101 & 0.1321 & 0.2222 & 0.2000 & 0.0000 & 0.0283 \\ 0.0101 & 0.0943 & 0.2222 & 0.2000 & 0.0000 & 0.0094 \\ 0.0000 & 0.0000 & 0.2000 & 0.0000 & 0.0000 & 0.0094 \\ 0.4747 & 0.1698 & 0.1111 & 0.2000 & 0.0000 & 0.5377 \end{pmatrix}$$

$$P^{(22)} = \begin{pmatrix} 0.4040 & 0.2075 & 0.0000 & 0.2000 & 1.0000 & 0.4340 \\ 0.1111 & 0.4717 & 0.3333 & 0.2000 & 0.0000 & 0.1321 \\ 0.0202 & 0.0566 & 0.3333 & 0.2000 & 0.0000 & 0.0094 \\ 0.0000 & 0.0000 & 0.0000 & 0.2000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.1111 & 0.2000 & 0.0000 & 0.0000 \\ 0.4646 & 0.2642 & 0.2222 & 0.2000 & 0.0000 & 0.4245 \end{pmatrix}$$

$$P^{(35)} = \begin{pmatrix} 0.2000 & 0.0947 & 0.1515 & 0.1639 & 0.0714 & 0.2154 \\ 0.2000 & 0.1895 & 0.2727 & 0.2295 & 0.1429 & 0.1846 \\ 0.2000 & 0.0421 & 0.0000 & 0.0000 & 0.0000 & 0.0154 \\ 0.2000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.2000 & 0.0105 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.2000 & 0.6632 & 0.5758 & 0.6066 & 0.7857 & 0.5846 \end{pmatrix}$$

$$P^{(44)} = \begin{pmatrix} 0.2000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.2000 & 0.4947 & 0.1389 & 0.0196 & 0.0588 & 0.6087 \\ 0.2000 & 0.0842 & 0.3056 & 0.1765 & 0.0588 & 0.1014 \\ 0.2000 & 0.0000 & 0.3056 & 0.5686 & 0.5294 & 0.0290 \\ 0.2000 & 0.0105 & 0.0556 & 0.1569 & 0.3529 & 0.0000 \\ 0.2000 & 0.4105 & 0.1944 & 0.0784 & 0.0000 & 0.2609 \end{pmatrix}$$

$$P^{(45)} = \begin{pmatrix} 0.2000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.2000 & 0.4737 & 0.2121 & 0.0328 & 0.0000 & 0.6462 \\ 0.2000 & 0.1053 & 0.2121 & 0.1967 & 0.0714 & 0.0923 \\ 0.2000 & 0.0000 & 0.2424 & 0.5410 & 0.5714 & 0.0308 \\ 0.2000 & 0.0105 & 0.0303 & 0.1803 & 0.2857 & 0.0000 \\ 0.2000 & 0.4105 & 0.3030 & 0.0492 & 0.0714 & 0.2308 \end{pmatrix}$$

$$P^{(54)} = \begin{pmatrix} 0.2000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.2000 & 0.4842 & 0.1667 & 0.0196 & 0.0588 & 0.6087 \\ 0.2000 & 0.1053 & 0.1667 & 0.1569 & 0.0588 & 0.1159 \\ 0.2000 & 0.0000 & 0.4444 & 0.6275 & 0.6471 & 0.0290 \\ 0.2000 & 0.0105 & 0.0278 & 0.1569 & 0.2353 & 0.0000 \\ 0.2000 & 0.4000 & 0.1944 & 0.0392 & 0.0000 & 0.2464 \end{pmatrix}.$$

According to the multivariate Markov chain model, Products A and B are closely related. In particular, the sales demand of Product A depends strongly on Product B. The main reason is that the chemical composition of Products A and B is the same, but they have different packaging for marketing purposes. Moreover, Products C, D and E are closely related. Similarly, products C and E have the same product flavor, but different packaging. It is interesting to note that even through Products D and E have different chemical nature but similar flavor, the results show that their sales demand are also closely related.

Next we use the multivariate Markov chain model, to make predictions on the state $\hat{\mathbf{x}}_t$ at time t which can be taken as the state with the maximum probability, i.e.,

$$\hat{\mathbf{x}}_t = j, \quad \text{if } [\hat{\mathbf{x}}_t]_i \le [\hat{\mathbf{x}}_t]_j, \forall 1 \le i \le m.$$

To evaluate the performance and effectiveness of our multivariate Markov chain model, a prediction result is measured by the prediction accuracy r defined as

$$r = \frac{1}{T} \times \sum_{t=n+1}^{T} \delta_t \times 100\,\%,$$

where T is the length of the data sequence and

$$\delta_t = \begin{cases} 1, & \text{if } \hat{\mathbf{x}}_t = \mathbf{x}_t \\ 0, & \text{otherwise.} \end{cases}$$

Table 7.1 Prediction accuracy in the sales demand data

	Product A	Product B	Product C	Product D	Product E
First-order Markov chain	46 %	45 %	63 %	51 %	53 %
Multivariate Markov chain	50 %	45 %	63 %	52 %	55 %

For the sake of comparison, we also give the results for the first-order Markov chain model of individual sales demand sequence. The results are reported in Table 7.1. There is noticeable improvement in prediction accuracy in Product A while improvements are also observed in Product D and Product E. The results show the effectiveness of our multivariate Markov chain model.

7.4 Applications to Credit Ratings Models

Credit risk analysis has long been an important topic in banking and finance. Its importance has been greatly emphasized during the past Global Financial Crisis (GFC) initiated by the subprime mortgage crisis in the United States. Different models have been proposed in the literature for evaluating the creditworthiness and likelihood of defaults of corporations, sovereigns and borrowers. There are two strands of literature, namely, the structural firm value model pioneered by Black and Scholes [19] and Merton [158] and the (modern) reduced-form approach developed in Jarrow and Turbull [125]. In the structural approach, the asset value of a firm is assumed to follow a geometric Brownian motion. Default occurs when the asset value of the firm falls below a barrier level, namely, the default barrier. This barrier level can be determined either endogenously or exogenously. In the reduced form approach, defaults are exogenous events and their occurrences are described by random point processes with certain intensity functions.

Credit ratings contain a rich source of information for evaluating creditworthiness and the likelihood of default of corporations, sovereigns and borrowers. There are two types of credit ratings, publicly available ratings and internal ratings. Publicly available ratings are produced and regularly revised by ratings agencies. Some major international ratings agencies include Standard & Poors, and Moodys' and Fitch. Internal ratings are produced by firms for the purposes of internal reference and are not publicly accessible.

Markov chain models provide a natural mathematical tool to describe the stochastic evolution of credit ratings over time. Indeed, ratings transition matrices presented by ratings agencies, such as Standard & Poors, come from the transition probability matrices of Markov chains. However, it appears that many credit rating models used in practice are based on univariate Markov chains. Consequently, the interdependency of credit ratings of corporations from the same industry or related industries is largely overlooked. The failure to incorporate this interdependency relationship may lead to underestimation of risk, and this has a serious economic consequence.

In this section, we shall discuss a multivariate Markov chain model for describing the interdependency of the credit ratings of several corporations from the same industry or similar industries. The multivariate Markov chain is presented in Sect. 7.2. In practice, it is often the case that a long credit rating data series is not easy to obtain, so the scarcity of credit ratings data should be taken into account when one wishes to estimate credit rating models. Here we adopt an approach based on actuarial credibility theory to handle the scarcity of credit rating data. The key idea of the actuarial credibility theory is to provide a method to combine two sources of information for estimating the premiums of insurance policies. The monograph of [31] provides a detailed discussion for credibility and many important models in credibility.

In our present context, we apply actuarial credibility theory to estimate the credit transition matrix in the form of a linear combination of an empirical credit transition matrix and a prior credit transition matrix. Here the prior matrix is assigned based on expert opinion. For example, it may be assigned based on credit transition matrices produced by ratings agencies. The empirical matrix is evaluated based on the observed frequencies of ratings transitions using historical ratings data. This section is based on the materials in [187].

7.4.1 The Credit Transition Matrix

In this subsection we consider an estimate of a credit transition matrix which is a linear combination of a prior credit transition matrix and an empirical credit transition matrix, where the empirical matrix is calculated using the frequencies of ratings transitions, (see Sect. 7.3). Then Proposition 7.1 tells us that there exists a vector X with a stationary probability distribution. Using this result, we can estimate the unknown parameters based on the stationary distributions for the credit ratings. In what follows, we present the mathematical formulation of the estimation method based on credibility theory and discuss how the estimation problem can be formulated in a linear programming problem.

Let $Q^{(jk)}$ be the prior credit transition matrix. The empirical estimate $\hat{P}^{(jk)}$ of the credit transition matrix can be obtained using the method in Sect. 7.2.1. We assign the prior matrix as the transition matrix produced by Standard & Poor's, which is widely used as a benchmark for credit risk analysis.

For ease of illustration, we focus on the case of a single common prior transition matrix for the credit ratings of all, possibly correlated, reference entries in a credit portfolio. In general, one considers several different prior matrices for different reference entities to reflect the prior belief that these reference entities are heterogeneous in terms of their creditworthiness.

We now consider the following estimate $P_e^{(jk)}$ of the credit transition probability $P^{(jk)}$ based on a linear combination of the prior matrix and the empirical matrix:

$$P_e^{(jk)} = w_{jk} Q^{(jk)} + (1 - w_{jk}) \hat{P}^{(jk)}, \quad j,k = 1,2,\ldots,n, \tag{7.6}$$

where $0 \le w_{jk} \le 1$, for each $j,k = 1,2,\ldots,n$.

From proposition 7.1,

$$\begin{pmatrix} \lambda_{11}P_e^{(11)} & \lambda_{12}P_e^{(12)} & \cdots & \lambda_{1n}P_e^{(1n)} \\ \lambda_{21}P_e^{(21)} & \lambda_{22}P_e^{(22)} & \cdots & \lambda_{2n}P_e^{(2n)} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{n1}P_e^{(n1)} & \lambda_{n2}P_e^{(n2)} & \cdots & \lambda_{nn}P_e^{(nn)} \end{pmatrix} \hat{\mathbf{x}} \approx \hat{\mathbf{x}}. \tag{7.7}$$

Let

$$\tilde{\lambda}^1_{jk} = \lambda_{jk} w_{jk}$$

and

$$\tilde{\lambda}^2_{jk} = \lambda_{jk}(1 - w_{jk}).$$

Then, it is easy to check that for each $j, k = 1, 2, \ldots, n$,

$$\tilde{\lambda}^1_{jk} + \tilde{\lambda}^2_{jk} = \lambda_{jk}.$$

Note that the estimation of λ_{jk} and w_{jk} is equivalent to the estimation of $\tilde{\lambda}^1_{jk}$ and $\tilde{\lambda}^2_{jk}$. Consequently, (7.7) can be written in the following form:

$$\begin{pmatrix} \tilde{\lambda}^1_{11}Q^{(11)} + \tilde{\lambda}^2_{11}\hat{P}^{(11)} & \cdots & \tilde{\lambda}^1_{1n}Q^{(1n)} + \tilde{\lambda}^2_{1n}\hat{P}^{(1n)} \\ \tilde{\lambda}^1_{21}Q^{(21)} + \tilde{\lambda}^2_{21}\hat{P}^{(21)} & \cdots & \tilde{\lambda}^1_{2n}Q^{(2n)} + \tilde{\lambda}^2_{2n}\hat{P}^{(2n)} \\ \vdots & \vdots & \vdots \\ \tilde{\lambda}^1_{n1}Q^{(n1)} + \tilde{\lambda}^2_{n1}\hat{P}^{(n1)} & \cdots & \tilde{\lambda}^1_{nn}Q^{(nn)} + \tilde{\lambda}^2_{nn}\hat{P}^{(nn)} \end{pmatrix} \hat{\mathbf{X}} \approx \hat{\mathbf{X}}. \tag{7.8}$$

We now formulate the estimation problem of the transition probability as follows:

$$\begin{cases} \min_{\tilde{\lambda}^1, \tilde{\lambda}^2} \max_i \left\{ \left| \left[\sum_{k=1}^m (\tilde{\lambda}^1_{jk} Q^{(jk)} + \tilde{\lambda}^2_{jk} \hat{P}^{(jk)}) \hat{X}^{(k)} - \hat{X}^{(j)} \right]_i \right| \right\} \\ \text{subject to} \\ \sum_{k=1}^n (\tilde{\lambda}^1_{jk} + \tilde{\lambda}^2_{jk}) = 1, \quad \tilde{\lambda}^1_{jk} \geq 0 \\ \text{and} \\ \tilde{\lambda}^2_{jk} \geq 0, \quad \forall j, k. \end{cases} \tag{7.9}$$

Let

$$O_j = \max_i \left| \left[\sum_{k=1}^m (\tilde{\lambda}^1_{jk} Q^{(jk)} + \tilde{\lambda}^2_{jk} \hat{P}^{(jk)}) \hat{\mathbf{x}}^{(k)} - \hat{\mathbf{x}}^{(j)} \right]_i \right|.$$

Then Problem (7.9) can be formulated in a set of n linear programming problems as in Sect. 7.1. Furthermore it is obvious that one can select the vector norm $||.||_1$

instead of the vector norm $||.||_\infty$. In this case, the resulting problem can also be formulated as a linear programming problem. For the numerical results of the linear programming problems for estimating credit transition matrices, we refer readers to Siu et al. [187].

7.5 Extension to a Higher-Order Multivariate Markov Chain

In this section, we present a higher-order multivariate Markov chain model [54] for modeling multiple categorical sequences based on the models in Sects. 6.2 and 7.2. Here we assume that there are s categorical sequences with order n and each has m possible states in $\mathcal{M}$. In the extended model, we assume that the state probability distribution of the jth sequence at time $t = r + 1$ depends on the state probability distribution of all the sequences (including itself) at times $t = r, r-1, \ldots, r-n+1$. Using the same notation as in the previous two subsections, our proposed higher-order (nth-order) multivariate Markov chain model takes the following form [54]:

$$\mathbf{x}_{r+1}^{(j)} = \sum_{k=1}^{s}\sum_{h=1}^{n} \lambda_{jk}^{(h)} P_h^{(jk)} \mathbf{x}_{r-h+1}^{(k)}, \quad j = 1, 2, \ldots, s \tag{7.10}$$

where

$$\lambda_{jk}^{(h)} \geq 0, \quad 1 \leq j, k \leq s, \quad 1 \leq h \leq n \tag{7.11}$$

and

$$\sum_{k=1}^{s}\sum_{h=1}^{n} \lambda_{jk}^{(h)} = 1, \quad j = 1, 2, \ldots, s.$$

The probability distribution of the jth sequence at time $t = r + 1$ depends on the weighted average of $P_h^{(jk)} \mathbf{x}_{r-h+1}^{(k)}$. Here $P_h^{(jk)}$ is the h-step transition probability matrix which describes the h-step transition from the states in the k-th sequence at time $t = r - h + 1$ to the states in the j-th sequence at time $t = r + 1$ and $\lambda_{jk}^{(h)}$ is the weighting of this term.

From (7.10), if we let

$$\mathbf{X}_r^{(j)} = (\mathbf{x}_r^{(j)}, \mathbf{x}_{r-1}^{(j)}, \ldots, \mathbf{x}_{r-n+1}^{(j)})^T \quad \text{for} \quad j = 1, 2, \ldots, s$$

be the $nm \times 1$ vectors then one can write down the following relation in matrix form:

$$\mathbf{X}_{r+1} \equiv \begin{pmatrix} \mathbf{X}_{r+1}^{(1)} \\ \mathbf{X}_{r+1}^{(2)} \\ \vdots \\ \mathbf{X}_{r+1}^{(s)} \end{pmatrix} = \begin{pmatrix} B^{(11)} & B^{(12)} & \cdots & B^{(1s)} \\ B^{(21)} & B^{(22)} & \cdots & B^{(2s)} \\ \vdots & \vdots & \vdots & \vdots \\ B^{(s1)} & B^{(s2)} & \cdots & B^{(ss)} \end{pmatrix} \begin{pmatrix} \mathbf{X}_r^{(1)} \\ \mathbf{X}_r^{(2)} \\ \vdots \\ \mathbf{X}_r^{(s)} \end{pmatrix} \equiv Q\mathbf{X}_r$$

where

$$B^{(ii)} = \begin{pmatrix} \lambda_{ii}^{(n)} P_n^{(ii)} & \lambda_{ii}^{(n-1)} P_{n-1}^{(ii)} & \cdots & \lambda_{ii}^{(2)} P_2^{(ii)} & \lambda_{ii}^{(1)} P_1^{(ii)} \\ I & 0 & \cdots & 0 & 0 \\ 0 & I & \cdots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & I & 0 \end{pmatrix}_{mn \times mn}$$

and if $i \neq j$ then

$$B^{(ij)} = \begin{pmatrix} \lambda_{ij}^{(n)} P_n^{(ij)} & \lambda_{ij}^{(n-1)} P_{n-1}^{(ij)} & \cdots & \lambda_{ij}^{(2)} P_2^{(ij)} & \lambda_{ij}^{(1)} P_1^{(ij)} \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 0 & 0 \end{pmatrix}_{mn \times mn} .$$

We note that each column sum of Q is not necessarily equal to one but each column sum of $P_h^{(jk)}$ is equal to one. We have the following propositions.

Proposition 7.4. *(Ching et al. [54]) If $\lambda_{jk}^{(h)} > 0$ for $1 \leq j,k \leq s$ and $1 \leq h \leq n$, then the matrix Q has an eigenvalue equal to one and the eigenvalues of Q have modulus less than or equal to one.*

Proposition 7.5. *(Ching et al. [54]) Suppose that the $P_h^{(jk)}$ $(1 \leq j,k \leq s,\ 1 \leq h \leq n)$ are irreducible and $\lambda_{jk}^{(h)} > 0$ for $1 \leq j,k \leq s$ and $1 \leq h \leq n$. Then there is a vector*

$$\mathbf{X} = (\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots, \mathbf{X}^{(s)})^T$$

with

$$\mathbf{X}^{(j)} = (\mathbf{x}^{(j)}, \mathbf{x}^{(j)}, \ldots, \mathbf{x}^{(j)})^T,$$

such that

$$\mathbf{X} = Q\mathbf{X} \quad \text{and} \quad \mathbf{1}\mathbf{x}^{(j)} = 1, \quad \text{for} \quad 1 \leq j \leq s$$

$\mathbf{1} = (1, 1, \ldots, 1)$ *of length m.*

The transition probabilities P_{jk}^{h} can be estimated by counting the transition frequency as described in Sect. 6.2 of Chap. 6. Moreover, we note that $\mathbf{X}$ is not a probability distribution vector, but $\mathbf{x}^{(j)}$ is a probability distribution vector. The above proposition suggests one possible way to estimate the model parameters $\lambda_{ij}^{(h)}$. The key idea is to find $\lambda_{ij}^{(h)}$ which minimizes $||Q\hat{\mathbf{x}} - \hat{\mathbf{x}}||$ under a certain vector norm $||\cdot||$. The estimation method is similar to those in Chap. 6. The proofs of Propositions 7.4 and 7.5 and detailed examples with an application in production planning can be found in Ching et al. [54]. We remark that in Proposition 7.5, actually we don't need to assume all $P_h^{(jk)}$ to be irreducible and the condition that all $\lambda_{jk}^{(h)}$ are positive can be relaxed. In fact, to achieve the same result, one just needs some $P_h^{(jk)}$ to be irreducible and some $\lambda_{jk}^{(h)}$ to be positive, see for instance [216].

7.6 An Improved Multivariate Markov Chain and Its Application to Credit Ratings

The models discussed in the previous sections in this chapter admit only non-negative weighting parameters λ. In other words, only a non-negative association is incorporated in the models presented in the previous sections. However, in many practical situations it appears that a negative association is also observed. To cater for this practical need, we present an improved model which can also incorporate a negative association in our modeling framework. Application of this improved model for describing dependency of credit ratings of several, possibly correlated, credit entities is also discussed. The theory of this section is based on the work of [55] and application on [69].

Before presenting the improved model, we need to introduce some notation. Let

$$\mathbf{Z}_{t+1} = \frac{1}{m-1}\left(\mathbf{1} - \mathbf{X}_t\right),$$

where $\mathbf{1}$ is the vector of all ones; the factor $(m-1)^{-1}$ is the normalization constant; the number of possible states $m \geq 2$.

There is a negative association between $\mathbf{Z}_{t+1}$ and $\mathbf{X}_t$. If $\mathbf{X}_t$ increases (decreases), $\mathbf{Z}_{t+1}$ decreases (increases). To reduce the number of model parameters, we assume that $\mathbf{P}^{(ij)} = \mathbf{I}$ when $i \neq j$, where $\mathbf{I}$ is an $(m \times m)$ identity matrix.

Under the improved model, the evolution of the probability mass functions or probability vectors of ratings of credit entities over time is governed by the following matrix equation:

$$\begin{pmatrix} \mathbf{X}_{t+1}^{(1)} \\ \mathbf{X}_{t+1}^{(2)} \\ \vdots \\ \mathbf{X}_{t+1}^{(s)} \end{pmatrix} = \underbrace{\Lambda^{+} \begin{pmatrix} \mathbf{X}_{t}^{(1)} \\ \mathbf{X}_{t}^{(2)} \\ \vdots \\ \mathbf{X}_{t}^{(s)} \end{pmatrix}}_{Positive\ associated\ part} + \frac{1}{m-1} \underbrace{\Lambda^{-} \begin{pmatrix} \mathbf{1}-\mathbf{X}_{t}^{(1)} \\ \mathbf{1}-\mathbf{X}_{t}^{(2)} \\ \vdots \\ \mathbf{1}-\mathbf{X}_{t}^{(s)} \end{pmatrix}}_{Negative\ associated\ part} \tag{7.12}$$

where

$$\boldsymbol{\Lambda}^{+} = \begin{pmatrix} \lambda_{1,1}\mathbf{P}^{(11)} & \lambda_{1,2}\mathbf{I} & \cdots & \lambda_{1,s}\mathbf{I} \\ \lambda_{2,1}\mathbf{I} & \lambda_{2,2}\mathbf{P}^{(22)} & \cdots & \lambda_{2,s}\mathbf{I} \\ \cdots & \cdots & \cdots & \cdots \\ \lambda_{s,1}\mathbf{I} & \cdots & \lambda_{s,s-1}\mathbf{I} & \lambda_{s,s}\mathbf{P}^{(ss)} \end{pmatrix},$$

and

$$\boldsymbol{\Lambda}^- = \begin{pmatrix} \lambda_{1,-1}\mathbf{P}^{(11)} & \lambda_{1,-2}\mathbf{I} & \cdots & \lambda_{1,-s}\mathbf{I} \\ \lambda_{2,-1}\mathbf{I} & \lambda_{2,-2}\mathbf{P}^{(22)} & \cdots & \lambda_{2,-s}\mathbf{I} \\ \cdots & \cdots & \cdots & \cdots \\ \lambda_{s,-1}\mathbf{I} & \cdots & \lambda_{s,-s+1}\mathbf{I} & \lambda_{s,-s}\mathbf{P}^{(ss)} \end{pmatrix}.$$

Here $\lambda_{i,j} \geq 0$, for $i = 1, 2, \ldots, s$ and $j = \pm 1, \ldots, \pm s$, and

$$\sum_{j=-s}^{s} \lambda_{i,j} = 1.$$

In Equation (7.12), $\boldsymbol{\Lambda}^+$ and $\boldsymbol{\Lambda}^-$ represent the transition probability matrices for the positive and negative associations, respectively. From the second term of (7.12), an increase (or decrease) in $\mathbf{X}_t$ leads to a decrease (or increase) in $\mathbf{X}_{t+1}$. In other words, there is a negative association between the two probability vectors $\mathbf{X}_t$ and $\mathbf{X}_{t+1}$. With the normalization constant $\frac{1}{m-1}$, $\mathbf{X}_{t+1}^{(j)}$ on the left hand side of (7.12) is a probability vector, for each $j = 1, 2, \ldots, s$.

The number of parameters involved here is $O(sm^2 + s^2)$, where s is the number of credit entities in a portfolio and m is the number of rating categories. So, the number of parameters in the improved model is less than that in the model in [187] (i.e. $O(s^2m^2)$).

The following example illustrates how the reduction of the number of parameters can be achieved in a practical situation. We consider a rating system consisting of 10 rating categories (i.e. $m = 10$). This is the number of rating categories in the most commonly used rating system in practice. Now, we consider a credit portfolio having 30 individual credit entities, say a portfolio of 30 corporate bonds. This is a reasonable proxy for the size of a typical credit portfolio in practice. In this case, the number of parameters in the improved model is 3,900, while the number of parameters in the model considered in [50, 187] is 90,000. So, we have a significant reduction in the number of parameters by considering the improved model.

7.6.1 Convergence Property of the Model

As in the multivariate Markov chain considered in [187], the calibration method can be developed by exploiting the existence of a unique set of stationary probability vectors for ratings of the credit entities. In the sequel, we discuss the existence of the stationary probability vectors under the improved model, the rate of convergence and the method to speed up the rate of convergence. The development below follows that in [55].

Firstly, we write the improved model in terms of the following matrix-valued difference equation:

$$\begin{aligned}
\mathbf{X}_{t+1} &= \begin{pmatrix} \mathbf{X}_{t+1}^{(1)} \\ \mathbf{X}_{t+1}^{(2)} \\ \vdots \\ \mathbf{X}_{t+1}^{(s)} \end{pmatrix} \\
&= \begin{pmatrix} \mathbf{H}_{1,1} & \mathbf{H}_{1,2} & \cdots & \mathbf{H}_{1,s} \\ \mathbf{H}_{2,1} & \mathbf{H}_{2,2} & \cdots & \mathbf{H}_{2,s} \\ \cdots & \cdots & \cdots & \cdots \\ \mathbf{H}_{s,1} & \mathbf{H}_{s,2} & \cdots & \mathbf{H}_{s,s} \end{pmatrix} \begin{pmatrix} \mathbf{X}_{t}^{(1)} \\ \mathbf{X}_{t}^{(2)} \\ \vdots \\ \mathbf{X}_{t}^{(s)} \end{pmatrix} \\
&\quad + \frac{1}{m-1} \begin{pmatrix} \mathbf{J}_{1,-1} & \mathbf{J}_{1,-2} & \cdots & \mathbf{J}_{1,-s} \\ \mathbf{J}_{2,-1} & \mathbf{J}_{2,-2} & \cdots & \mathbf{J}_{2,-s} \\ \cdots & \cdots & \cdots & \cdots \\ \mathbf{J}_{s,-1} & \cdots & \mathbf{J}_{s,-s+1} & \mathbf{J}_{s,-s} \end{pmatrix} \begin{pmatrix} \mathbf{1} \\ \mathbf{1} \\ \vdots \\ \mathbf{1} \end{pmatrix} \\
&\equiv \mathbf{M}_s \mathbf{X}_t + \mathbf{b}. \qquad (7.13)
\end{aligned}$$

Here $\mathbf{M}_s$ is defined as a hypermatrix with elements $\mathbf{H}_{i,j}$; $\mathbf{b}$ is an $(s \times s)$ matrix with elements being all ones;

$$\mathbf{H}_{ij} = \begin{cases} (\lambda_{i,j} - \frac{\lambda_{i,-j}}{m-1})\mathbf{P}^{(ii)} & \text{if } i = j \\ (\lambda_{i,j} - \frac{\lambda_{i,-j}}{m-1})\mathbf{I} & \text{otherwise,} \end{cases}$$

and

$$\mathbf{J}_{ij} = \begin{cases} \lambda_{i,-j}\mathbf{P}^{(ii)} & \text{if } i = j \\ \lambda_{i,-j}\mathbf{I} & \text{otherwise.} \end{cases}$$

We note that

$$\begin{aligned}
\mathbf{X}_{t+1} &= \mathbf{M}_s^2 \mathbf{X}_{t-1} + (\mathbf{I} + \mathbf{M}_s)\mathbf{b} = \mathbf{M}_s^3 \mathbf{X}_{t-2} + (\mathbf{I} + \mathbf{M}_s + \mathbf{M}_s^2)\mathbf{b} \\
&= \mathbf{M}_s^{t+1} \mathbf{X}_0 + \sum_{k=0}^{t} \mathbf{M}_s^k \mathbf{b}, \qquad (7.14)
\end{aligned}$$

where $\mathbf{M}_s^0 = \mathbf{I}$.

Then, from (7.14), there exists a unique set of stationary probability vectors satisfying the matrix-valued difference equation (7.13) if $||\mathbf{M}_s||_\infty < 1$, where $||\mathbf{M}_s||_\infty$ is the supremum matrix norm. If this is the case then,

$$\lim_{t\to\infty} \mathbf{X}_t = \lim_{t\to\infty} \sum_{k=0}^{t} \mathbf{M}_s^k \mathbf{b} = (\mathbf{I} - \mathbf{M}_s)^{-1}\mathbf{b},$$

which is the matrix containing the stationary probability vectors for the ratings of credit entities.

We can also see that the rate of convergence to the set of stationary probability vectors depends on the magnitude of $||\mathbf{M}_s||_\infty$. In particular, the smaller $||\mathbf{M}_s||_\infty$ is, the quicker the convergence will be. Thus by controlling the value of $||\mathbf{M}_s||_\infty$, the convergence can be speeded up.

It has been shown in [55, 74] that

$$||\mathbf{M}_s||_\infty \leq \max_{1\leq k\leq s} \left\{ m \left| \lambda_{k,k} - \frac{\lambda_{k,-k}}{(m-1)} \right| + \sum_{k\neq i} \left| \lambda_{k,i} - \frac{\lambda_{k,-i}}{(m-1)} \right| \right\}.$$

This means that one can control the rate of convergence by imposing an upper bound $\alpha < 1$ and introducing the following additional constraints:

$$m \left| \lambda_{k,k} - \frac{\lambda_{k,-k}}{(m-1)} \right| + \sum_{k\neq i} \left| \lambda_{k,i} - \frac{\lambda_{k,-i}}{(m-1)} \right| \leq \alpha \quad \text{for} \quad i = 1, 2, \ldots, s.$$

Clearly if a smaller value of α is chosen, the rate of convergence to the stationary probability vectors for the ratings of credit entities becomes faster. Thus, reasonably accurate estimates for the unknown parameters can be obtained even when the ratings dataset is short.

7.6.2 Estimation of Model Parameters

In this subsection, we present efficient methods for estimating $\mathbf{P}^{(jj)}$ and λ_{jk}. For each ratings sequence of a credit entity, we adopt the method in [50, 55] to estimate the transition probability matrix. More precisely, given a ratings sequence, we count the transition frequencies from one arbitrary state to the others and construct the transition frequency matrix for the ratings sequence. Then, an estimate of the transition probability matrix of the ratings of the entity can be obtained by normalization. In the improved multivariate Markov chain model for ratings, we need to estimate $O(sm^2)$ transition frequency matrices when there are s credit entities and the number of possible rating categories is m. The stationary probability vectors $\mathbf{X}$ are estimated from the proportion of occurrence of each rating category in each of the ratings sequences.

Based on the idea at the end of the previous section, if we take $||.||$ to be $||.||_\infty$, then an estimate of λ_{jk} can be obtained by solving the following optimization problem (see for instance [50, 55, 74]):

$$\min_{\lambda} \left\{ \max_{i} \left| \left[\mathbf{b}_{j,k} - \hat{\mathbf{X}}^{(j)} \right]_i \right| \right\}$$

subject to

$$\begin{cases} \mathbf{b}_{j,k} = \displaystyle\sum_{k=1}^{s} \left((\lambda_{j,k} - \frac{\lambda_{j,-k}}{m-1}) \Delta_{jk} \hat{\mathbf{X}}^{(k)} + \frac{1}{m-1} \lambda_{j,k} \Delta_{jk} \mathbf{1} \right) \\ \displaystyle\sum_{k=-s}^{s} \lambda_{jk} = 1, \quad \forall j = 1, 2, \ldots, s. \\ \lambda_{jk} \geq 0, \quad \forall k = \pm 1, \ldots, \pm s, \quad j = 1, 2, \ldots, s. \\ m \left| \lambda_{k,k} - \dfrac{\lambda_{k,-k}}{(m-1)} \right| + \displaystyle\sum_{k \neq i} \left| \lambda_{k,i} - \frac{\lambda_{k,-i}}{(m-1)} \right| \leq \alpha \quad \text{for} \quad k = 1, 2, \ldots, s. \end{cases} \tag{7.15}$$

Here,

$$\boldsymbol{\Delta}_{jk} = \begin{cases} \mathbf{P}^{(jj)} & \text{if } j = k \\ \mathbf{I} & \text{if } j \neq k. \end{cases}$$

Indeed, Problem (7.15) can be formulated as a set of s linear programming problems, see for example [76, p. 221]

$$\min_{\lambda} \sum_{i} w_i$$

subject to

$$\begin{cases} w_i \geq \left[\mathbf{b}_{j,k} - \hat{\mathbf{X}}^{(j)} \right]_i \\ w_i \geq - \left[\mathbf{b}_{j,k} - \hat{\mathbf{X}}^{(j)} \right]_i \\ \displaystyle\sum_{k=-s}^{s} \lambda_{jk} = 1, \quad \forall j. \\ \lambda_{jk} \geq 0, \quad \forall k = \pm 1, \ldots, \pm s, \forall j. \\ A_j \Lambda_j \leq \alpha \mathbf{1}, \forall j. \end{cases}$$

where

$$\boldsymbol{\Lambda}_j = (\lambda_{j,1}, \lambda_{j,2}, \ldots, \lambda_{j,s}, \lambda_{j,-1}, \ldots, \lambda_{j,-s})^T, \quad \mathbf{A}_j = [\mathbf{A}_{1j}, \mathbf{A}_{2j}],$$

$$\mathbf{A}_{1j} = \begin{pmatrix} 1 & \cdots & m & \cdots & 1 & 1 \\ 1 & \cdots & m & \cdots & 1 & -1 \\ 1 & \cdots & m & \cdots & -1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -1 & \cdots & -m & \cdots & -1 & 1 \\ -1 & \cdots & \underbrace{-m}_{j\ th\ column} & \cdots & -1 & -1 \end{pmatrix}$$

and

$$\mathbf{A}_{2j} = -\frac{1}{m-1}\mathbf{A}_{1j}.$$

Here $\mathbf{A}_{1j}$ covers all of the rows in which each component takes one of the two possible values, 1 and -1. Hence, $\mathbf{A}$ has $s \times 2^s$ rows. Thus these problems can be solved efficiently and independently. They can be solved easily using speadsheet software such as EXCEL. This provides market practitioners with a handy way to estimate the unknown parameters of the improved multivariate Markov chain model for ratings.

To compare the empirical performance of various models, we adopt the Bayesian Information Criterion (BIC) as a criterion, which is defined as below:

$$BIC := -2L + q \log n.$$

Here

$$L := \sum_{j=1}^{s} \left(\sum_{i_0,k_1,\cdots k_s=1}^{m} n^{(j)}_{i_0,k_1,\cdots,k_s} \log l \right),$$

$$l := \sum_{l=1}^{m} \sum_{k=1}^{s} (\lambda_{jk} - \frac{1}{m-1}\lambda_{j,-k}) p^{(jk)}_{i_0,k_l} + \frac{1}{m-1}\lambda_{j,-k},$$

where

$$n^{(j)}_{i_0,k_1,k_2,\cdots,k_s} = \sum x^{(j)}_{n+1}(i_0) x^1_n(k_1) x^2_n(k_2) \cdots x^s_n(k_s).$$

Note that L and l are the maximum log-likelihood function and likelihood function of the improved model, respectively. Here, q represents the number of independent parameters and n is the length of the ratings sequence. The "best" model is given by the one with the smallest BIC.

7.6.3 Practical Implementation, Accuracy and Computational Efficiency

In this section, we first provide a numerical example to illustrate the practical implementation of the improved model and to compare its computational efficiency with that of the model considered in [187]. Here we consider the following three hypothetical ratings sequences, each of which has three possible rating classes: $1, 2, 3$. In this case, $s = 3$, $m = 3$ and $\mathcal{S} := \{s_1, s_2, s_3\}$.

A :	$1, 2, 2, 1, 3, 3, 2, 3, 1, 3, 3, 2, 2, 3, 2, 1, 2, 1, 2, 2$
B :	$2, 2, 1, 1, 2, 1, 2, 1, 2, 2, 2, 2, 3, 2, 2, 1, 2, 1, 2, 2$
C :	$3, 1, 1, 2, 2, 1, 2, 3, 1, 1, 3, 3, 3, 2, 3, 2, 3, 3, 1, 2$

Table 7.2 The BIC for different models

Models	BIC
The model in (Siu et al. (2005))	607.4
The New Model ($\alpha = 0.1$)	247.0
The New Model ($\alpha = 0.2$)	246.2
The New Model ($\alpha = 0.3$)	245.3
The New Model ($\alpha = 0.4$)	245.5
The New Model ($\alpha = 0.5$)	247.2
The New Model ($\alpha = 0.6$)	246.5
The New Model ($\alpha = 0.7$)	245.9
The New Model ($\alpha = 0.8$)	245.5
The New Model ($\alpha = 0.9$)	245.2
The New Model ($\alpha^* = 1.0$)	244.9

Firstly, we estimate all of the transition probability matrices $\mathbf{P}^{(jj)}$ by counting the frequencies of transitions from one rating category to the others. The estimates of these transition probability matrices are presented as follows:

$$\mathbf{P}^{(11)} = \begin{pmatrix} 0.0000 & 0.3750 & 0.1667 \\ 0.6000 & 0.3750 & 0.5000 \\ 0.4000 & 0.2500 & 0.3333 \end{pmatrix},$$

$$\mathbf{P}^{(22)} = \begin{pmatrix} 0.1667 & 0.4167 & 0.0000 \\ 0.8333 & 0.5000 & 1.0000 \\ 0.0000 & 0.0833 & 0.0000 \end{pmatrix},$$

and

$$\mathbf{P}^{(33)} = \begin{pmatrix} 0.3333 & 0.2000 & 0.3750 \\ 0.5000 & 0.2000 & 0.2500 \\ 0.1667 & 0.6000 & 0.3750 \end{pmatrix}.$$

We suppose that the upper bound α for controlling the rate of convergence ranges from 0.1 to 1.0, with an increment of 0.1. For each value of α, the estimates of the parameters λ_{ij}'s $(i, j = 1, 2, 3)$ can be computed by solving the set of linear programming problems. In particular, we need to solve three linear programming problems since we have three ratings sequences.

As mentioned previously, we adopt the BIC for model selection. We compute the BIC of the improved model with different values of α and the BIC of the model considered in [187]. The results are reported in Table 7.2.

From Table 7.2, the "optimal" model is the improved model with $\alpha^* = 1.0$. This "optimal" model is also better than the model considered in [187] in terms of fitting the hypothetical ratings data. This may be the consequence of the existence of negative associations among the three hypothetical ratings sequences. The optimal value $\alpha^* = 1.0$ is also the most appropriate one to accommodate the given length of the ratings sequences according to the BIC.

When the "optimal" value of $\alpha^* = 0.6$, the estimates of λ_{ij}'s $(i, j = 1, 2, 3)$ are given by:

$$\begin{pmatrix} \lambda_{1,1} & \lambda_{1,2} & \lambda_{1,3} & \lambda_{1,-1} & \lambda_{1,-2} & \lambda_{1,-3} \\ \lambda_{2,1} & \lambda_{2,2} & \lambda_{2,3} & \lambda_{2,-1} & \lambda_{2,-2} & \lambda_{2,-3} \\ \lambda_{3,1} & \lambda_{3,2} & \lambda_{3,3} & \lambda_{3,-1} & \lambda_{3,-2} & \lambda_{3,-3} \end{pmatrix}$$

$$= \begin{pmatrix} 0.2146 & 0.1599 & 0.1101 & 0.3094 & 0.0876 & 0.1184 \\ 0.0000 & 0.4707 & 0.0000 & 0.0000 & 0.3383 & 0.1910 \\ 0.2094 & 0.0000 & 0.0157 & 0.0000 & 0.3246 & 0.4503 \end{pmatrix},$$

$\mathbf{M}_1 =$

$$\left(\begin{array}{ccc|ccc|ccc} 0.0000 & 0.0225 & 0.01000 & 0.1161 & 0.0000 & 0.0000 & 0.0509 & 0.0000 & 0.0000 \\ 0.0359 & 0.0225 & 0.0299 & 0.0000 & 0.1161 & 0.0000 & 0.0000 & 0.0509 & 0.0000 \\ 0.0240 & 0.0150 & 0.0200 & 0.0000 & 0.0000 & 0.1161 & 0.0000 & 0.0000 & 0.0509 \\ \hline 0.0000 & 0.0000 & 0.0000 & 0.0503 & 0.1256 & 0.0000 & -0.0955 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.2512 & 0.1507 & 0.3015 & 0.0000 & -0.0955 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0251 & 0.0000 & 0.0000 & 0.0000 & -0.0955 \\ \hline 0.2094 & 0.0000 & 0.0000 & -0.16230 & 0.0000 & 0.0000 & -0.0698 & -0.0419 & -0.0785 \\ 0.0000 & 0.2094 & 0.0000 & 0.0000 & -0.1623 & 0.0000 & -0.1047 & -0.0419 & -0.0524 \\ 0.0000 & 0.0000 & 0.2094 & 0.0000 & 0.0000 & -0.1623 & -0.0349 & -0.1257 & -0.0785 \end{array}\right),$$

and

$$\mathbf{b}_1 = (0.2868, 0.3312, 0.2551, 0.1942, 0.4902, 0.1096, 0.3668, 0.3762, 0.4193)^T.$$

We remark that the computational time of the model considered in [187] is 0.11 s, while the computation time of the improved model with $\alpha = 1.0$ is 0.125 s. So, in this example, the computational times of the two models are comparable.

7.7 Summary

In this chapter we first discuss a multivariate Markov chain model with estimation methods based on solving a system of linear programming problems. The model is then applied to a multi-product demand estimation problem and to modelling the interdependency of credit ratings. Then we discuss various extensions of the multivariate Markov chain model including a higher-order multivariate Markov chain model and an improved model incorporating a negative association. Application of the improved model for modeling the interdependency in credit ratings and its practical implementation, accuracy, as well as computational efficiency are discussed.

7.8 Exercise

1. Write a computer program to solve the above linear programming problems (7.9).
2. Prove Proposition 7.4.
3. Prove Proposition 7.5.
4. Given two categorical data sequences as follow:

$$S_1 : 1, 2, 2, 3, 2, 2, 1, 2, 1, 2, 3, 1, 2, 1, 2, 3$$

and

$$S_2 : 1, 2, 1, 1, 2, 1, 3, 2, 2, 2, 2, 1, 2, 2, 2, 1.$$

Construct a multivariate Markov chain model discussed in Sect. 7.2 by using the Euclidean norm $||.||_2$ to find the parameters.

Chapter 8
Hidden Markov Chains

8.1 Introduction

Hidden Markov models (HMMs) have been applied to many real-world applications. Usually HMMs only deal with the first-order transition probability distribution among the hidden states, see for instance Sect. 1.4. Moreover, the observable states are affected by the hidden states but not vice versa. In this chapter, we study both higher-order hidden Markov models and interactive HMMs in which the hidden states are directly affected by the observed states. We will also develop estimation methods for the model parameters in both cases.

The remainder of this chapter is organized as follows. In Sect. 8.2, we present a higher-order hidden Markov model. In Sect. 8.3, we discuss a class of double higher-order hidden Markov models. In Sect. 8.4, we discuss an interactive HMM. In Sect. 8.5, the interactive HMM is then applied to modeling portfolio credit risk. Then finally, a summary will be given to conclude this chapter in Sect. 8.6.

8.2 Higher-Order HMMs

In this section, we present a higher-order Hidden Markov Model (HMM) [60]. HMMs have become increasingly popular in the last few decades. Since HMMs are very rich in mathematical structure, they can form the theoretical basis in a wide range of applications such as speech recognition [175] and computer version [33]. A standard HMM is usually characterized by the following elements [175]:

1. N, the number of states in the model. Although the states are hidden, for many practical applications, there are often physical significance to the states. We denote the individual states as

$$\mathbf{S} = \{S_1, S_2, \ldots, S_N\},$$

and the state at the length t as q_t.

W.-K. Ching et al., *Markov Chains*, International Series in Operations Research
& Management Science 189, DOI 10.1007/978-1-4614-6312-2_8,

2. M, the number of distinct observation symbols (or states) for the hidden states. The observation symbols correspond to the physical output of the system being modeled. We denote the individual symbols as
$$\mathbf{V} = \{v_1, v_2, \ldots, v_M\}.$$
3. The state transition probability distribution
$$A = \{a_{ij}\}$$
where
$$a_{ij} = P(q_{t+1} = S_j | q_t = S_i), \quad 1 \le i, j \le N.$$
4. The observation probability distribution in state j, $B = \{b_j(k)\}$, where
$$b_j(k) = P(O_t = v_k | q_t = S_j), \quad 1 \le j \le N, 1 \le k \le M.$$
5. The initial state distribution $\Pi = \{\pi_i\}$ where
$$\pi_i = P(q_1 = S_i), \quad 1 \le i \le N.$$

Given appropriate values of N, M, A, B and Π, the HMM can be used as a generator to give an observation sequence
$$O = O_1 O_2 \ldots O_T$$
where each observation O_t is one of the symbols from $\mathbf{V}$, and T is the number of observations in the sequence. For simplicity, we use the compact notation
$$\Lambda = (A, B, \Pi)$$
to indicate the complete parameter set of the HMM. According to the above specification, very often a first order Markov process is used in modeling the transitions among the hidden states in a HMM. The main difference between the traditional HMM and a higher-order HMM is that in the hidden layer, the state transition probability is governed by the mth order higher-order Markov model
$$a_{i_{t-m+1},\ldots,i_{t+1}} = P(q_{t+1} = S_{i_{t+1}} | q_t = S_{i_t}, \ldots, q_{t-m+1} = S_{i_{t-m+1}}).$$
We assume that the distribution Π of initial m states is given by
$$\pi_{i_1,i_2,\ldots,i_m} = P(q_1 = S_{i_1}, q_2 = S_{i_2}, \ldots, q_m = S_{i_m}).$$

Here we will present solutions to the three problems for higher-order HMMs. Recall that they are practical problems in the traditional HMMs (see Sect. 1.4).

- **Problem 1** Given the observation sequence

  $$O = O_1 O_2 \dots O_T$$

  and a higher-order HMM, how does one efficiently compute the probability of the observation sequence?
- **Problem 2** Given the observation sequence

  $$O = O_1 O_2 \dots O_T$$

  and a higher-order HMM, how does one choose a corresponding state sequence

  $$Q = q_1 q_2 \dots q_T$$

  which is optimal in a certain sense (e.g. in the sense of maximum likelihood)?
- **Problem 3** Given the observation sequence

  $$O = O_1 O_2 \dots O_T$$

  and a higher-order HMM, how does one choose the model parameters?

8.2.1 Problem 1

For Problem 1, we calculate the probability of the observation sequence,

$$O = O_1 O_2 \dots O_T,$$

given the higher-order HMM, i.e., $P[O|\Lambda]$. One possible way of doing this is through enumerating each possible state sequence of length T. However, this calculation is computationally infeasible even for small values of T and N. We apply the forward-backward procedure [11] to calculate this probability of the observation sequence. We define the forward variable

$$\alpha_t(i_{t-m+1}, \dots, i_t)$$

as follows:

$$\alpha_t(i_{t-m+1}, \dots, i_t) = P(O_1, \dots, O_t, q_{t-m+1} = S_{i_{t-m+1}}, \dots, q_t = S_{i_t} | \Lambda),$$

where $m \leq t \leq T$, i.e., the conditional probability that the subsequence of the first t observations and the subsequence of the last m hidden states ending at time t are equal to

$$v_1 \ldots v_t \quad \text{and} \quad S_{i_{t-m+1}} \ldots S_{i_t}$$

respectively, and are given by the model parameters Λ. We see that if we can obtain the values of

$$\alpha_T(i_{T-m+1}, \ \ldots, i_T) \quad \forall \quad i_{T-m+1}, \ldots, i_T,$$

then it is obvious that $P[O|\Lambda]$ can be obtained by summing up all the values of

$$\alpha_T(i_{T-m+1}, \ldots, i_T).$$

It is interesting to note that the values of $\alpha_T(i_{T-m+1}, \ldots, i_T)$ can be obtained by the following recursive equation and the details are given as follows:

(F1) Initialization: $\alpha_m(i_1, i_2, \ldots, i_m) = \pi_{i_1, i_2, \ldots, i_m} \cdot \prod_{j=1}^{m} b_{i_j}(v_j).$

(F2) Recursive Equation: $\alpha_{t+1}(i_{t-m+2}, i_{t-m+3}, \ldots, i_{t+1}) =$

$$\begin{aligned} &\sum_{i_{t-m+1}=1}^{N} \alpha_t(i_{t-m+1}, \ldots, i_t) \cdot P(O_{t+1}|\Lambda, q_{t+1} = S_{i_{t+1}}) \cdot \\ &P(q_{t+1} = S_{i_{t+1}}|\Lambda, q_{t-m+1} = S_{i_{t-m+1}}, \ldots, q_t = S_{i_t})) \\ &= \sum_{i_{t-m+1}=1}^{N} \alpha_t(i_{t-m+1}, \ldots, i_t) \cdot a_{i_{t-m+1} i_t, i_{t+1}} b_{i_{t+1}}(v_{t+1}). \end{aligned}$$

(F3) Termination: $P(O|\Lambda) = \sum_{i_{T-m+1}, \ldots, i_T=1}^{N} \alpha_T(i_{T-m+1}, \ldots, i_T).$

The initiation step and the recursion step calculate the forward probabilities as the joint probability of hidden states and initial observations. The last step gives the desired calculation of $P[O|\Lambda]$ as the sum of the terminal forward variables $\alpha_T(i_{T-m+1}, \ldots, i_T)$.

In a similar manner, a backward variable $\beta_t(i_1, i_2, \ldots, i_m)$ can be defined as follows: $\beta_t(i_1, i_2, \ldots, i_m) =$

$$P(O_{t+m} \ldots O_T | q_t = S_{i_t}, \ldots, q_{t+m-1} = S_{i_{t+m-1}}, \Lambda), \ 0 \leq t \leq T - m.$$

(B1) Initialization: $\beta_{T-t}(i_1, \ldots, i_m) = 1, 0 \leq t \leq m-1, 1 \leq i_1, \ldots, i_m \leq N.$

(B2) Recursive equation: $\beta_t(i_1, i_2, \ldots, i_m) =$

$$\sum_{i_{t+m}=1}^{N} P(O_{t+m+1} \ldots O_T | q_{t+1} = S_{i_{t+1}}, \ldots, q_{t+m-1} = S_{i_{t+m-1}}, q_{t+m} = S_{i_{t+m}}, \Lambda) \cdot$$
$$P(O_{t+m} | q_{t+m} = S_{i_{t+m}}, \Lambda) \cdot P(q_{t+m} = S_{i_{t+m}} | q_t = S_{i_t}, \ldots, q_{t+m-1} = S_{i_{t+m-1}}, \Lambda)$$
$$= \sum_{k=1}^{N} b_k(O_{t+m}) \beta_{t+1}(i_2, \ldots, i_m, k) \cdot a_{i_2,\ldots,i_m,k}.$$

We note that in the backward algorithm, $\beta_{T-t}(i_1, i_2, \ldots, i_m)$ to be 1.

8.2.2 Problem 2

In Problem 2, we attempt to uncover the whole hidden sequence given the observations, i.e., to find the most likely state sequence. In practical situations, we use an optimality criteria to solve this problem. The most widely used criterion to find the "best" sequence is by maximizing $P[Q|\Lambda, O]$. This is equivalent to maximizing $P(Q, O|\Lambda)$. We note that

$$P(Q|\Lambda, O) = \frac{P(Q, O|\Lambda)}{P(O|\Lambda)}.$$

The Viterbi algorithm [204] is a technique for finding "the most likely" hidden sequence

$$Q = \{q_1, q_2, \ldots, q_T\}$$

for a given observation sequence

$$O = \{O_1, O_2, \ldots, O_T\}.$$

Here we need to define the following quantity:

$$\delta_t(i_{t-m+1}, \ldots, i_t) = \max_{q_1,\ldots,q_{t-m}} P(q_1 = S_{i_1}, \ldots, q_t = S_{i_t}, O_1, \ldots, O_t|\Lambda),$$

for $m \le t \le T$ and $\delta_t(i_{t-m+1}, \ldots, i_t)$. It is the best score (highest probability) along a single best state sequence at time t, which accounts for the first t observations and ends in state S_{i_t}. By induction, we have

$$\begin{aligned} &\delta_{t+1}(i_{t-m+2}, \ldots, i_{t+1}) \\ &= \max_{1\le q_{t-m+1}\le N} \{\delta_t(i_{t-m+1}, \ldots, i_t) \cdot a_{i_{t-m+1},\ldots,i_{t+1}}\} \cdot b_{i_{t+1}}(O_{t+1}). \end{aligned} \tag{8.1}$$

To retrieve the state sequence, one needs to keep track of the argument maximized (8.1) for each t and i_{t-m+1}, …, i_t. This can be done via the array $\Delta_{t+1}(i_{t-m+2},\dots,i_{t+1})$. The complete procedure for finding the best state sequence is as follows:

(U1) Initialization:

$$\begin{aligned}\delta_m(i_1,\dots,i_m) &= P(q_1 = S_{i_1},\dots,q_m = S_{i_m}, O_1,\dots,O_m|\Lambda)\\ &= P(q_1 = S_{i_1},\dots,q_m = S_{i_m}|\Lambda)\cdot\prod_{j=1}^{m} P(O_j|\Lambda, q_j = S_{i_j})\\ &= \pi_{i_1,i_2,\dots,i_m}\prod_{j=1}^{m} b_{i_j}(v_j),\quad 1\le i_1,i_2,\dots,i_m\le N.\end{aligned}$$

We also set $\Delta_m(i_1,\dots,i_m) = 0$.

(U2) Recursion:

$$\begin{aligned}&\delta_{t+1}(i_{t-m+2},\dots,i_{t+1})\\ &= \max_{q_1,\dots,q_{t-m+1}} P(q_{t+1} = S_{i_{t+1}}, O_{t+1}|\Lambda, q_1 = i_1,\dots,q_t = i_t, O_1,\dots,O_t)\cdot\\ &\quad P(q_1 = S_{i_1},\dots,q_t = S_{i_t}, O_1,\dots,O_t|\Lambda)\\ &= \max_{1\le q_{t-m+1}\le N}\delta_t(i_{t-m+1},\dots,i_t)\cdot\\ &\quad P(O_{t+1}|\Lambda, q_1 = S_{i_1},\dots,q_{t+1} = S_{i_{t+1}}, O_1,\dots,O_t)\cdot\\ &\quad P(q_{t+1} = S_{i_{t+1}}|\Lambda, q_1 = S_{i_1},\dots,q_t = S_{i_t}, O_1,\dots,O_t)\\ &= \max_{1\le q_{t-m+1}\le N}\delta_t(i_{t-m+1},\dots,i_t)\cdot P(O_{t+1}|\Lambda, q_{t+1} = S_{i_{t+1}})\cdot\\ &\quad P(q_{t+1} = S_{i_{t+1}}|\Lambda, q_{t-m+1} = S_{i_{t-m+1}},\dots,q_t = S_{i_t})\\ &= \max_{1\le q_{t-m+1}\le N}\{\delta_t(i_{t-m+1},\dots,i_t)\cdot a_{i_{t-m+1},\dots,i_{t+1}}\}\cdot b_{i_{t+1}}(v_{t+1}).\end{aligned}$$

For $m+1\le t\le T$ and $1\le i_{t+1}\le N$, we have

$$\begin{aligned}&\Delta_{t+1}(i_{t-m+2},\dots,i_{t+1})\\ &= \text{argmax}_{1\le q_{t-m+1}\le N}\{\delta_t(i_{t-m+1},\dots,i_t)\cdot a_{i_{t-m+1},\dots,i_{t+1}}\}.\end{aligned}$$

(U3) Termination

$$P^* = \max_{1\le q_{T-m+1},\dots,q_T\le N}\{\delta_{q_{T-m+1},\dots,q_T}\}$$

$$(q^*_{T-m+1},\dots,q^*_T) = \text{argmax}_{1\le q_{T-m+1},\dots,q_T\le N}\{\delta_{q_{T-m+1},\dots,i_T}\}$$

8.2.3 Problem 3

In Problem 3, we attempt to adjust the model parameters Λ by maximizing the probability of the observation sequence given the model. Here we choose Λ such that $P[O|\Lambda]$ is maximized with the assumption that the distribution Π of the initial m states is known by using the EM algorithm. Define

$$C(\Lambda, \overline{\Lambda}) = \sum_{Q} P(Q|O, \Lambda) \log P(O, Q|\overline{\Lambda}).$$

The EM algorithm includes two main steps, namely E-step, calculating the function $C(\Lambda, \overline{\Lambda})$ and the M-step, maximizing $C(\Lambda, \overline{\Lambda})$ with respect to $\overline{\Lambda}$. Now, we define $\epsilon_t(i_1, i_2, \ldots, i_{m+1})$ as follows:

$$\epsilon_t(i_1, i_2, \ldots, i_{m+1}) = P(q_t = S_{i_1}, q_{t+1} = S_{i_2}, \ldots, q_{t+m} = S_{i_{m+1}}|O, \Lambda).$$

We can write down the expression of $\epsilon_t(i_1, i_2, \ldots, i_{m+1})$ in terms of $\alpha(\cdot)$ and $\beta(\cdot)$ that are computed in the previous two sub-sections:

$$\begin{aligned}
&\epsilon_t(i_1, i_2, \ldots, i_{m+1}) \\
&= b_{i_{m+1}}(O_{t+m}) P[O_{t+m+1} \ldots O_T | q_{t+1} = S_{i_2}, \ldots, q_{t+m} = S_{i_{m+1}}, \Lambda] \cdot \\
&\quad P(q_{t+m} = S_{i_{m+1}} | q_t = S_{i_1}, q_{t+1} = S_{i_2}, \ldots, q_{t+m-1} = S_{i_m}, \Lambda] \cdot \\
&\quad P[O_1 O_2 \ldots O_{t+m-1}, q_t = S_{i_1}, q_{t+1} = S_{i_2}, \ldots, q_{t+m-1} = S_{i_m} | \Lambda) \\
&= \alpha_{t+m-1}(i_1, i_2, \ldots, i_m) a_{i_1, \ldots, i_{m+1}} b_{i_{m+1}}(O_{t+m}) \beta_{t+1}(i_2, i_3, \ldots, i_{m+1}).
\end{aligned}$$

Therefore we obtain

$$\begin{aligned}
\epsilon_t(i_1, i_2, \ldots, i_{m+1}) &= P(q_t = S_{i_1}, q_{t+1} = S_{i_2}, \ldots, q_{t+m} = S_{i_{m+1}}|O, \Lambda) \\
&= \frac{\alpha_{t+m-1}(i_1, i_2, \ldots, i_m) a_{i_1, \ldots, i_{m+1}} b_{i_{m+1}}(O_{t+m}) \beta_{t+1}(i_2, i_e, \ldots, i_{m+1})}{P[O|\Lambda]}.
\end{aligned}$$

Next we define

$$\gamma_t(i_1, i_2, \ldots, i_k) = \sum_{i_{k+1}=1}^{N} \cdots \sum_{i_{m+1}=1}^{N} \epsilon_t(i_1, i_2, \ldots, i_{m+1}).$$

If we sum $\epsilon_t(i_1, i_2, \ldots, i_{m+1})$ over the index t, we get a quantity which can be interpreted as the expected number of times that the state sequence $S_{i_1} S_{i_2} \cdots S_{i_{m+1}}$ occurred. Similarly, if we sum $\gamma_t(i_1, i_2, \ldots, i_m)$ over t, we get a quantity which can be interpreted as the expected number of times that the state sequence $S_{i_1} S_{i_2} \cdots S_{i_m}$ occurred. Hence, a set of re-estimation formulae is given as follows:

$$\begin{cases}
\gamma_t(i_1) & = \displaystyle\sum_{i_2=1}^{N}\sum_{i_3=1}^{N}\cdots\sum_{i_{m+1}=1}^{N} \epsilon_t(i_1,i_2,\ldots,i_{m+1}),\\
\gamma_t(i_1,i_2) & = \displaystyle\sum_{i_3=1}^{N}\cdots\sum_{i_{m+1}=1}^{N} \epsilon_t(i_1,i_2,\ldots,i_{m+1}),\\
 & \vdots\\
\gamma_t(i_1,i_2,\ldots,i_m) & = \displaystyle\sum_{i_{m+1}=1}^{N} \epsilon_t(i_1,i_2,\ldots,i_{m+1}),\\
\overline{\pi_{i_1}} & = \gamma_1(i_1),\\
\overline{\pi_{i_1 i_2}} & = \gamma_1(i_1,i_2),\\
 & \vdots\\
\overline{\pi_{i_1 i_2\ldots i_m}} & = \gamma_1(i_1,i_2,\ldots,i_m),\\
A_{i_1 i_2\ldots i_{m+1}} & = \displaystyle\sum_{t=1}^{T-m} \epsilon_t(i_1,i_2,\ldots,i_{m+1}),\\
A_{i_1 i_2\ldots i_m} & = \displaystyle\sum_{i_{m+1}=1}^{N} A_{i_1 i_2\ldots i_{m+1}},\\
\overline{a_{i_1,\ldots,i_{m+1}}} & = A_{i_1 i_2\ldots i_{m+1}}/\displaystyle\sum_{i_{m+1}=1}^{N} A_{i_1 i_2\ldots i_{m+1}},\\
E_j(v_k) & = \displaystyle\sum_{t=1,\text{ such that } O_t=v_k}^{T-m} \gamma_t(j),\\
\overline{b_j(v_k)} & = E_j(v_k)/\displaystyle\sum_{k=1}^{M} E_j(v_k).
\end{cases}$$

8.2.4 The EM Algorithm

In this subsection, we discuss the convergence of the EM algorithm. We begin with the following lemma.

Lemma 8.1. *Given $p_i, q_i \geq 0$ such that*

$$\sum_i p_i = \sum_i q_i = 1,$$

then

$$\sum_i p_i \log \frac{p_i}{q_i} \geq 0$$

and the equality holds if and only if $p_i = q_i$ for all i.

Proof. Suppose that $p_i, q_i \geq 0$ and

$$\sum_i p_i = \sum_i q_i = 1,$$

then we have

$$\begin{aligned} -\sum_i p_i \log \frac{p_i}{q_i} &= \sum_i p_i \log \frac{q_i}{p_i} \\ &\leq \sum_i p_i (\frac{q_i}{p_i} - 1) \\ &= \sum_i (q_i - p_i) \\ &= 0. \end{aligned}$$

This is true because we have the following inequality

$$\log x \leq x - 1 \quad \text{for} \quad x \geq 0$$

and the equality holds if and only if $x = 1$. Hence the result follows.

Now, suppose we have a model with parameter set Λ and we want to obtain a better model with parameter set $\overline{\Lambda}$. Then one can consider the log likelihood as follows:

$$\log P[O|\overline{\Lambda}] = \sum_Q \log P[O, Q|\overline{\Lambda}].$$

Since

$$P[O, Q|\overline{\Lambda}] = P[Q|O, \overline{\Lambda}] P[O|\overline{\Lambda}],$$

we get

$$\log P[O|\overline{\Lambda}] = \log P[O, Q|\overline{\Lambda}] - \log P[Q|O, \overline{\Lambda}].$$

By multiplying this with $P[Q|O, \Lambda]$ and summing over Q, we get the following

$$\log P[O|\overline{\Lambda}] = \sum_Q P[Q|O, \Lambda] \log P[O, Q|\overline{\Lambda}] - \sum_Q P[Q|O, \Lambda] \log P[Q|O, \overline{\Lambda}].$$

We denote

$$C(\Lambda, \overline{\Lambda}) = \sum_Q P[Q|O, \Lambda] \log P[O, Q|\overline{\Lambda}]$$

then we have

$$\begin{aligned} \log P[O|\overline{\Lambda}] - \log P[O|\Lambda] &= C(\Lambda, \overline{\Lambda}) - C(\Lambda, \Lambda) \\ &\quad + \sum_Q P[Q|O, \Lambda] \log \frac{P[Q|O, \Lambda]}{P[Q|O, \overline{\Lambda}]}. \end{aligned}$$

The last term of the right-hand-side is the relative entropy of $P[Q|O,\Lambda]$ relative to $P[Q|O,\overline{\Lambda}]$ which is always non-negative by Lemma 8.1.

Hence we have

$$\log P[O|\overline{\Lambda}] - \log P[O|\Lambda] \geq C(\Lambda,\overline{\Lambda}) - C(\Lambda,\Lambda)$$

and the equality holds only if

$$\overline{\Lambda} = \Lambda$$

or if

$$P[Q|O,\overline{\Lambda}] = P[Q|O,\Lambda]$$

for some other $\overline{\Lambda} \neq \Lambda$. By choosing

$$\overline{\Lambda} = \arg\max_{\Lambda'} C(\Lambda,\Lambda')$$

one can always make the difference non-negative. Thus the likelihood of the new model is greater than or equal to the likelihood of the old model. In fact, if a maximum is reached then $\overline{\Lambda} = \Lambda$ and the likelihood remains unchanged. Therefore it can be shown that the EM algorithm converges to a (local or global) maximum.

Proposition 8.2. *The EM algorithm converges to a (local or global) maximum.*

8.2.5 Heuristic Method for Higher-Order HMMs

The conventional model for an mth order Markov model has $O(N^{m+1})$ unknown parameters (transition probabilities) where N is the number of states. The major problem in using this kind of model is that the number of parameters (transition probabilities) increases exponentially with respect to the order of the model. This large number of parameters discourages the use of higher-order Markov models directly. In this subsection, we develop an efficient estimation method for building a higher-order HMM when the observation symbol probability distribution B is known.

We consider the higher-order Markov model discussed in Chap. 6 whose number of states is linear in m. Our idea is to approximate an nth order Markov model of the demand as follows:

$$\mathbf{Q}_{t+m} = \sum_{i=1}^{m} \lambda_i P_i \mathbf{Q}_{t+m-i} \tag{8.2}$$

where $\mathbf{Q}_{t+i}$ is the state probability distribution vector at time $(t+i)$. In this model we assume that $\mathbf{Q}_{t+n+1}$ depends on $\mathbf{Q}_{t+i}$ $(i = 1, 2, \ldots, n)$ via the matrices P_i and the parameters λ_i. One may relate P_i to the ith step transition probability matrix for the hidden states. In the model, the number of parameters is $O(mN^2)$ whereas the conventional nth order Markov model has $O(N^{m+1})$ parameters to be determined.

Given the hidden state probability distribution, the observation probability distribution is given by

$$\mathbf{Y}_t = B\mathbf{X}_t \tag{8.3}$$

where B is the emission probabilities matrix. Hence (8.2) and (8.3) form a higher-order HMM.

For Model (8.2), in Chap. 6 we have proposed efficient methods to estimate A_i and λ_i. Given an observed sequence of $\{\mathbf{X}_t\}_{t=1}^{T}$, A_i is estimated by first counting the i-step transition frequency from the observed data sequence and then by a normalization to get the transition probabilities. In Chap. 6, we have proved that

$$\lim_{t\to\infty} \mathbf{X}_t = \mathbf{Z} \quad \text{and} \quad \mathbf{Z} = \sum_{i=1}^{m} \lambda_i P_i \mathbf{Z}$$

where $\mathbf{Z}$ can be estimated from $\{\mathbf{X}_t\}_{t=1}^{T}$ by first counting the occurrence frequency of each state and then by a normalization. We considered solving λ_i by the following minimization problem:

$$\min \left\| \mathbf{Z} - \sum_{i=1}^{m} \lambda_i P_i \mathbf{Z} \right\|$$

subject to

$$\sum_{i=1}^{m} \lambda_i = 1 \quad \text{and} \quad \lambda_i \geq 0.$$

It can be shown easily that if $||.||$ is taken to be $||.||_1$ or $||.||_\infty$ then the above problem can be reduced to a linear programming problem and hence can be solved efficiently.

In a higher-order HMM with known emission probabilities B and observation data sequence

$$O_1 O_2 \ldots O_T,$$

how does one choose A_i and λ_i so as to build a higher-order HMM? We note that by (8.3), the stationary probability distribution vector for the observation symbols is given by $\mathbf{W} = B\mathbf{Z}$. Therefore if $\mathbf{W}$ can be estimated and B is given, the probability distribution vector $\mathbf{Z}$ for the hidden states can be obtained. For the stationary vector $\mathbf{Z}$, the first-order transition probability matrix A for the hidden states is then given by

$$A = \mathbf{Z}(1, 1, \ldots, 1)^T \tag{8.4}$$

(noting that $A\mathbf{Z} = vecZ$). With this idea, we propose the following steps to construct a higher-order HMM.

Step 1: The lth element of $\mathbf{W}$ is approximated by

$$\frac{1}{T}\sum_{i=1}^{T} I_{O_i=v_l}.$$

Step 2: From (8.3), we expect $(\mathbf{W}-B\mathbf{Z})$ to be close to the zero vector. Therefore we consider solving $\mathbf{Z}$ by minimizing

$$||\mathbf{W} - B\mathbf{Z}||_\infty.$$

Step 3: Find the most probable hidden sequence Q_1, Q_2, …, Q_T based on the observation sequence

$$O_1, O_2, \ldots, O_T,$$

the matrix A is computed by (8.4).

Step 4: With the most probable hidden sequence

$$Q_1, Q_2, \ldots, Q_T,$$

we can estimate P_i by counting the number of the transition frequency of the hidden states and then by a normalization.

Step 5: Solve λ_i by solving

$$\min \left\| \mathbf{Z} - \sum_{i=1}^{m} \lambda_i P_i \mathbf{Z} \right\|_\infty$$

subject to

$$\sum_{i=1}^{m} \lambda_i = 1 \quad \text{and} \quad \lambda_i \geq 0.$$

The advantage of our proposed method is that one can solve the model parameters efficiently with reasonable accuracy.

8.3 The Double Higher-Order Hidden Markov Model

In this section, we present a discrete model for extracting information about the hidden or unobservable states from two observation sequences. The observations in each sequence not only depend on the hidden state information, but also depend on the hidden state's previous observations. It is clear that both the dynamics of hidden and observation states are required to model higher-order Markov chains. We call these kinds of models Double Higher-order Hidden Markov Models (DHHMMs).

The model can be described as follows. We write $\mathcal{T}$ for the time index set

$$\{0, 1, 2, \ldots\}$$

of the model. Let $\{V_t\}_{t \in \mathcal{T}}$ be an unobservable process representing the hidden states over different time periods. We assume that $\{V_t\}_{t \in \mathcal{T}}$ is an nth-order discrete-time time-homogeneous Markov chain process with the state space

$$\mathcal{V} = \{v_1, v_2, \ldots, v_M\}.$$

The state transition probabilities matrix $A = \{a(j_{t+n})\}$ of the nth-order Markov chain $\{V_t\}_{t \in \mathcal{T}}$ is given by

$$a(j_{t+n}) = P(V_{t+n} = v_{j_{t+n}} | V_t = v_{j_t}, \ldots, V_{t+n-1} = v_{j_{t+n-1}}) \\ 1 \leq j_t, \ldots, j_{t+n-1} \leq M. \tag{8.5}$$

To determine the probability structure for the nth-order Markov chain $\{V_t\}_{t \in \mathcal{T}}$ uniquely, we need to specify the initial state conditional probabilities $\Pi = \{\pi(i_j)\}$ as follows:

$$\pi(j_k) = P(V_k = v_{j_k} | V_1 = v_{j_1}, V_2 = v_{j_2}, \ldots, V_{k-1} = v_{j_{k-1}}), \quad 1 \leq k \leq n. \tag{8.6}$$

Let $\{I_t\}_{t \in \mathcal{T}}$ for a stochastic process, where it is assumed to be a (l, n)-order double hidden Markov chain process. Their corresponding states are given by $\{i_t\}_{t \in \mathcal{T}}$. Let

$$\mathbf{I}_t = (I_t, I_{t-1}, \ldots, I_{t-l+1})$$

and

$$\mathbf{i}_t = (i_t, i_{t-1}, \ldots, i_{t-l+1}).$$

Then, we assume that the transition probabilities matrix

$$B = \{b_{\mathbf{i}_t, v}(i_{t+1})\}$$

of the process $\{I_t\}_{t \in \mathcal{T}}$ and when $\mathbf{I}_t = \mathbf{i}_t$ and the hidden state $V_{t+1} = v$. The initial distribution Π for $\{I_t\}_{t \in \mathcal{T}}$ should be specified. Given appropriate values for n, M, I, A, l, Π and B, the DHHMM can be adopted to describe the generator that drives the realization of the observable sequence

$$I = I_1 I_2 \ldots I_T,$$

where T is the number of observations in the sequence. In order to determine the DHHMM for our applications one can apply similar method of maximum likelihood estimation and the EM algorithm discussed in Sect. 8.2. A detailed discussion of the model and method of estimation with applications to the extraction of unobservable states of an economy from observable spot interest rates and credit ratings can be found in Siu et al. [188].

8.4 The Interactive Hidden Markov Model

In this section, we propose an Interactive Hidden Markov Model (IHMM) where the transitions of hidden states depend on the current observable states. The IHHM is a generalization of the HMM discussed in Chap. 4. We note that this kind of HMM is different from classical HMMs where the next hidden states are governed by the previous hidden states only. An example is given to demonstrate IHMM. We then extend the results to give a class of general IHMMs.

8.4.1 An Example

Suppose that we are given a categorical data sequence (in steady state) of the volume of transactions as follows:

$$1, 2, 1, 2, 1, 2, 2, 4, 1, 2, 2, 1, 3, 3, 4, 1.$$

Here 1=high transaction volume, 2= medium transaction volume, 3=low transaction volume and 4=very low transaction volume. Suppose there are two hidden states: A (bull market period) and B (bear market period). In period A, the probability distribution of the transaction volume is assumed to follow

$$(1/4, 1/4, 1/4, 1/4).$$

In period B, the probability distribution of the transaction volume is assumed to follow

$$(1/6, 1/6, 1/3, 1/3).$$

In the proposed model, we assume that hidden states are unobservable but the transaction volume are observable. We would like to uncover the hidden state by modelling the dynamics with a Markov chain.

In the Markov chain, the states are

$$A, B, 1, 2, 3, 4.$$

We assume that when the observable state is i then the probabilities that the hidden state is A and B are given by α_i and $1-\alpha_i$ (depending on i) respectively in next time step. The transition probability matrix governing the Markov chain is given by

$$P_1 = \left(\begin{array}{cc|cccc} 0 & 0 & 1/4 & 1/4 & 1/4 & 1/4 \\ 0 & 0 & 1/6 & 1/6 & 1/3 & 1/3 \\ \hline \alpha_1 & 1-\alpha_1 & 0 & 0 & 0 & 0 \\ \alpha_2 & 1-\alpha_2 & 0 & 0 & 0 & 0 \\ \alpha_3 & 1-\alpha_3 & 0 & 0 & 0 & 0 \\ \alpha_4 & 1-\alpha_4 & 0 & 0 & 0 & 0 \end{array}\right).$$

8.4.2 Estimation of Parameters

In order to define the IHMM, one has to estimate the model parameters $\alpha_1, \alpha_2, \alpha_3$ and α_4 from an observed data sequence. One may consider the following two-step transition probability matrix as follows:

$$P_1^2 = \left(\begin{array}{cc|cccc} \frac{\alpha_1+\alpha_2+\alpha_3+\alpha_4}{4} & 1-\frac{\alpha_1+\alpha_2+\alpha_3+\alpha_4}{4} & 0 & 0 & 0 & 0 \\ \frac{\alpha_1+\alpha_2}{6}+\frac{\alpha_3+\alpha_4}{3} & 1-\frac{\alpha_1+\alpha_2}{6}-\frac{\alpha_3+\alpha_4}{3} & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & \frac{1}{6}+\frac{\alpha_1}{12} & \frac{1}{6}+\frac{\alpha_1}{12} & \frac{1}{3}-\frac{\alpha_1}{12} & \frac{1}{3}-\frac{\alpha_1}{12} \\ 0 & 0 & \frac{1}{6}+\frac{\alpha_2}{12} & \frac{1}{6}+\frac{\alpha_2}{12} & \frac{1}{3}-\frac{\alpha_2}{12} & \frac{1}{3}-\frac{\alpha_2}{12} \\ 0 & 0 & \frac{1}{6}+\frac{\alpha_3}{12} & \frac{1}{6}+\frac{\alpha_3}{12} & \frac{1}{3}-\frac{\alpha_3}{12} & \frac{1}{3}-\frac{\alpha_3}{12} \\ 0 & 0 & \frac{1}{6}+\frac{\alpha_4}{12} & \frac{1}{6}+\frac{\alpha_4}{12} & \frac{1}{3}-\frac{\alpha_4}{12} & \frac{1}{3}-\frac{\alpha_4}{12} \end{array}\right).$$

Using the same trick as in Chap. 4, one can extract the one-step transition probability matrix of the observable states from P_1^2 as follows:

$$\tilde{P}_2 = \begin{pmatrix} \frac{1}{6}+\frac{\alpha_1}{12} & \frac{1}{6}+\frac{\alpha_1}{12} & \frac{1}{3}-\frac{\alpha_1}{12} & \frac{1}{3}-\frac{\alpha_1}{12} \\ \frac{1}{6}+\frac{\alpha_2}{12} & \frac{1}{6}+\frac{\alpha_2}{12} & \frac{1}{3}-\frac{\alpha_2}{12} & \frac{1}{3}-\frac{\alpha_2}{12} \\ \frac{1}{6}+\frac{\alpha_3}{12} & \frac{1}{6}+\frac{\alpha_3}{12} & \frac{1}{3}-\frac{\alpha_3}{12} & \frac{1}{3}-\frac{\alpha_3}{12} \\ \frac{1}{6}+\frac{\alpha_4}{12} & \frac{1}{6}+\frac{\alpha_4}{12} & \frac{1}{3}-\frac{\alpha_4}{12} & \frac{1}{3}-\frac{\alpha_4}{12} \end{pmatrix}.$$

However, in this case, we do not have a closed form solution for the stationary distribution of the process. To estimate the parameter α_i, we first estimate the one-step transition probability matrix from the observed sequence. This can be done by counting the transition frequencies of the states in the observed sequence and we have

$$\hat{P}_2 = \begin{pmatrix} 0 & \frac{4}{5} & \frac{1}{5} & 0 \\ \frac{1}{2} & \frac{1}{3} & 0 & \frac{1}{6} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 1 & 0 & 0 & 0 \end{pmatrix}.$$

We expect that

$$\tilde{P}_2 \approx \hat{P}_2$$

and hence α_i can be obtained by solving the following minimization problem:

$$\min_{\alpha_i} ||\tilde{P}_2 - \hat{P}_2||_F^2 \tag{8.7}$$

subject to

$$0 \leq \alpha_i \leq 1.$$

Here $||.||_F$ is the Frobenius norm, i.e.

$$||A||_F^2 = \sum_{i=1}^{n} \sum_{i=1}^{n} A_{ij}^2.$$

This is equivalent to solving the following four independent minimization problems (i)–(iv) and they can be solved in parallel. This is an advantage of the estimation method. We remark that one can also consider other matrix norms for the objective function (8.7), let us say $||.||_{M_1}$ or $||.||_{M_\infty}$ and they may result in linear programming problems.

(i) α_1: $\min\limits_{0\leq\alpha_1\leq 1} \{(\frac{1}{6} + \frac{\alpha_1}{12})^2 + (\frac{1}{6} + \frac{\alpha_1}{12} - \frac{4}{5})^2 + (\frac{1}{3} - \frac{\alpha_1}{12} - \frac{1}{5})^2 + (\frac{1}{3} - \frac{\alpha_1}{12})^2\};$

(ii) α_2: $\min\limits_{0\leq\alpha_2\leq 1} \{(\frac{1}{6} + \frac{\alpha_1}{12} - \frac{1}{2})^2 + (\frac{1}{6} + \frac{\alpha_1}{12} - \frac{1}{3})^2 + (\frac{1}{3} - \frac{\alpha_1}{12})^2 + (\frac{1}{3} - \frac{\alpha_1}{12} - \frac{1}{6})^2\};$

(iii) α_3: $\min\limits_{0\leq\alpha_3\leq 1} \{(\frac{1}{6} + \frac{\alpha_1}{12})^2 + (\frac{1}{6} + \frac{\alpha_1}{12})^2 + (\frac{1}{3} - \frac{\alpha_1}{12} - \frac{1}{2})^2 + (\frac{1}{3} - \frac{\alpha_1}{12} - \frac{1}{2})^2\};$

(iv) α_4: $\min\limits_{0\leq\alpha_4\leq 1} \{(\frac{1}{6} + \frac{\alpha_1}{12} - 1)^2 + (\frac{1}{6} + \frac{\alpha_1}{12})^2 + (\frac{1}{3} - \frac{\alpha_1}{12})^2 + (\frac{1}{3} - \frac{\alpha_1}{12})^2\}.$

Solving the above optimization problems, we have

$$\alpha_1^* = 1, \quad \alpha_2^* = 1, \quad \alpha_3^* = 0, \quad \alpha_4^* = 1.$$

Hence we have

$$P_2 = \left(\begin{array}{cc|cccc} 0 & 0 & 1/4 & 1/4 & 1/4 & 1/4 \\ 0 & 0 & 1/6 & 1/6 & 1/3 & 1/3 \\ \hline 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{array}\right) \tag{8.8}$$

and

$$P_2^2 = \left(\begin{array}{cc|cccc} 3/4 & 1/4 & 0 & 0 & 0 & 0 \\ 2/3 & 1/3 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1/4 & 1/4 & 1/4 & 1/4 \\ 0 & 0 & 1/4 & 1/4 & 1/4 & 1/4 \\ 0 & 0 & 1/6 & 1/6 & 1/3 & 1/3 \\ 0 & 0 & 1/4 & 1/4 & 1/4 & 1/4 \end{array}\right). \tag{8.9}$$

8.4.3 Extension to the General Case

The method can be extended to a general case of m hidden states and n observable states. We note the one-step transition probability matrix of the observable states is given by

$$\tilde{P}_2 = \begin{pmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1m} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_{n1} & \alpha_{m2} & \cdots & \alpha_{nm} \end{pmatrix} \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ p_{m1} & p_{m2} & \cdots & p_{mn} \end{pmatrix}, \tag{8.10}$$

i.e.

$$[\tilde{P}_2]_{ij} = \sum_{k=1}^{m} \alpha_{ik} p_{kj} \quad i, j = 1, 2, \ldots, n.$$

Here we assume that α_{ij} are unknowns and the probabilities p_{ij} are given. Suppose $[Q]_{ij}$ is the one-step transition probability matrix estimated from the observed sequence. Then for each fixed i, α_{ij}, $j = 1, 2, \ldots, m$ can be obtained by solving the following constrained least squares problem:

$$\min_{\alpha_{ik}} \left\{ \sum_{j=1}^{n} \left(\sum_{k=1}^{m} \alpha_{ik} p_{kj} - [Q]_{ij} \right)^2 \right\}$$

subject to

$$\sum_{k=1}^{m} \alpha_{ik} = 1$$

and

$$\alpha_{ik} \geq 0 \quad \text{for all } i, k.$$

The idea of the IHMM presented in this subsection is further extended to address the following applications and problems in Ching et al. [66].

1. IHMM is applied to some practical data sequences in sales demand data sequences.
2. There are only a few works on modeling the non-linear behavior of categorical time series that can be found in the literature. In the continuous-state case, the threshold auto-regressive model is a well-known approach. The idea is to provide a piecewise linear approximation to a non-linear autoregressive time series model. This is achieved by dividing the state space into several regimes via the threshold principle. The IHMM provides a first-order approximation of the non-linear behavior of categorical time series by dividing the state space of the Markov chain process into several regimes.

8.5 The Binomial Expansion Model for Portfolio Credit Risk Modulated by the IHMM

The binomial expansion model for portfolio credit risk is a simple and intuitive technique for describing defaults in a credit portfolio. It is popular in the banking and finance industries. The key idea of the binomial expansion model is to describe the number of defaults from a credit portfolio, for a particular period of time as a binomially distributed random variable. The credit portfolio has a finite number of consistent entities (or credit). It is implicit in this distributional assumption that

1. The reference entities are homogeneous in the sense that their probabilities of defaults are the same.
2. The common probability of defaults remains the same over time.
3. The defaults of the reference entities are independent to each other.

Indeed, when the size of the portfolio becomes large and the common probability of defaults becomes small, the number of defaults in the large credit portfolio will approximately follow a Poisson distribution. From this asymptotic relationship, the relationship between the binomial expansion model for portfolio credit risk and the (top-down) reduced-form credit risk model is revealed. Note that the (top-down) reduced-form model means modeling defaults at a portfolio level using the reduced-form (intensity-based) credit risk model. Another reduced-form credit risk model is a bottom-up one, where defaults are modeled by random point processes at the level of individual reference entities and a certain aggregation procedure is then used to describe defaults at a portfolio level.

Despite its simplicity and popularity, the independence and homogeneity assumptions in the binomial expansion model can hardly be justified by empirical experience. This motivates us to consider an extension of the binomial expansion model, where the probability of default of each reference entity is modulated by an IHMM. Another extension of the binomial expansion model was given in [209].

In this section, we discuss the binomial expansion model for portfolio credit risk modulated by the IHMM. This model can incorporate the impact of different risk states on the probability of defaults, time-varying probability of defaults and the dependency of defaults, of the reference entities.

Suppose there are m hidden common risk states to all of the reference entities in a credit portfolio. For example, these entities could be corporate bonds issued by firms in the same sector, or related sectors. We assume that there are n entities in the portfolio. In other words, the number of surviving entities in the portfolio at the beginning of the first period equal n, (i.e., $S_0 = n$).

We consider a discrete-time economy with a time parameter set $\mathcal{T} := \{0, 1, \cdots, T\}$. To describe uncertainty, we define a complete probability space $(\Omega, \mathcal{F}, P)$, where $\mathcal{P}$ is a real-world probability measure. The probability space is rich enough to model all sources of uncertainty in our modeling framework.

Let $X := \{X_t, t \in \mathcal{T}\}$ denote a hidden sequence of random variables defined on $(\Omega, \mathcal{F}, P)$ with state-space

$$\mathcal{S} := \{s_1, s_2, \ldots, s_m\},$$

where $s_i \in \mathcal{R}^m$, for $i = 1, 2, \ldots, m$.

We interpret the states of X as the common risk states of the entities in the portfolio. In particular, “1” represents the lowest risk state and “m” represents the highest risk state. Again, without loss of generality, we identify the state-space $\mathcal{S}$ to be the set $\{e_1, e_2, \ldots, e_m\}$ of standard unit vectors in $\mathcal{R}^m$. This set of standard unit vectors represents the orthornormal basis in the Hilbert space $\mathcal{R}^m$. In the sequel, we model the state process X by an IHMM model.

Let $\{S_t, t \in \mathcal{T}\}$ be a stochastic process defined on $(\Omega, \mathcal{F}, P)$, where S_t represents the number of surviving entities at the end of the t^{th} period, for each $t \in \mathcal{T}$. Then, for each $t \in \mathcal{T} \backslash \{0, 1\}$ and each $k = 0, 1, 2, \ldots, S_{t-2}$, the transition probabilities of the hidden risk state process X are specified as below:

$$a_{ij}(k) = \mathcal{P}(X_t = e_j | X_{t-1} = e_i, M_{t-1} = k)$$

and

$$A(k) = [a_{ij}(k)]_{1 \leq i,j \leq m}.$$

Here $\{M_t, t \in \mathcal{T}\}$ is a stochastic process defined on $(\Omega, \mathcal{F}, P)$ and M_t represents the number of defaults in the t^{th} period. From the transition probabilities we can see that the feedback effect of the number of default entities in the portfolio in the previous time period on the common hidden risk state of the consistent entities in the current time period is incorporated. Intuitively, if the number of defaults in the portfolio in the previous period is large, the common hidden risk state of the portfolio in the current period will be large.

Let $F^S := \{\mathcal{F}_t^S, t \in \mathcal{T}\}$ be the P-completed natural filtration generated by the survival process S, where, for each $t \in \mathcal{T}$, $\mathcal{F}_t^S$ is the σ-field generated by the survival process S up to and including time t augmented by the P-null sets in $\mathcal{F}$. For each $t \in \mathcal{T}$, conditional on $\mathcal{F}_t^S$, M_{t+1} takes values in the set $\{0, 1, 2, \dots, S_t\}$, since the number of defaulting entities in the current time period cannot exceed the number of surviving entities in the previous time period.

Let $\langle \cdot, \cdot \rangle$ be the scalar, (or inner), product in $\mathcal{R}^m$. We can then define the probability of default for each consistent entity in the portfolio at time t as follows:

$$\Theta_t := \Theta(t, X_t) := \langle \mathbf{\Theta}, X_t \rangle = \sum_{i=1}^{m} \Theta_i \, \langle X_t, e_i \rangle,$$

where

$$\mathbf{\Theta} := (\Theta_1, \Theta_2, \dots, \Theta_m) \in \mathcal{R}^m, \quad \Theta_i \in (0, 1) \quad \text{for } i = 1, 2, \dots, m,$$

and

$$\Theta_1 < \Theta_2 < \cdots < \Theta_m.$$

This is consistent with the assumption that "1" is the lowest risk state and "m" is the highest risk state.

For each $t \in \mathcal{T}$, given $\mathcal{F}_t^S$ and X_t, the conditional probability distribution of M_{t+1} under P is given by the following regime-switching binomial distribution modulated by the IHMM:

$$\begin{aligned} P(M_{t+1} = j | X_t, \mathcal{F}_t^S) &= P(M_{t+1} = j | X_t, S_t) \qquad (8.11) \\ &= \binom{S_t}{j} (\langle \mathbf{\Theta}, X_t \rangle)^j (1 - \langle \mathbf{\Theta}, X_t \rangle)^{S_t - j} \\ &= \sum_{i=1}^{m} \binom{S_t}{j} (\Theta_i)^j (1 - \Theta_i)^{S_t - j} \, \langle X_t, e_i \rangle, \qquad (8.12) \end{aligned}$$

for each $j = 1, 2, \dots, S_t$, where $\binom{S_t}{j}$ is the number of combinations of j objects among S_t objects.

The following example provides an illustration of the model. The version of the binomial expansion model presented in the example is an extension to that of Giampieri et al. [105] in the sense that the IHMM is used to incorporate the feedback effect.

8.5.1 Examples

Consider the situation that the hidden risk state takes two possible values, namely, "N" (normal risk) and "E" (enhanced risk). In the normal risk state, the number of observed defaults in each time step is modeled by a binomial expansion model with the default probability of a bond being equal to P_N. In the enhanced risk state, let P_E denote the default probability of a bond. Here $P_E > P_N$, which means that the probability of the default of a bond in the enhanced risk state is higher than that in the normal risk state.

Then, in the normal state N, given the number of surviving bonds at the current time is k, the conditional probability distribution of the number of defaults in the next period is:

$$P(m|Nk) = \binom{k}{m}(P_N)^m(1-P_N)^{k-m}, \quad m = 1, 2, \ldots, k.$$

In the enhanced state E, the corresponding probability distribution is

$$P(m|Ek) = \binom{k}{m}(P_E)^m(1-P_E)^{k-m}.$$

In the following example, we shall define an augmented Markov chain associated with the binomial expansion model modulated by an IHMM with two hidden risk states. The state-space of the augmented Markov chain is formed by both hidden risk states and the observable number of defaults. We obtain the transition probability matrix of the augmented Markov chain, which provides a complete description for the probabilistic behavior of transitions of both observable states and hidden states. The transition probability matrix also plays a key role for the estimation of the unknown parameters of the IHMM.

Consider the binomial expansion model in the above example again. Suppose there are two surviving bonds at the beginning of the first period, i.e., $S_0 = 2$. In this case, there are three observable states 0, 1 and 2, while there are six unobservable states (or more precisely, partially unobservable), namely, $N2$, $N1$, $N0$, $E2$, $E1$ and $E0$. Here, $N2$ represents that the credit risk is normal and there are two surviving bonds and $E0$ means that the credit risk is enhanced and there is no surviving bond. We shall describe the probabilistic behavior of the transitions of both observable states and unobservable states by an augmented Markov chain with the following nine states:

$$\{N2, N1, N0, E2, E1, E0, 2, 1, 0\}.$$

We further suppose that when the observable state is k ($k = 0, 1, 2$), the probabilities that the hidden state is N and E in next time step are given by α_k and $1 - \alpha_k$, respectively.

Then, the transition probability matrix governing the augmented Markov chain is:

$$P_2 = \left(\begin{array}{cccccc|ccc} 0 & 0 & 0 & 0 & 0 & 0 & P(0|N2) & P(1|N2) & P(2|N,2) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & P(0|N1) & P(1|N1) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & P(0|N0) \\ 0 & 0 & 0 & 0 & 0 & 0 & P(0|E2) & P(1|E2) & P(2|E2) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & P(0|E1) & P(1|E1) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & P(0|E0) \\ \hline \alpha_2 & 0 & 0 & 1-\alpha_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \alpha_1 & 0 & 0 & 1-\alpha_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha_0 & 0 & 0 & 1-\alpha_0 & 0 & 0 & 0 \end{array} \right). \qquad (8.13)$$

Suppose

$$P_N = 0.15 \quad \text{and} \quad P_E = 0.3.$$

Then, the transition probability matrix is

$$P_2 = \left(\begin{array}{cccccc|ccc} 0 & 0 & 0 & 0 & 0 & 0 & 0.7225 & 0.2550 & 0.0225 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.8500 & 0.1500 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.000 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.4900 & 0.4200 & 0.0900 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.7000 & 0.3000 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.000 \\ \hline \alpha_2 & 0 & 0 & 1-\alpha_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \alpha_1 & 0 & 0 & 1-\alpha_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha_0 & 0 & 0 & 1-\alpha_0 & 0 & 0 & 0 \end{array} \right).$$

8.5.2 *Estimation of the Binomial Expansion Model Modulated by the IHMM*

To illustrate how to use the above method to estimate the binomial expansion model modulated by the IHMM, we first consider the situation where P_N and P_E are given. To define the IHMM for the hidden risk states, one must first estimate $\alpha = (\alpha_0, \alpha_1, \alpha_2)$ from the observed sequence of the default data. We consider the 3×3 sub-matrix related to the observable states of the two-step transition probability matrix P_2^2 as follows: $\tilde{P}_2 =$

$$\begin{pmatrix} \alpha_2 P(0|N,2) + (1-\alpha_2)P(0|N,2) & \alpha_2 P(1|N,2) + (1-\alpha_2)P(1|N,2) & \alpha_2 P(2|N,2) + (1-\alpha_2)P(2|N,2) \\ 0 & \alpha_1 P(0|N,1) + (1-\alpha_2)P(0|N,1) & \alpha_1 P(1|N,1) + (1-\alpha_2)P(1|N,1) \\ 0 & 0 & 1 \end{pmatrix}.$$

Suppose $P_N = 0.15$ and $P_E = 0.3$. Then,

$$\tilde{P}_2 = \begin{pmatrix} 0.2325\alpha_2 + 0.4900 & -0.1650\alpha_2 + 0.4200 & -0.0675\alpha_2 + 0.0900 \\ 0.0000 & 0.1500\alpha_1 + 0.7000 & -0.1500\alpha_1 + 0.3000 \\ 0.0000 & 0.0000 & 1.0000 \end{pmatrix}.$$

Note that α_0 is not important in the model and can be assumed to be 0. We also notice that $P(0|N0) = P(0|E0) = 1$.

To estimate the parameters α_i $(i = 1, 2)$ given P_N and P_E, we must first estimate the one-step transition probability matrix from the observed sequence of default data. This can be done by counting transition frequencies among the states in the observed sequence of default data and then follow by a normalization (see, for example, [59, 155]). Suppose that the estimated transition probability matrix is given by:

$$\hat{P}_2 = \begin{pmatrix} \frac{3}{4} & \frac{1}{4} & 0 \\ 0 & \frac{3}{4} & \frac{1}{4} \\ 0 & 0 & 1 \end{pmatrix}.$$

Note that $\tilde{P}_2 \approx \hat{P}_2$. Consequently, α_i can be estimated by solving the following minimization problem:

$$\min_{\alpha_i} ||\tilde{P}_2 - \hat{P}_2||_F^2 \qquad (8.14)$$

subject to the constraints:

$$0 \le \alpha_i \le 1, \quad i = 1, 2\ .$$

Here $||.||_F$ represents the Frobenius norm. Solving the above optimization problem, we obtain

$$\alpha_1^* = 0.3333 \quad \text{and} \quad \alpha_2^* = 0.9602.$$

This then gives:

$$P_2 = \left(\begin{array}{cccccc|ccc} 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.7225 & 0.2550 & 0.0225 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.8500 & 0.1500 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 1.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.4900 & 0.4200 & 0.0900 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.7000 & 0.3000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 1.0000 \\ \hline 0.9602 & 0.0000 & 0.0000 & 0.0398 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.3333 & 0.0000 & 0.0000 & 0.6667 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{array}\right).$$

The general situation where both P_N and P_E are unknown can be handled using a bi-level programming technique (see [70], Appendix A therein).

8.5.3 Numerical Examples and Comparison

In this subsection, we present the estimation results of the IHMMs using the observed default data in [105]. In [105], Giampieri et al. applied the HMM to the quarterly bond default data of four sectors (consumer, energy, media and transportation) in the United States, taken from Standard & Poors' ProCredit6.2 database. The data set covers the period from January 1981 to December 2002. The total number of bonds in January 1981 was 281 while the total number of bonds in December 2002 was 222. For the convenience of comparison, we extracted the credit default data and also the most likely hidden risk state directly from the figures in [105]. We then applied our IHMM to the extracted data. Comparisons were made and the results are reported in Figs. 8.1–8.8. The details of the computational procedures are given below.

Now, we need to set the initial probability for each sector and for each model. We assume that the initial probability matrices of the consumer sector for both the IHMM and the HMM are the same and that the common probability matrix is:

$$P^{(0)} = \begin{pmatrix} 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 \end{pmatrix}. \qquad (4.1)$$

We recall that the number of observable states is equal to the maximum number of defaults observed plus one in the IHMM. The initial probability matrices of the energy sector and the transportation sector for both the IHMM and the HMM are supposed to be the same and the common probability matrix is:

$$P^{(0)} = \begin{pmatrix} 1/3 & 1/3 & 1/3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/3 & 1/3 & 1/3 \end{pmatrix} \qquad (4.2)$$

The initial probability matrices of the media sector for both the IHMM and the HMM are assumed to be the same and the common probability matrix is:

$$P^{(0)} = \begin{pmatrix} 1/4 & 1/4 & 1/4 & 1/4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/4 & 1/4 & 1/4 & 1/4 \end{pmatrix} \qquad (4.3)$$

Figures 8.1–8.8 depict the results of the most likely hidden risk state extracted from the IHMM and also those extracted from the figures in [105] using the observed default data in the consumer/service sector, the energy and natural resources sector, the leisure time/media sector and the transportation sector respectively. There are 13, 6, 8 and 6 observable states in the consumer/service sector, the energy and natural resources sector, the leisure time/media sector and the transportation sector respectively.

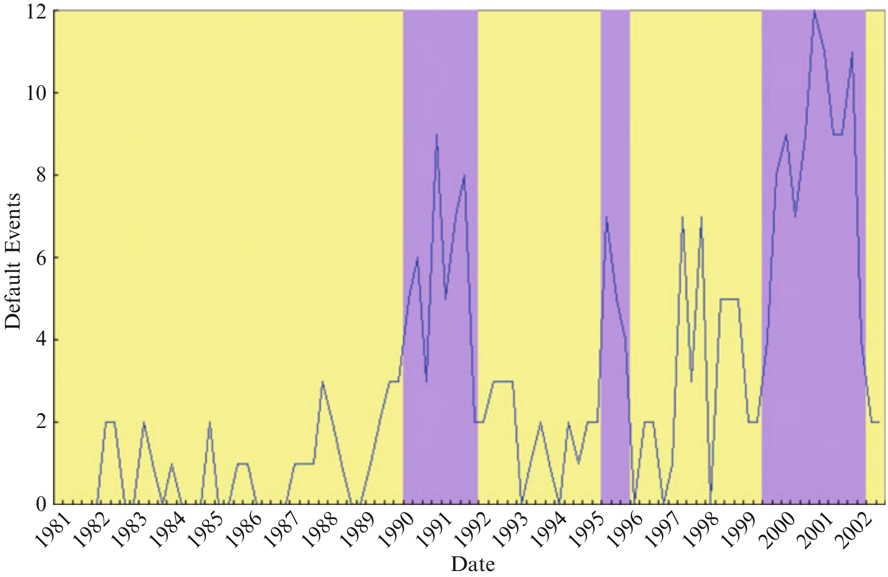


Fig. 8.1 Consumer/service sector (HMM in [105]) (Taken from [70])

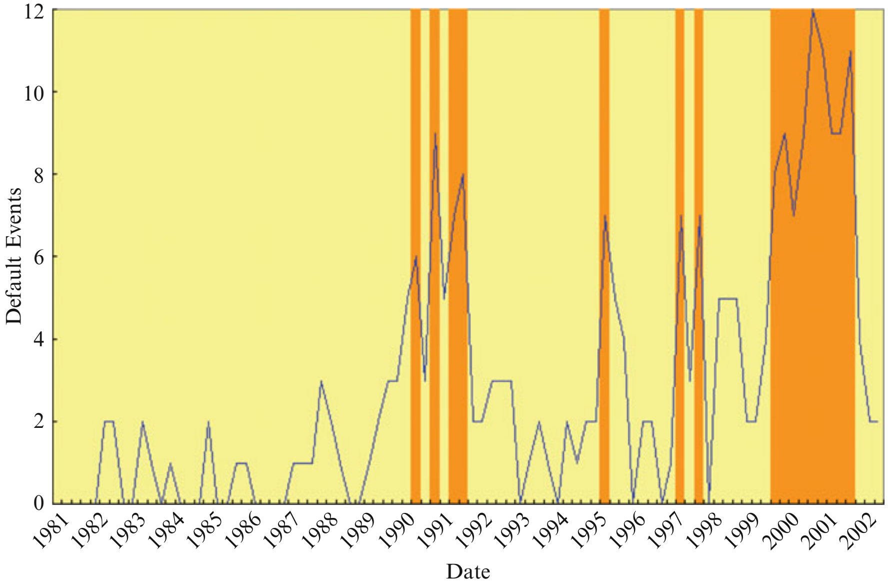


Fig. 8.2 Consumer/service sector (IHMM) (Taken from [70])

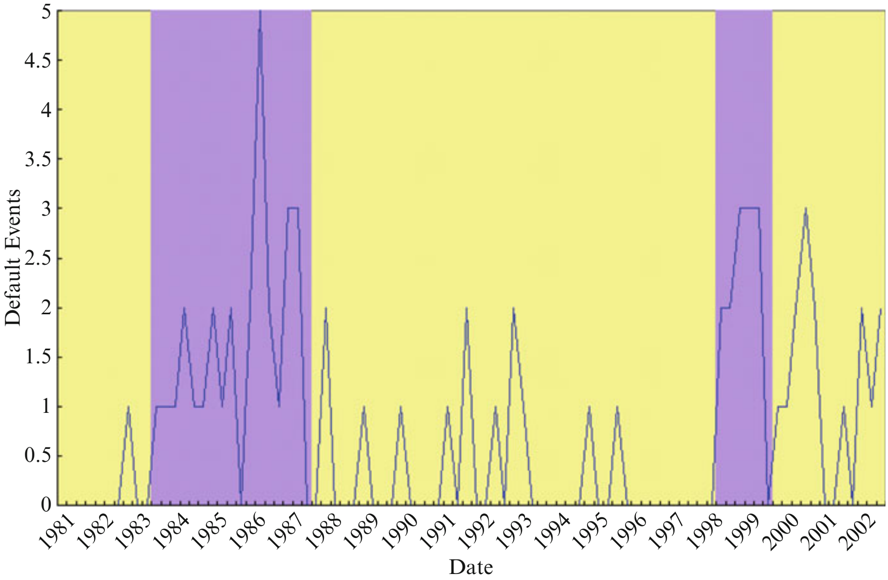


Fig. 8.3 Energy and natural resources sector (HMM in [105]) (Taken from [70])

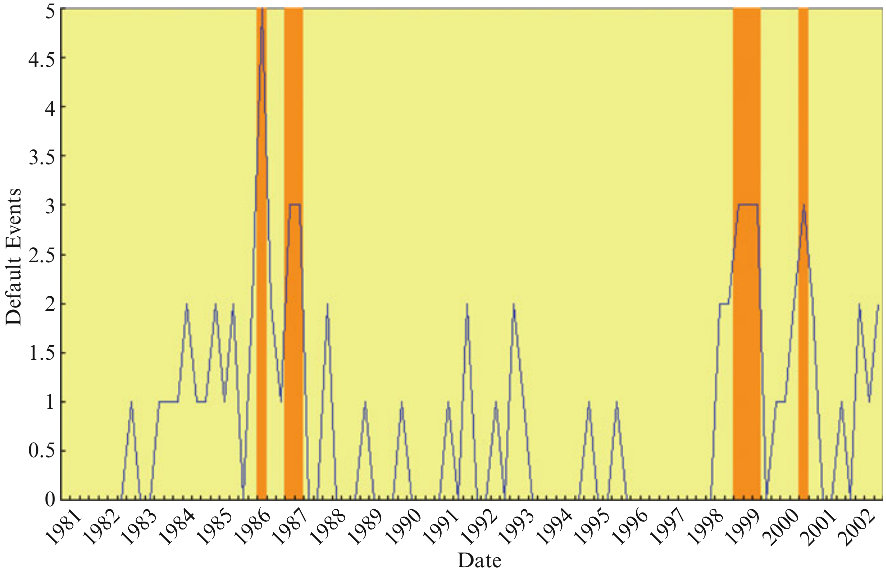


Fig. 8.4 Energy and natural resources sector (IHMM) (Taken from [70])

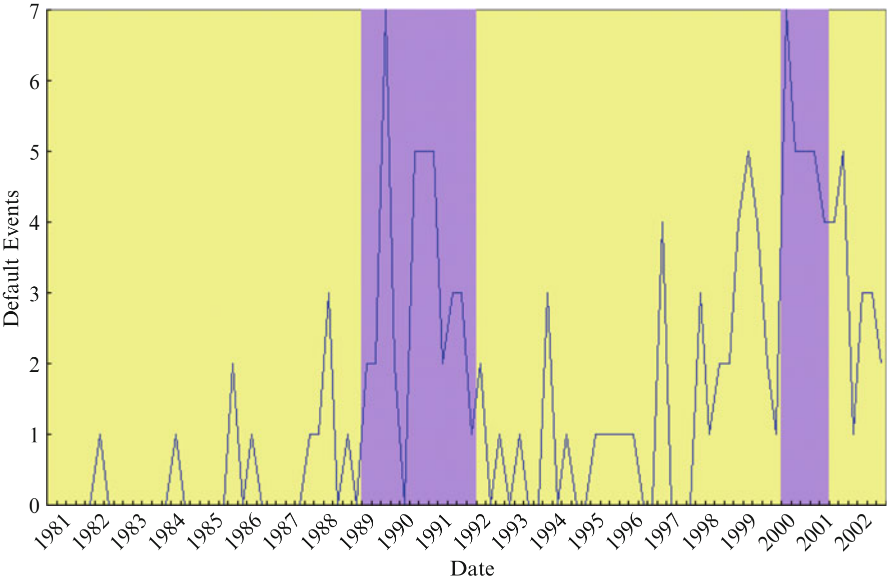


Fig. 8.5 Leisure time/media sector (HMM in [105]) (Taken from [70])

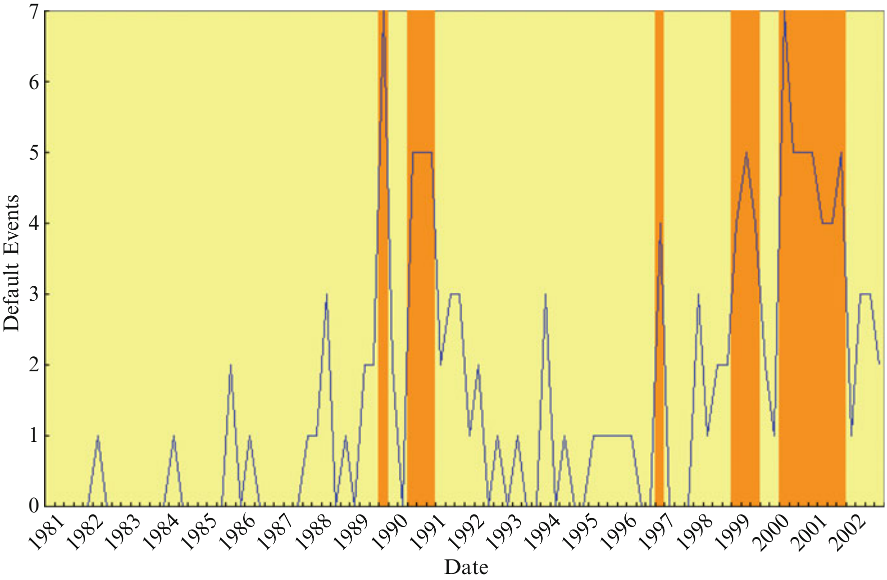


Fig. 8.6 Leisure time/media sector (IHMM) (Taken from [70])

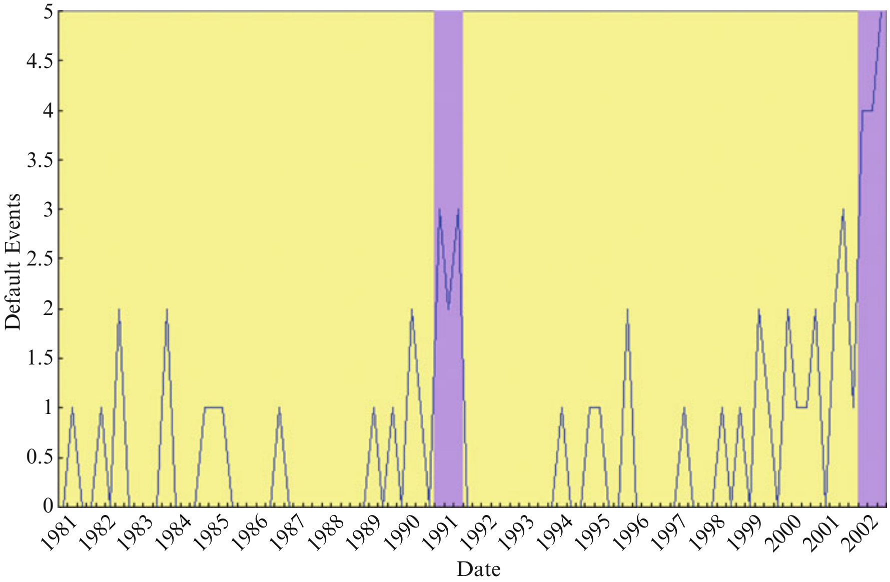


Fig. 8.7 Transportation sector (HMM in [105]) (Taken from [70])

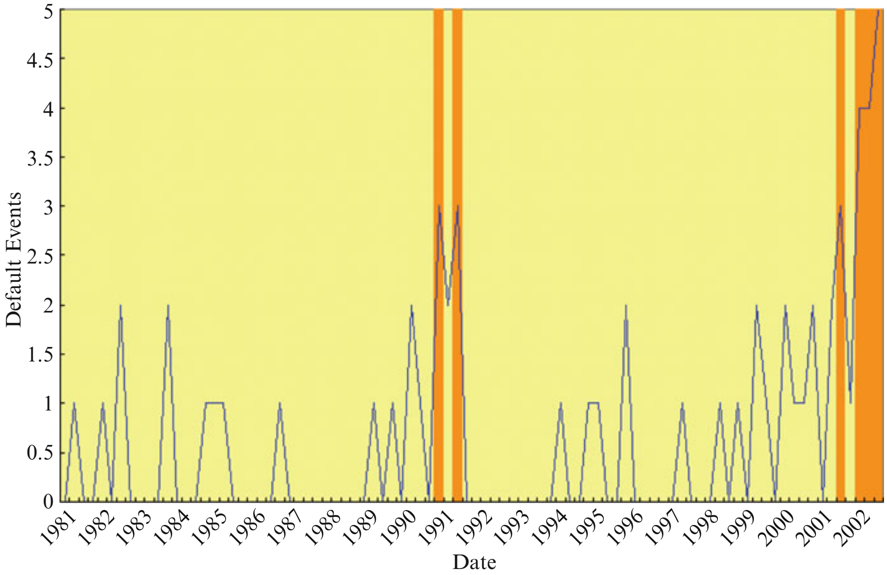


Fig. 8.8 Transportation sector (IHMM) (Taken from [70])

Table 8.1 Prediction accuracy in the sales demand data

			IHMM			HMM in [105]			
Sectors	Total	Default	α	P_N	P_E	q	p	P_N	P_E
Consumer	1,041	251	0.63	0.0022	0.0075	0.95	0.81	0.0026	0.0159
Energy	420	71	0.68	0.0015	0.0085	0.95	0.88	0.0014	0.0099
Media	650	133	0.50	0.0015	0.0085	0.96	0.83	0.0027	0.0194
Transport	281	59	0.63	0.0017	0.0153	0.97	0.78	0.0025	0.0223

From Figs. 8.1–8.8, we can see that our IHMM is more sensitive in detecting the changes in the hidden risk states than the HMM in [105] for the consumer/service sector, the energy and natural resources sector, the leisure time/media sector and the transportation sector. This reveals that the incorporation of the feedback effect by the IHMM can improve the ability in detecting the changes in the hidden risk states. As expected, the IHMM gives a threshold-type classification of the hidden risk states. For example, the IHMM classifies those periods having six or more defaults as enhanced risk in the consumer/service sector. The threshold values for the remaining three sectors are $3, 4$ and 3 defaults respectively. For the HMM, generally speaking, it is unlikely to have two transitions of hidden risk states in three consecutive transitions. Therefore, the HMM might not be adaptive to the rapid changes of hidden risk states. This is consistent with the numerical examples in Figs. 8.1–8.8.

We then apply the binomial expansion model modulated by the IHMM to the default data again. In this case, since the number of model parameters α_i is much more than the number of available data points, we assume that $\alpha_i = \alpha$, for all i, in the estimation. Based on the observed default data and the hidden risk state process for each sector extracted by our IHMM [43] the likelihood function, or the joint probability distribution, for the hidden risk states with the observed default data can be obtained in the following form:

$$\alpha^P(1-\alpha)^Q P_N^R(1-P_N)^S P_E^T(1-P_E)^U. \tag{4.4}$$

Here P, Q, R, T and U can be obtained from the observed default data. The estimates of all model parameters are then obtained by maximizing the above likelihood function (4.4).

The parameter estimates of the binomial expansion models, modulated either by the IHMM or HMM, for the four industry sectors are presented in Table 8.1. Under the binomial expansion model modulated by the HMM, the hidden risk state is assumed to follow a first-order Markov chain having the following transition probability matrix:

$$\begin{pmatrix} q & 1-q \\ 1-p & p \end{pmatrix}.$$

Here q is the probability of remaining in the normal risk state while p represents the probability of remaining in the enhanced risk state. We observe that the default probabilities under the enhanced risk, state estimated using the binomial expansion model modulated by the HMM, are always significantly greater than those estimated by the binomial expansion model modulated by IHMM. It can also be observed that the default probabilities under the normal risk state obtained by both of the models are relatively consistent with each other.

We considered the IHMM and a binomial expansion model modulated by an IHMM for modeling the occurrence of defaults of bonds issued by firms in the same sector. The main idea of the two models is to assume that the transitions of the hidden risk states of the sector depend on the current observed number of bonds defaulting within the sector. We presented an efficient estimation method for the model parameters and an efficient method for extracting the most likely hidden risk state process. We conducted empirical studies on the models and compared the hidden risk state process extracted from the IHMM model with that extracted from the HMM using the real default data from Giampieri et al. [105]. We found that the incorporation of the interactive or feedback effect can provide a more sensitive way to detect the transitions in the hidden risk states.

8.6 Summary

In this chapter, we presented several frameworks for hidden Markov models (HMMs). These frameworks include the Higher-order Hidden Markov Model (HHMM), the Interactive Hidden Markov Model (IHMM) and the Double Higher-order Hidden Markov Model (DHHMM). For both HHMM and IHMM, we present both methods and efficient algorithms for the estimation of model parameters. Applications of these models for extracting economic information from observed interest rate and credit ratings data and for extending the binomial expansion model for portfolio credit risk analysis are discussed.

8.7 Exercises

1. Derive the conditional probability distribution of M_{t+1} in (8.11) given $\mathcal{F}_t^S$ and X_t under P if X_t is an hidden Markov chain.
2. Derive the transition probability matrix P_2 in (8.13) of the augmented Markov chain.
3. Write a computer program (use EXCEL) to solve the minimization problem (8.14).
4. Use a bi-level programming technique to estimate α_1, α_2 and P_2 in (8.14) when both P_N and P_E are unknown.

References

1. Adke S, Deshmukh D (1988) Limit distribution of a high order Markov chain. J Roy Stat Soc Ser B 50:105–108
2. Albrecht D, Zukerman I, Nicholson A (1999) Pre-sending documents on the WWW: a comparative study. In: Proceedings of the sixteenth international joint conference on artificial intelligence IJCAI99
3. Altman E (1999) Constrained markov decision processes. Chapman and Hall/CRC, London
4. Ammar G, Gragg W (1988) Superfast solution of real positive definite Toeplitz systems. SIAM J Matrix Anal Appl 9:61–76
5. Artzner P, Delbaen F, Eber J, Heath D (1997) Thinking coherently. Risk 10:68–71
6. Artzner P, Delbaen F, Eber J, Heath D (1999) Coherent measures of risk. Math Finance 9:203–228
7. Avrachenkov L, Litvak N (2004) Decomposition of the Google PageRank and optimal linking strategy. Research report, INRIA, Sophia Antipolis
8. Axelsson O (1996) Iterative solution methods. Cambridge University Press, New York
9. Axsäter S (1990) Modelling emergency lateral transshipments in inventory systems. Manag Sci 36:1329–1338
10. Baldi P, Frasconi P, Smith P (2003) Modeling the internet and the web. Wiley, England
11. Baum L (1972) An inequality and associated maximization techniques in statistical estimation for probabilistic function of Markov processes. Inequality 3:1–8
12. Bell D, Atkinson J, Carlson J (1999) Centrality measures for disease transmission networks. Soc Network 21:1–21
13. Berger P, Nasr N (1998) Customer lifetime value: marketing models and applications. J Interact Market 12:17–30
14. Berger P, Nasr N (2001) The allocation of promotion budget to maximize customer equity. Omega 29:49–61
15. Berman A, Plemmons R (1994) Nonnegative matrices in the mathematical sciences. Society for industrial and applied mathematics, Philadelphia
16. Bernardo J, Smith A (2001) Bayesian theory. Wiley, New York
17. Best P (1998) Implementing value at risk. Wiley, England
18. Bini D, Latouche G, Meini B (2005) Numerical methods for structured Markov chains. Oxford University Press, New York
19. Black F, Scholes M (1973) The pricing of options and corporate liabilities. J Polit Econ 81:637–659
20. Blattberg R, Deighton J (1996) Manage market by the customer equity. Harv Bus Rev 73:136–144

W.-K. Ching et al., *Markov Chains*, International Series in Operations Research
& Management Science 189, DOI 10.1007/978-1-4614-6312-2,

21. Blumberg D (2005) Introduction to management of reverse logistics and closed loop supply chain processes. CRC, Boca Raton
22. Bonacich P, Lloyd P (2001) Eigenvector-like measures of centrality for asymmetric relations. Soc Network 23:191–201
23. Bonacich P, Lloyd P (2004) Calculating status with negative relations. Soc Network 26:331–338
24. Bramble J (1993) Multigrid methods. Longman Scientific and Technical, Essex
25. Brockwell P, Davis R (1991) Time series: theory and methods. Springer, New York
26. Buchholz P (1994) A class of hierarchical queueing networks and their analysis. Queueing Syst 15:59–80
27. Buchholz P (1995) Hierarchical Markovian models: symmetries and aggregation. Perform Eval 22:93–110
28. Buchholz P (1995) Equivalence relations for stochastic automata networks. In: Computations of Markov chains: proceedings of the 2nd international workshop on numerical solutions of Markov chains. Kluwer, Dordecht, pp 197–216
29. Buffington J, Elliott R (2002) Regime switching and European options. In: Stochastic theory and control, proceedings of a workshop, Lawrence, K.S. Springer, Berlin, pp 73–81
30. Buffington J, Elliott R (2002) American options with regime switching. Int J Theoret Appl Finance 5:497–514
31. Bühlmann H, Gisler A (2005) A course in credibility theory and its applications. Springer, Berlin
32. Bunch J (1985) Stability of methods for solving Toeplitz systems of equations. SIAM J Scient Stat Comput 6:349–364
33. Bunke H, Caelli T (2001) In: Bunke H, Caelli T (eds) Hidden Markov models: applications in computer vision. World Scientific, Singapore
34. Buzacott J, Shanthikumar J (1993) Stochastic models of manufacturing systems. Prentice-Hall International Editions, New Jersey
35. Carpenter P (1995) Customer lifetime value: do the math. Market Comput 15:18–19
36. Chan R, Ching W (1996) Toeplitz-circulant preconditioners for Toeplitz systems and their applications to queueing networks with batch arrivals. SIAM J Scientif Comput 17:762–772
37. Chan R, Ching W (2000) Circulant preconditioners for stochastic automata networks. Numerise Mathematik 87:35–57
38. Chan R, Ma K, Ching W (2006) Boundary value methods for solving transient solutions of Markovian queueing networks. J Appl Math Comput 172:690–700
39. Chan R, Ng M (1996) Conjugate gradient method for Toeplitz systems. SIAM Rev 38:427–482
40. Chang Q, Ma S, Lei G (1999) Algebraic multigrid method for queueing networks. Int J Comput Math 70:539–552
41. Ching W (1997) Circulant preconditioners for failure prone manufacturing systems. Linear Algebra Appl 266:161–180
42. Ching W (1997) Markov modulated poisson processes for multi-location inventory problems. Int J Product Econ 53:217–223
43. Ching W (1998) Iterative methods for manufacturing systems of two stations in Tandem. Appl Math Lett 11:7–12
44. Ching W (2000) Circulant preconditioning for unreliable manufacturing systems with batch arrivals. Int J Appl Math 4:11–21
45. Ching W (2001) Machine repairing models for production systems. Int J Product Econ 70:257–266
46. Ching W (2001) Iterative methods for queuing and manufacturing systems. Springer monographs in mathematics. Springer, London
47. Ching W (2001) Markovian approximation for manufacturing systems of unreliable machines in Tandem. Int J Naval Res Logist 48:65–78
48. Ching W (2003) Iterative methods for queuing systems with batch arrivals and negative customers. BIT 43:285–296

49. Ching W, Chan R, Zhou X (1997) Circulant preconditioners for Markov modulated poisson processes and their applications to manufacturing systems. SIAM J Matrix Anal Appl 18:464–481
50. Ching W, Fung E, Ng M (2002) A multivariate Markov chain model for categorical data sequences and its applications in demand predictions. IMA J Manag Math 13:187–199
51. Ching W, Fung E, Ng M (2003) A higher-order Markov model for the Newsboy's problem. J Oper Res Soc 54:291–298
52. Ching W, Fung E, Ng M (2004) Higher-order Markov chain models for categorical data sequences. Int J Naval Res Logist 51:557–574
53. Ching W, Fung E, Ng M (2004) Building higher-order Markov chain models with EXCEL. Int J Math Educ Sci Technol 35:921–932
54. Ching W, Fung E, Ng M (2008) Higher-order multivariate Markov chains and their applications. Lin Algebra Appl 428(2-3):492–507
55. Ching W, Li L, Li T, Zhang S (2007) A new multivariate Markov chain model with applications to sales demand forecasting. In: Proceedings of the international conference on industrial engineering and systems management, Beijing, 2007, (in CD-ROM)
56. Ching W, Loh A (2003) Iterative methods for flexible manufacturing systems. J Appl Math Comput 141:553–564
57. Ching W, Ng M (2003) Recent advance in data mining and modeling. World Scientific, Singapore
58. Ching W, Ng M (2004) Building simple hidden Markov models. Int J Math Educ Sci Eng 35:295–299
59. Ching W, Ng M (2006) Markov chains: models, algorithms and applications. International series on operations research and management science. Springer, New York, 224 Pages
60. Ching W, Ng M, Fung E (2003) In: Liu J, Cheung Y, Yin H (eds) Higher-order Hidden Markov models with applications to DNA sequences. IDEAL2003, Lecture notes in computer science, vol 2690. Springer, Berlin, pp 535–539
61. Ching W, Ng M, So M (2004) Customer migration, campaign budgeting, revenue estimation: the elasticity of Markov decision process on customer lifetime value. Electron Int J Adv Model Optim 6(2):65–80
62. Ching W, Ng M, Wong K (2003) Higher-order Markov decision process and its applications in customer lifetime values. In: The 32nd international conference on computers and industrial engineering, vol 2, Limerick, Ireland, pp 821–826, 2003
63. Ching W, Ng M, Wong K (2004) Hidden Markov models and its applications to customer relationship management. IMA J Manag Math 15:13–24
64. Ching W, Ng M, Wong K, Atlman E (2004) Customer lifetime value: a stochastic programming approach. J Oper Res Soc 55:860–868
65. Ching W, Ng M, Yuen W (2003) In: Kumar V, Gavrilova M, Tan C, L'Ecuyer P (eds) A direct method for block-toeplitz systems with applications to re-manufacturing systems. Lecture notes in computer science 2667, vol 1. Springer, Berlin, pp 912–920
66. Ching W, Siu T, Fung E, Ng M, Li W (2007) Interactive hidden Markov models and their applications. IMA J Manag Math 18:85–97
67. Ching W, Ng M, Yuen W (2005) A direct method for solving block-toeplitz with near-circulant-block systems with applications to hybrid manufacturing systems. J Numer Linear Algebra Appl 12:957–966
68. Ching W, Ng M, Zhang S (2005) On computation with higher-order Markov chain. In: Zhang W, Chen Z, Glowinski R, Tong W (eds) Current trends in high performance computing and its applications proceedings of the international conference on high performance computing and applications, 8–10 August 2004. Springer, Shanghai, China, pp 15–24
69. Ching W, Siu T, Li L, Jiang H, Li T, Li W (2009) An improved parsimonious multivariate Markov chain model for credit risk. J Credit Risk 5:1–25
70. Ching W, Siu T, Li L, Li T, Li W (2009) Modeling default data via an interactive hidden Markov model. Comput Econ 34:1–19

71. Ching W, Yuen W (2002) Iterative methods for re-manufacturing systems. Int J Appl Math 9:335–347
72. Ching W, Yuen W, Loh A (2003) An inventory model with returns and lateral transshipments. J Oper Res Soc 54:636–641
73. Ching W, Yuen W, Ng M, Zhang S (2006) A linear programming approach for solving optimal advertising policy. IMA J Manag Math 17:83–96
74. Ching W, Zhang S, Ng M (2007) On multi-dimensional Markov chain models. Pac J Optim 3:235–243
75. Cho D, Parlar M (1991) A survey of maintenance models for multi-unit systems. Eur J Oper Res 51:1–23
76. Chvatal V (1983) Linear programming. Freeman, New York
77. Cooper R (1972) Introduction to queueing theory. Macmillan, New York
78. Cutland N, Kopp P, Willinger W (1995) From discrete to continuous stochastic calculus. Stochast Stochast Rep 52(3–4):173–192
79. Davis P (1979) Circulant matrices. Wiley, New York
80. Dekker R, Fleischmann M, Inderfurth K, van Wassenhove L (2004) Reverse logistics: quantitative models for closed-loop supply chains. Springer, Berlin
81. Duffie D, Pan J (1997) An overview of value at risk. J Derivat 4(3):7–49
82. DuWors R, Haines G (1990) Event history analysis measure of brand loyalty. J Market Res 27:485–493
83. Elliott R (1993) New finite-dimensional filters and smoothers for noisily observed Markov chains. IEEE Trans Inform Theory 39(1):265–271
84. Elliott R, Aggoun L, Moore J (1994) Hidden Markov models: estimation and control. Springer, New York
85. Elliott R, Chan L, Siu T (2005) Option pricing and Esscher transform under regime switching. Ann Finance 1(4):423–432
86. Elliott R, van der Hoek J (1997) An application of hidden Markov models to asset allocation problems. Finance Stochast 3:229–238
87. Elliott R, Hunter W, Jamieson B (1998) Drift and volatility estimation in discrete time. J Econ Dyn Contr 22:209–218
88. Elliott R, Liew C, Siu T (2011) On filtering and estimation of a threshold stochastic volatility model. Appl Math Comput 218(1):61–75
89. Elliott R, Malcolm W, Tsoi A (2003) Robust parameter estimation for asset price models with Markov modulated volatilities. J Econ Dyn Contr 27(8):1391–1409
90. Elliott R, Mamon R (2003) An interest rate model with a Markovian mean-reverting level. Quant Finance 2:454–458
91. Elliott R, Miao H (2006) Stochastic volatility model with filtering. Stochast Anal Appl 24:661–683
92. Elliott R, Miao H (2009) VaR and expected shortfall: a non-normal regime switching framework. Quant Finance 9:747–755
93. Elliott R, Siu T (2009) On Markov-modulated exponential-affine bond price formulae. Appl Math Finance 16:1–15
94. Elliott R, Siu T (2010) On risk minimizing portfolios under a Markovian regime-switching black-scholes economy. Ann Oper Res 176:271–291
95. Elliott R, Siu T (2011) Pricing and hedging contingent claims with regime switching risk. Commun Math Sci 9:477–498
96. Elliott R, Siu T (2011) A stochastic differential game for optimal investment of an insurer with regime switching. Quant Finance 11:365–380
97. Elliott R, Siu T, Fung E (2011) Filtering a nonlinear stochastic volatility model. Nonlinear Dynam 67(2):1295–1313
98. Fang S, Puthenpura S (1993) Linear optimization and extensions. Prentice-Hall, New Jersey
99. Fleischmann M (2001) Quantitative models for reverse logistics. Lecture notes in economics and mathematical systems, vol 501. Springer, Berlin

100. Garfield E (1955) Citation indexes for science: a new dimension in documentation through association of ideas. Science 122:108–111
101. Garfield E (1972) Citation analysis as a tool in journal evaluation. Science 178:471–479
102. Gelenbe E (1989) Random neural networks with positive and negative signals and product solution. Neural Comput 1:501–510
103. Gelenbe E (1991) Product form networks with negative and positive customers. J Appl Probab 28:656–663
104. Gelenbe E, Glynn P, Sigman K (1991) Queues with negative arrivals. J Appl Probab 28:245–250
105. Giampieri G, Davis M, Crowder M (2005) Analysis of default data using hidden Markov models. Quant Finance 5:27–34
106. Goldberg D (1989) Genetic algorithm in search, optimization, and machine learning. Addison-Wesley, Reading
107. Goldfeld S, Quandt R (1973) A Markov model for switching regressions. J Econ 1:3–16
108. Golub G, van Loan C (1989) Matrix computations. John Hopkins University Press, Baltimore
109. Gowda K, Diday E (1991) Symbolic clustering using a new dissimilarity measure. Pattern Recogn 24(6):567–578
110. Guo X (2001) Information and option pricing. Quant Finance 1:38–44
111. Häggström (2002) Finite Markov chains and algorithmic applications. London mathematical society, Student Texts 52. Cambridge University Press, Cambridge
112. Hamilton J (1989) A new approach to the economic analysis of nonstationary time series and the business cycle. Econometrica 57:357–384
113. Haveliwala T, Kamvar S (2003) The second eigenvalue of the google matrix. Technical Report, Stanford University
114. Haveliwala T, Kamvar S, Jeh G (2003) An analytical comparison of approaches to personalizing PageRank. Stanford University (preprint)
115. Haveliwala T, Kamvar S, Klein D, Manning C, Golub G (2003) Computing PageRank using power extrapolation. Technical Report, Stanford University
116. He J, Xu J, Yao X (2000) Solving equations by hybrid evolutionary computation techniques. IEEE Trans Evol Comput 4:295–304
117. Hestenes M, Stiefel E (1952) Methods of conjugate gradients for solving linear systems. J Res Nat Bureau Stand 49:490–436
118. Heyman D (1977) Optimal disposal policies for single-item inventory system with returns. Naval Res Logist 24:385–405
119. Horn R, Johnson C (1985) Matrix analysis. Cambridge University Press, Cambridge
120. Huang J, Ng M, Ching W, Cheung D, Ng J (2001) A cube model for web access sessions and cluster analysis. In: Kohavi R, Masand B, Spiliopoulou M and Srivastava J (eds) WEBKDD 2001, Workshop on mining web log data across all customer touch points. The seventh ACM SIGKDD international conference on knowledge discovery and data mining. Lecture notes in computer science. Springer, Berlin, pp 47–58
121. Hughes A, Wang P (1995) Media selection for database marketers. J Direct Market 9:79–84
122. Inderfurth K, van der Laan E (2001) Leadtime effects and policy improvement for stochastic inventory control with remanufacturing. Int J Product Econ 71:381–390
123. Jackson B (1985) Winning and keeping industrial customers. Lexington Books, Lexington, MA
124. Jain D, Singh S (2002) Customer lifetime value research in marketing: a review and future directions. J Interact Market 16:34–46
125. Jarrow R, Turnbull S (1995) Pricing derivatives on financial securities subject to credit risk. J Finance 50:53–85
126. Joachims T, Freitag D, Mitchell T (1997) WebWatch: a tour guide for the world wide web. In: Proceedings of the fifteenth international joint conference on artificial intelligence IJCAI 97, pp 770–775
127. Jorion P (1997) Value at risk: the new benchmark for controlling market risk. McGraw-Hill, New York

128. Kahan W (1958) Gauss-Seidel methods of solving large systems of linear equations. Ph.D. thesis, University of Toronto
129. Kamvar S, Haveliwala T, Golub G (2004) Adaptive methods for the computation of PageRank. Linear Algebra Appl 386:51–65
130. Kaufman L (1982) Matrix methods for queueing problems. SIAM J Scient Stat Comput 4:525–552
131. Kiesmüller G, van der Laan E (2001) An inventory model with dependent product demands and returns. Int J Product Econ 72:73–87
132. Kincaid D, Cheney W (2002) Numerical analysis: mathematics of scientific computing, 3rd edn. Books/Cole Thomson Learning, CA
133. Klose A, Speranze G, Van Wassenhove LN (2002) Quantitative approaches to distribution logistics and supply chain management. Springer, Berlin
134. Kotler P, Armstrong G (1995) Principle of marketing, 7th edn. Prentice Hall, Englewood Cliffs
135. Koski T (2001) Hidden Markov models for bioinformatics. Kluwer, Dordrecht
136. van der Laan E (2003) An NPV and AC analysis of a stochastic inventory system with joint manufacturing and remanufacturing. Int J Product Econ 81–82:317–331
137. van der Laan E, Dekker R, Salomon M, Ridder A (2001) An (s,Q) inventory model with re-manufacturing and disposal. Int J Product Econ 46:339–350
138. van der Laan E, Salomon M (1997) Production planning and inventory control with re-manufacturing and disposal. Eur J Oper Res 102:264–278
139. Langville A, Meyer C (2005) A survey of eigenvector methods for web information retrieval. SIAM Rev 47:135–161
140. Langville A, Meyer C (2006) Google's PageRank and beyond : the science of search engine rankings. Princeton University Press, Princeton
141. Latouche G, Ramaswami V (1999) Introduction to matrix analytic methods in stochastic modeling. SIAM, Pennsylvania
142. Latouche G, Taylor P (2002) Matrix-analytic methods theory and applications. World Scientific, Singapore
143. Lee P (1997) Bayesian statistics: an introduction. Edward Arnold, London
144. Leonard K (1975) Queueing systems. Wiley, New York
145. Leung H, Ching W, Leung I (2008) A stochastic optimization model for consecutive promotion. Qual Technol Quant Manag 5:403–414
146. Li W, Kwok M (1989) Some results on the estimation of a higher order Markov chain. Department of Statistics, The University of Hong Kong
147. Lieberman H (1995) Letizia: an agent that assists web browsing. In: Proceedings of the fourteenth international joint conference on artificial intelligence IJCAI 95, pp 924–929
148. Lilien L, Kotler P, Moorthy K (1992) Marketing models. Prentice Hall, Englewood Cliffs
149. Lim J (1990) Two-dimensional signal and image processing. Prentice Hall, Englewood Cliffs
150. Lin C, Lin Y (2007) Robust analysis on promotion duration for two competitive brands. J Oper Res Soc 1:1–8
151. Logan J (1981) A structural model of the higher-order Markov process incorporating reversion effects. J Math Sociol 8: 75–89
152. Lu L, Ching W, Ng M (2004) Exact algorithms for singular tridiagonal systems with applications to Markov chains. J Appl Math Comput 159:275–289
153. Luo S, Tsoi A (2007) Filtering of hidden weak Markov chain: discrete range obervations. In: Mamon R, Elliott R (eds) Hidden Markov models in finance. Springer, New York, pp 101–119
154. McCormick S (1987) Multigrid methodst. Society for Industrial and Applied Mathematics, Philadelphia
155. MacDonald I, Zucchini W (1997) Hidden Markov and other models for discrete-valued time series. Chapman & Hall, London
156. Mamon R, Elliott R (2007) Hidden Markov models in finance. Springer's international series in operations research and management science, vol 104. Springer, New York

157. McNeil A, Frey R, Embrechts P (2005) Quantitative risk management: concepts, techniques and tools. Princeton University Press, Princeton
158. Merton R (1973) The theory of rational option pricing. Bell J Econ Manag Sci 4:141–183
159. Mesak H (2003) On deriving and validating comparative statics of a symmetric model of advertising competition. Comput Oper Res 30:1791–1806
160. Mesak H, Calloway J (1999) Hybrid subgames and copycat games in a pulsing model of advertising competition. J Oper Res Soc 50:837–849
161. Mesak H, Means T (1998) Modelling advertising budgeting and allocation decisions using modified multinomial logit market share models. J Oper Res Soc 49:1260–1269
162. Mesak H, Zhang H (2001) Optimal advertising pulsation policies: a dynamic programming approach. J Oper Res Soc 11:1244–1255
163. Muckstadt J (2005) Analysis and algorithms for service parts supply chains. Springer, New York
164. Muckstadt J, Isaac M (1981) An analysis of single item inventory systems with returns. Int J Naval Res Logist 28:237–254
165. Nahmias S (1981) Managing repairable item inventory systems: a review in TIMS studies. Manag Sci 16:253–277
166. Naik V (1993) Option valuation and hedging strategies with jumps in the volatility of asset returns. J Finance 48:1969–1984
167. Neuts M (1981) Matrix-geometric solutions in stochastic models: an algorithmic approach. Johns Hopkins University Press, Baltimore
168. Neuts M (1995) Algorithmic probability : a collection of problems. Chapman & Hall, London
169. Page L, Brin S, Motwani R, Winograd T (1998) The PageRank citation ranking: bring order to the web, technical report. Stanford University, Stanford
170. Pfeifer P, Carraway R (2000) Modeling customer relationships as Markov chain. J Int Market 14:43–55
171. Pliska S (2003) Introduction to mathematical finance: discrete time models. Blackwell, Oxford
172. Priestley M (1981) Spectral anslysis and time series. Academic, New York
173. Puterman M (1994) Markov decision processes: discrete stochastic dynamic programming. Wiley, New York
174. Quandt R (1958) The estimation of parameters of linear regression system obeying two separate regimes. J Am Stat Assoc 55:873–880
175. Rabiner L (1989) A tutorial on hidden Markov models and selected applications in speech recognition. Proc IEEE 77:257–286
176. Raftery A (1985) A model for high-order Markov chains. J Roy Stat Soc Ser B 47:528–539
177. Raftery A, Tavare S (1994) Estimation and modelling repeated patterns in high order Markov chains with the mixture transition distribution model. J Appl Stat 43:179–199
178. Richter K (1994) An EOQ repair and waste disposal. In: Proceedings of the eighth international working seminar on production economics. Igls/Innsbruch, Austria, pp 83–91
179. Robert C (2001) The Bayesian choice. Springer, New York
180. Robinson L (1990) Optimal and approximate policies in multi-period, multi-location inventory models with transshipments. Oper Res 38:278–295
181. Ross S (2000) Introduction to probability models, 7th edn. Academic, New York
182. Saad Y (2003) Iterative methods for sparse linear systems society for industrial and applied mathematics, 2nd edn. SIAM, Philadelphia
183. Shahabi C, Faisal A, Kashani F, Faruque J (2000) INSITE: a tool for real time knowledge discovery from users web navigation. In: Proceedings of VLDB2000, Cairo, Egypt
184. Siu T (2008) A game theoretic approach to option valuation under Markovian regime-switching models. Insur Math Econ 42:1146–1158
185. Siu T (2010) A Markov regime switching marked point process for short rate analysis with credit risk. Int J Stochast Anal 2010, Article ID 870516, 18 pages
186. Siu T (2010) Bond pricing under a Markovian regime-switching jump-augmented vasicek model via stochastic flows. Appl Math Comput 216:3184–3190

187. Siu T, Ching W, Fung E, Ng M (2005) On a multivariate Markov chain model for credit risk measurement. Quantit Finance 5:543–556
188. Siu T, Ching W, Fung E, Ng M (2005) Extracting information from spot interest rates and credit ratings using double higher-order hidden Markov models. Comput Econ 26:251–284
189. Siu T, Ching E, Fung E, Ng M, Li X (2009) A higher-order Markov-switching model for risk measurement. Comput Math Appl 58:1–10
190. Sonneveld P (1989) A fast Lanczos-type solver for non-symmetric linear systems. SIAM J Scientif Comput 10:36–52
191. Steward W (1994) Introduction to the numerical solution of Markov chain. Princeton University Press, Princeton
192. Tai A, Ching W, Cheung W (2005) On computing prestige in a network with negative relations. Int J Appl Math Sci 2:56–64
193. Tavana M, Rappaport J (1997) Optimal allocation of arrivals to a collection of parallel workstations. Int J Oper Product Manag 17:305–325
194. Teunter R, van der Laan E (2002) On the non-optimality of the average cost approach for inventory models with remanufacturing. Int J Product Econ 79:67–73
195. Thierry M, Salomon M, van Nunen J, van Wassenhove L (1995) Strategic issues in product recovery management. Calif Manag Rev 37:114–135
196. Trench W (1964) An algorithm for the inversion of finite Toeplitz matrices. SIAM J Appl Math 12:515–522
197. Tong H (1978) On a threshold model. In: Pattern recognition and signal processing. Sijthoff and Noordhoff, The Netherlands
198. Tong H (1983) Threshold models in non-linear time series analysis. Springer, Berlin
199. Tong H (1990) Non-linear time series: a dynamical system approach. Oxford University Press, Oxford
200. Tong H, Lim K (1980) Threshold autoregression, limit cycles and cyclical data. J Roy Stat Soc Ser B, Methodol 42:245–292
201. Tsoi A (2007) Discrete time weak Markov term structure model (preprint)
202. Tsoi A (2007) Discrete time reversal and duality of weak Markov chain (preprint)
203. Varga R (1963) Matrix iterative analysis. Prentice-Hall, New Jersey
204. Viterbi A (1967) Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Trans Inform Theory 13:260–269
205. Wang Y (1992) Approximation k^{th}-order two-state Markov chains. J Appl Probab 29:861–868
206. Wasserman S, Faust K (1994) Social network analysis: methods and applications. Cambridge Univeristy Press, Cambridge
207. White D (1993) Markov decision processes. Wiley, Chichester
208. Winston W (1994) Operations research: applications and algorithms, Belmont Calif., 3rd edn. Duxbury, North Scituate
209. Woo W, Siu T (2004) A dynamic binomial expansion technique for credit risk measurement: a bayesian filtering approach. Appl Math Finance 11:165–186
210. Yang Q, Huang Z, Ng M (2003) A data cube model for prediction-based web prefetching. J Intell Inform Syst 20:11–30
211. Yin G, Zhou X (2004) Markowitz's mean-variance portfolio selection with regime switching: from discrete-time models to their continuous-time limits. IEEE Trans Automat Contr 49:349–360
212. Young T, Calvert T (1974) Classification, estimation and pattern recognition. American Elsevier Publishing Company, INC., New York
213. Yuen W, Ching W, Ng M (2004) A hybrid algorithm for queueing systems. CALCOLO 41:139–151
214. Yuen W, Ching W, Ng M (2005) A hybrid algorithm for solving the PageRank. In: Zhang W, Chen Z, Glowinski R, Tong W (eds) Current trends in high performance computing and its applications proceedings of the international conference on high performance computing and applications, 8–10 August 2004. Springer, Shanghai, China, pp 257–264

215. Zhou X, Yin G (2003) Markowitz's mean-variance portfolio selection with regime switching: a continuous time model. SIAM J Contr Optim 42:1466–1482
216. Zhu D, Ching W (2011) A note on the stationary property of high-dimensional Markov chain models. Int J Pure Appl Math 66:321–330

Index

W.-K. Ching et al., *Markov Chains*, International Series in Operations Research & Management Science 189, DOI 10.1007/978-1-4614-6312-2,