Notation and abbreviations

> Since the underlying mathematical ideas are the important quantities, no notation should be adhered to slavishly. It is all a question of who is master.
>
> Bellman (1960, p. 82)

> [M]any writers have acted as though they believe that the success of the Box–Jenkins models is largely due to the use of the acronyms.
>
> Granger (1982)

Notation

Although notation is defined as it is introduced, it may also be helpful to list here the most common meanings of symbols, and the pages on which they are introduced. Matrices and vectors are denoted by bold type. Transposition of matrices and vectors is indicated by the prime symbol: $'$. All vectors are row vectors unless indicated otherwise.

Symbol	Meaning	Page
$A_n(\kappa)$	$I_n(\kappa)/I_0(\kappa)$	243
$\mathbf{B}_t$	$\mathbf{\Gamma P}(x_t)$	82
C_t	state occupied by Markov chain at time t	14
$\mathbf{C}^{(t)}$	$(C_1, C_2, \ldots, C_t)$	14
$\mathbf{e}_i$	$(0, \ldots, 0, 1, 0, \ldots, 0)$, with the 1 in the ith position	93
$\{g_t\}$	parameter process of a stochastic volatility model	263
I_n	modified Bessel function of the first kind of order n	242
l	log-likelihood	19
L or L_T	likelihood	19, 34
log	logarithm to the base e	
m	number of states in a Markov chain,	16
	or number of components in a mixture	7
$\mathbb{N}$	the set of all positive integers	
N_t	nutrient level	298
$N(\cdot; \mu, \sigma^2)$	distribution function of general normal distribution	264
$n(\cdot; \mu, \sigma^2)$	density of general normal distribution	264
p_i	probability mass or density function in state i	31
$\mathbf{P}(x)$	diagonal matrix with ith diagonal element $p_i(x)$	32
$\mathbb{R}$	the set of all real numbers	

T	length of a time series	34
$\mathbf{U}$	square matrix with all elements equal to 1	18
$\mathbf{u}(t)$	vector $(\Pr(C_t = 1), \ldots, \Pr(C_t = m))$	16
$u_i(t)$	$\Pr(C_t = i)$, i.e. ith element of $\mathbf{u}(t)$	32
w_t	$\boldsymbol{\alpha}_t \mathbf{1}' = \sum_i \alpha_t(i)$	48
X_t	observation at time t, or just tth observation	30
$\mathbf{X}^{(t)}$	$(X_1, X_2, \ldots, X_t)$	30
$\mathbf{X}^{(-t)}$	$(X_1, \ldots, X_{t-1}, X_{t+1}, \ldots, X_T)$	82
$\mathbf{X}_a^b$	$(X_a, X_{a+1}, \ldots, X_b)$	67
$\boldsymbol{\alpha}_t$	(row) vector of forward probabilities	38
$\alpha_t(i)$	forward probability, i.e. $\Pr(\mathbf{X}^{(t)} = \mathbf{x}^{(t)}, C_t = i)$	65
$\boldsymbol{\beta}_t$	(row) vector of backward probabilities	66
$\beta_t(i)$	backward probability, i.e. $\Pr(\mathbf{X}_{t+1}^T = \mathbf{x}_{t+1}^T \mid C_t = i)$	66
$\boldsymbol{\Gamma}$	transition probability matrix of Markov chain	15
γ_{ij}	(i, j) element of $\boldsymbol{\Gamma}$; probability of transition from state i to state j in a Markov chain	15
$\boldsymbol{\delta}$	stationary or initial distribution of Markov chain, or vector of mixing probabilities	17 7
$\boldsymbol{\phi}_t$	vector of forward probabilities, normalized to have sum equal to 1, i.e. $\boldsymbol{\alpha}_t / w_t$	48
Φ	distribution function of standard normal distribution	
$\boldsymbol{\Omega}$	t.p.m. driving state switches in an HSMM	165
$\mathbf{1}$	(row) vector of ones	18

Abbreviations

ACF	autocorrelation function
AIC	Akaike's information criterion
BIC	Bayesian information criterion
CDLL	complete-data log-likelihood
c.o.d.	change of direction
c.v.	coefficient of variation
HM(M)	hidden Markov (model)
HSMM	hidden semi-Markov model
MC	Markov chain
MCMC	Markov chain Monte Carlo
ML	maximum likelihood
MLE	maximum likelihood estimator or estimate
p.d.f.	probability density function
p.m.f.	probability mass function
qq-plot	quantile–quantile plot
SV	stochastic volatility
t.p.m.	transition probability matrix

PART I

Model structure, properties and methods

CHAPTER 1

Preliminaries: mixtures and Markov chains

1.1 Introduction

Hidden Markov models (HMMs) are models in which the distribution that generates an observation depends on the state of an underlying and unobserved Markov process. They provide flexible general-purpose models for univariate and multivariate time series, especially for discrete-valued series, including categorical series and series of counts.

The purposes of this chapter are to provide a brief and informal introduction to HMMs, and to their many potential uses, and then to discuss two topics that will be fundamental in understanding the structure of such models. In Section 1.2 we give an account of (finite) mixture distributions, because the marginal distribution of a hidden Markov model is a mixture distribution. Then, in Section 1.3, we introduce Markov chains, which provide the underlying 'parameter process' of a hidden Markov model.

Consider, as an example, the series of annual counts of major earthquakes (i.e. magnitude 7 and above) for the years 1900–2006, both inclusive, displayed in Table 1.1 and Figure 1.1.* For this series, the application of standard models such as autoregressive moving-average (ARMA) models would be inappropriate, because such models are based on the normal distribution. Instead, the usual model for unbounded counts is the Poisson distribution, but, as will be demonstrated later, the series displays considerable overdispersion relative to the Poisson distribution, and strong positive serial dependence. A model consisting of independent Poisson random variables would therefore for two reasons also be inappropriate. An examination of Figure 1.1 suggests that there may be some periods with a low rate of earthquakes, and some with a relatively high rate. HMMs, which allow the probability distribution of each observation to depend on the unobserved (or 'hidden') state of a Markov chain, can accommodate both overdispersion and serial dependence. We

* These data were downloaded from `http://neic.usgs.gov/neis/eqlists` on 25 July 2007. Note, however, that the US Geological Survey undertook a systematic review, and there may be minor differences between the information now available and the data we present here.

Table 1.1 *Number of major earthquakes (magnitude 7 or greater) in the world, 1900–2006; to be read across rows.*

13	14	8	10	16	26	32	27	18	32	36	24	22	23	22	18	25	21	21	14
8	11	14	23	18	17	19	20	22	19	13	26	13	14	22	24	21	22	26	21
23	24	27	41	31	27	35	26	28	36	39	21	17	22	17	19	15	34	10	15
22	18	15	20	15	22	19	16	30	27	29	23	20	16	21	21	25	16	18	15
18	14	10	15	8	15	6	11	8	7	18	16	13	12	13	20	15	16	12	18
15	16	13	15	16	11	11													

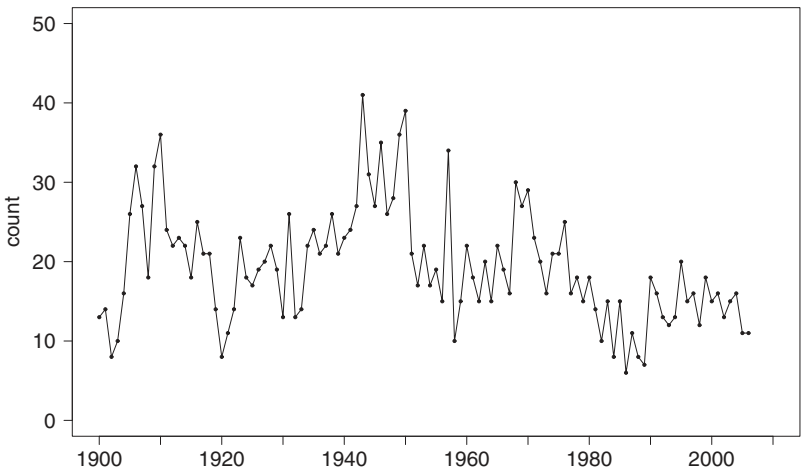


Figure 1.1 *Number of major earthquakes (magnitude 7 or greater) in the world, 1900–2006.*

shall use this series of earthquake counts as a running example in Part I of the book, in order to illustrate the fitting of a Poisson–HMM and many other aspects of that model.

HMMs have been used for at least three decades in signal-processing applications, especially in the context of automatic speech recognition, but interest in their theory and application has expanded to other fields, for example:

- all kinds of recognition – face, gesture, handwriting, signature;
- bioinformatics – biological sequence analysis;
- environment – rainfall, earthquakes, wind direction;

- finance – series of daily returns;
- biophysics – ion channel modelling;
- ecology – animal behaviour.

Attractive features of HMMs include their simplicity, their general mathematical tractability, and specifically the fact that the likelihood is relatively straightforward to compute. The main aim of this book is to illustrate how HMMs can be used as general-purpose models for time series.

Following this preliminary chapter, the book introduces what we shall call the **basic HMM**: basic in the sense that it is univariate, is based on a homogeneous Markov chain, and has neither trend nor seasonal variation. The observations may be either discrete- or continuous-valued, but we initially ignore information that may be available on covariates. We focus on the following issues:

- parameter estimation (Chapters 3 and 4);
- point and interval forecasting (Chapter 5);
- decoding, i.e. estimating the sequence of hidden states (Chapter 5);
- model selection, model checking and outlier detection (Chapter 6).

In Chapter 7 we give one example of the Bayesian approach to inference. In Chapter 8 we give examples of how several **R** packages can be used to fit basic HMMs to data and to decode.

In Part II we discuss the many possible extensions of the basic HMM to a wider range of models. These include HMMs for series with trend and seasonal variation, methods to include covariate information from other time series, multivariate models of various types, HMM approximations to hidden semi-Markov models and to models with continuous-valued state process, and HMMs for longitudinal data.

Part III of the book offers fairly detailed applications of HMMs to time series arising in a variety of subject areas. These are intended to illustrate the theory covered in Parts I and II, and also to demonstrate the versatility of HMMs. Indeed, so great is the variety of HMMs that it is hard to imagine this diversity being exhaustively covered by any single software package. In some applications the model needs to accommodate some special features of the time series, which makes it necessary to write one's own code. We have found the computing environment **R** (Ihaka and Gentleman, 1996; R Core Team, 2015) to be particularly convenient for this purpose.

Many of the chapters contain exercises, some theoretical and some practical. Because one always learns more about models by applying them in practice, and because some aspects of the theory of HMMs are covered only in these exercises, we regard these as an important part of

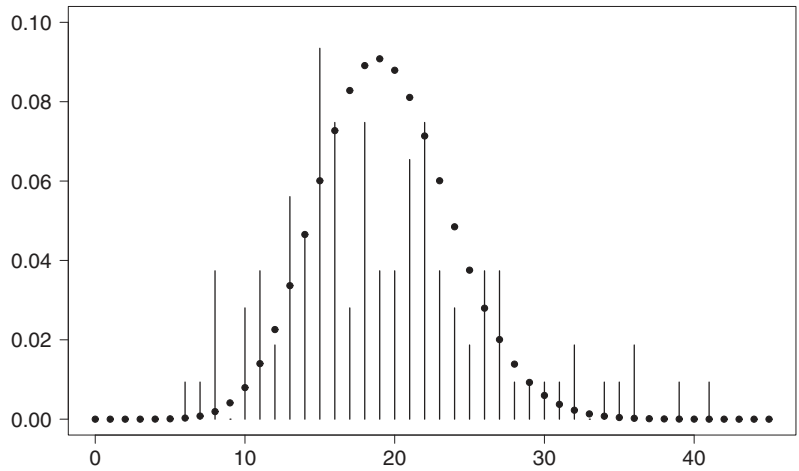


Figure 1.2 *Major earthquakes, 1900–2006: bar plot of relative frequencies of counts, and fitted Poisson distribution.*

the book. As regards the practical exercises, our strategy has been to give examples of **R** functions for some important but simple cases, and to encourage readers to learn to write their own code, initially just by modifying the functions given in Appendix A.

1.2 Independent mixture models

1.2.1 Definition and properties

Consider again the series of earthquake counts displayed in Figure 1.1. A standard model for unbounded counts is the Poisson distribution, with its probability function $p(x) = e^{-\lambda}\lambda^x/x!$ and the property that the variance equals the mean. However, for the earthquakes series the sample variance, $s^2 \approx 52$, is much larger than the sample mean, $\bar{x} \approx 19$, which indicates strong overdispersion relative to the Poisson distribution. The lack of fit is confirmed by Figure 1.2, which displays the fitted Poisson distribution and a bar plot of the relative frequencies of the counts.

One method of dealing with overdispersed observations with a bimodal or (more generally) multimodal distribution is to use a mixture model. Mixture models are designed to accommodate unobserved heterogeneity in the population; that is, the population may consist of unobserved groups, each having a distinct distribution for the observed variable.

Consider, for example, the distribution of the number, X, of packets of cigarettes bought by the customers of a supermarket. The customers can be divided into groups, for example, non-smokers, occasional smokers, and regular smokers. Now even if the number of packets bought by customers within each group were Poisson-distributed, the distribution of X would not be Poisson; it would be overdispersed relative to the Poisson, and maybe even multimodal.

Analogously, suppose that each count in the earthquakes series is generated by one of two Poisson distributions, with means λ_1 and λ_2, where the choice of mean is determined by some other random mechanism which we call the **parameter process**. Suppose also that λ_1 is selected with probability δ_1 and λ_2 with probability $\delta_2 = 1 - \delta_1$. We shall see later in this chapter that the variance of the resulting distribution exceeds the mean by $\delta_1\delta_2(\lambda_1 - \lambda_2)^2$. If the parameter process is a series of independent random variables, the counts are also independent, hence the term 'independent mixture'.

In general, an independent mixture distribution consists of a finite number, say m, of component distributions and a 'mixing distribution' which selects from these components. The component distributions may be either discrete or continuous. In the case of two components, the mixture distribution depends on two probability or density functions:

component	1	2
probability or density function	$p_1(x)$	$p_2(x)$.

To specify the component, one needs a discrete random variable C which performs the mixing:

$$C = \begin{cases} 1 & \text{with probability } \delta_1 \\ 2 & \text{with probability } \delta_2 = 1 - \delta_1. \end{cases}$$

The structure of that process for the case of two continuous component distributions is illustrated in Figure 1.3. In that example one can think of C as the outcome of tossing a coin with probability 0.75 of 'heads': if the outcome is 'heads', then $C = 1$ and an observation is drawn from p_1; if it is 'tails', then $C = 2$ and an observation is drawn from p_2. We suppose that we do not know the value C, that is, which of p_1 or p_2 was active when the observation was generated.

The extension to m components is straightforward. Let $\delta_1, \ldots, \delta_m$ denote the probabilities assigned to the different components, and let $p_1, \ldots, p_m$ denote their probability or density functions. Let X denote the random variable which has the mixture distribution. In the discrete

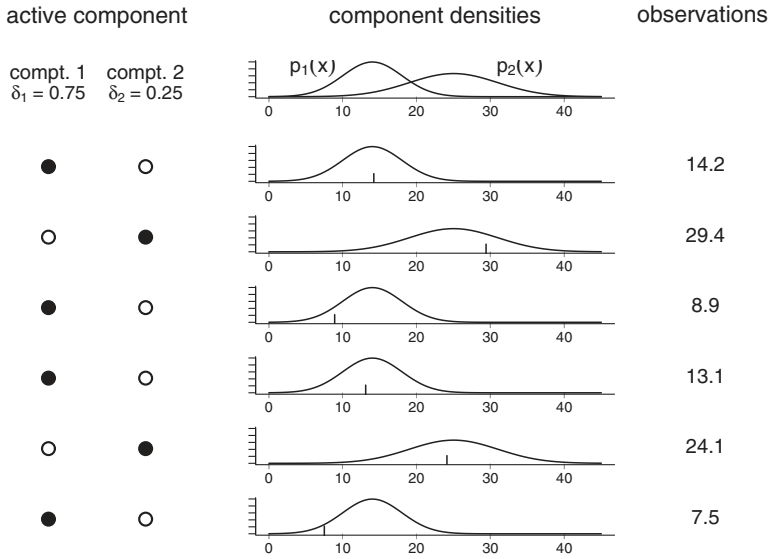


Figure 1.3 *Process structure of a two-component mixture distribution. From top to bottom, the states are* $1, 2, 1, 1, 2, 1$. *The corresponding component distributions are shown in the middle. The observations are generated from the active component density.*

case the probability function of X is given by

$$\begin{aligned} p(x) &= \sum_{i=1}^{m} \Pr(X = x \mid C = i) \Pr(C = i) \\ &= \sum_{i=1}^{m} \delta_i p_i(x). \end{aligned}$$

The continuous case is analogous. The expectation of the mixture can be given in terms of the expectations of the component distributions. Letting Y_i denote the random variable with probability function p_i, we have

$$\mathrm{E}(X) = \sum_{i=1}^{m} \Pr(C = i) \, \mathrm{E}(X \mid C = i) = \sum_{i=1}^{m} \delta_i \, \mathrm{E}(Y_i).$$

The same result holds for a mixture of continuous distributions.

More generally, for a mixture the kth moment about the origin is

simply a linear combination of the kth moments of its components Y_i:

$$\mathrm{E}(X^k) = \sum_{i=1}^{m} \delta_i \, \mathrm{E}(Y_i^k), \quad k = 1, 2, \ldots.$$

Note that the analogous result does not hold for central moments. In particular, the variance of X is not a linear combination of the variances of its components Y_i. Exercise 1 asks the reader to prove that, in the two-component case, the variance of the mixture is given by

$$\mathrm{Var}(X) = \delta_1 \mathrm{Var}(Y_1) + \delta_2 \mathrm{Var}(Y_2) + \delta_1 \delta_2 \big(\mathrm{E}(Y_1) - \mathrm{E}(Y_2)\big)^2.$$

1.2.2 Parameter estimation

The estimation of the parameters of a mixture distribution is often performed by maximum likelihood (ML). The likelihood of a mixture model with m components is given, for both discrete and continuous cases, by

$$L(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_m, \delta_1, \ldots, \delta_m \mid x_1, \ldots, x_n) = \prod_{j=1}^{n} \sum_{i=1}^{m} \delta_i p_i(x_j, \boldsymbol{\theta}_i). \qquad (1.1)$$

Here $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_m$ are the parameter vectors of the component distributions, $\delta_1, \ldots, \delta_m$ are the mixing parameters, totalling 1, and $x_1, \ldots, x_n$ are the n observations. Thus, in the case of component distributions each specified by one parameter, $2m - 1$ independent parameters have to be estimated. Except perhaps in special cases, analytic maximization of such a likelihood is not possible, but it is in general straightforward to evaluate it fast; see Exercise 3. Numerical maximization will be illustrated here by considering the case of a mixture of Poisson distributions.

Suppose that $m = 2$ and the two components are Poisson-distributed with means λ_1 and λ_2. Let δ_1 and δ_2 be the mixing parameters (with $\delta_1 + \delta_2 = 1$). The mixture distribution p is then given by

$$p(x) = \delta_1 \frac{\lambda_1^x e^{-\lambda_1}}{x!} + \delta_2 \frac{\lambda_2^x e^{-\lambda_2}}{x!}.$$

Since $\delta_2 = 1 - \delta_1$, there are only three parameters to be estimated: λ_1, λ_2 and δ_1. The likelihood is

$$L(\lambda_1, \lambda_2, \delta_1 \mid x_1, \ldots, x_n) \quad = \quad \prod_{i=1}^{n} \left(\delta_1 \frac{\lambda_1^{x_i} e^{-\lambda_1}}{x_i!} + (1 - \delta_1) \frac{\lambda_2^{x_i} e^{-\lambda_2}}{x_i!} \right).$$

The analytic maximization of L with respect to λ_1, λ_2 and δ_1 would be awkward, as L is the product of n factors, each of which is a sum. First taking the logarithm and then differentiating does not greatly simplify matters either. Therefore parameter estimation is more conveniently carried out by direct numerical maximization of the likelihood (or its logar-

ithm), although the EM algorithm is a commonly used alternative; see, for example, McLachlan and Peel (2000) or Frühwirth-Schnatter (2006). (We shall in Chapter 4 discuss the EM algorithm more fully in the context of the estimation of HMMs.) A useful **R** package for estimation in mixture models is `flexmix` (Leisch, 2004). However, it is straightforward to write one's own **R** code to evaluate, and then maximize, mixture likelihoods in simple cases.

This log-likelihood can then be maximized by using (for example) the **R** function `nlm`. However, the parameters $\boldsymbol{\delta}$ and $\boldsymbol{\lambda}$ are constrained by $\sum_{i=1}^{m} \delta_i = 1$ and (for $i = 1, \ldots, m$) $\delta_i > 0$ and $\lambda_i > 0$. It is therefore necessary to reparametrize if one wishes to use an unconstrained optimizer such as `nlm`. One possibility is to maximize the likelihood with respect to the $2m - 1$ unconstrained 'working parameters'

$$\eta_i = \log \lambda_i \quad (i = 1, \ldots, m)$$

and

$$\tau_i = \log \left(\frac{\delta_i}{1 - \sum_{j=2}^{m} \delta_j} \right) \quad (i = 2, \ldots, m).$$

One recovers the original 'natural parameters' via

$$\lambda_i = e^{\eta_i} \quad (i = 1, \ldots, m),$$

$$\delta_i = \frac{e^{\tau_i}}{1 + \sum_{j=2}^{m} e^{\tau_j}} \quad (i = 2, \ldots, m),$$

and $\delta_1 = 1 - \sum_{j=2}^{m} \delta_i$. The following code implements the above ideas in order to fit a mixture of four Poisson distributions to the earthquake counts. The results are given for $m = 1$, 2, 3, 4 in Table 1.2.

```
# Function to compute -log(likelihood)
mllk <- function(wpar,x){ zzz <- w2n(wpar)
        -sum(log(outer(x,zzz$lambda,dpois)%*%zzz$delta)) }

# Function to transform natural to working parameters
n2w  <- function(lambda,delta)log(c(lambda,delta[-1]/(1-sum(delta[-1]))))

# Function to transform working to natural parameters
w2n  <- function(wpar){m <- (length(wpar)+1)/2
        lambda <- exp(wpar[1:m])
        delta  <- exp(c(0,wpar[(m+1):(2*m-1)]))
return(list(lambda=lambda,delta=delta/sum(delta))) }

# Read data, specify starting values, minimize -log(likelihood),
# and transform to natural parameters
x        <- read.table("earthquakes.txt")[,2] # Set your own path.
wpar     <- n2w(c(10,20,25,30),c(1,1,1,1)/4)
w2n(nlm(mllk,wpar,x)$estimate)
```

Notice how, in this code, the use of the function `outer` makes it possible to evaluate a Poisson mixture log-likelihood in a single compact expression rather than a loop. But if the distributions being mixed were distributions with more than one parameter (e.g. normal), a slightly different approach would be needed.

1.2.3 Unbounded likelihood in mixtures

There is one aspect of mixtures of continuous distributions that differs from the discrete case and is worth highlighting. It is this: it can happen that, in the vicinity of certain parameter combinations, the likelihood is unbounded. For instance, in the case of a mixture of normal distributions, the likelihood becomes arbitrarily large if one sets a component mean equal to one of the observations and allows the corresponding variance to tend to zero. The problem has been extensively discussed in the literature on mixture models, and there are those who would say that, if the likelihood is thus unbounded, the ML estimates simply 'do not exist'; see, for instance, Scholz (2006, p. 4630).

The source of the problem, however, is just the use of densities rather than probabilities in the likelihood; it would not arise if one were to replace each density value in a likelihood by the probability of the interval corresponding to the recorded value. (For example, an observation recorded as '12.4' is associated with the interval $[12.35, 12.45)$.) In the context of independent mixtures one replaces the expression

$$\prod_{j=1}^{n} \sum_{i=1}^{m} \delta_i p_i(x_j, \boldsymbol{\theta}_i)$$

for the likelihood (see equation (1.1)) by the **discrete likelihood**

$$L = \prod_{j=1}^{n} \sum_{i=1}^{m} \delta_i \int_{a_j}^{b_j} p_i(x, \boldsymbol{\theta}_i)\, \mathrm{d}x, \tag{1.2}$$

where the interval (a_j, b_j) consists of those values which, if observed, would be recorded as x_j. This simply amounts to acknowledging explicitly the interval nature of all supposedly continuous observations. More generally, the discrete likelihood of observations on a set of random variables $X_1, X_2, \ldots, X_n$ is a probability of the form $\Pr(a_t < X_t < b_t, \text{ for all } t)$. We use the term **continuous likelihood** for the joint density evaluated at the observations.

Another way of avoiding the problem is to impose a lower bound on the variances and search for the best local maximum subject to that bound. It can happen, though, that one is fortunate enough to avoid the likelihood 'spikes' when searching for a local maximum; in this respect

Table 1.2 *Poisson independent mixture models fitted to the earthquakes series. The number of components is m, the mixing probabilities are denoted by δ_i, and the component means by λ_i. The maximized likelihood is L.*

Model	i	δ_i	λ_i	$-\log L$	Mean	Variance
$m = 1$	1	1.000	19.364	391.9189	19.364	19.364
$m = 2$	1	0.676	15.777	360.3690	19.364	46.182
	2	0.324	26.840			
$m = 3$	1	0.278	12.736	356.8489	19.364	51.170
	2	0.593	19.785			
	3	0.130	31.629			
$m = 4$	1	0.093	10.584	356.7337	19.364	51.638
	2	0.354	15.528			
	3	0.437	20.969			
	4	0.116	32.079			
observations					19.364	51.573

good starting values can help. The phenomenon of unbounded likelihood does not arise for discrete-valued observations because the likelihood is in that case a probability and thereby bounded by 0 and 1.

For a thorough account of the unbounded likelihood 'problem', see Liu, Wu and Meeker (2015). Liu *et al.* use the terms 'density-approximation likelihood' and 'correct likelihood' for what we call the continuous likelihood and discrete likelihood, respectively.

1.2.4 Examples of fitted mixture models

Mixtures of Poisson distributions

If one uses `nlm` to fit a mixture of m Poisson distributions ($m = 1, 2, 3, 4$) to the earthquakes data, one obtains the results displayed in Table 1.2. Notice that there is a very clear improvement in likelihood resulting from the addition of a second component, and very little improvement from addition of a fourth – apparently insufficient to justify the additional two parameters. Section 6.1 will discuss the model selection problem in more detail. Figure 1.4 presents a histogram of the observed counts and the

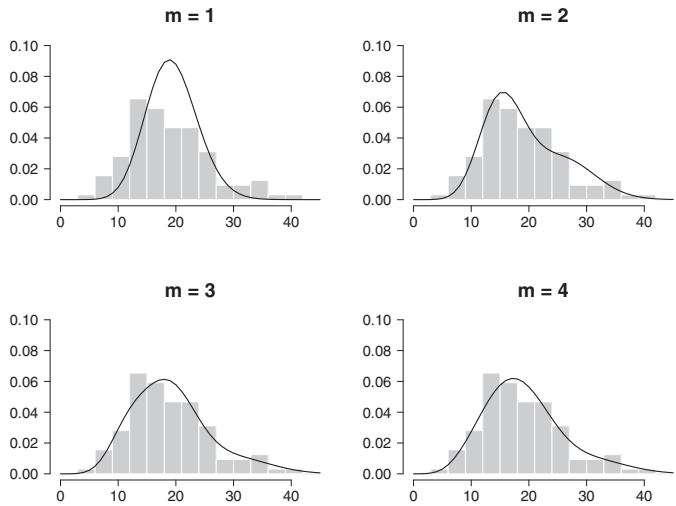


Figure 1.4 *Earthquakes data: histogram of counts, compared to mixtures of one, two, three and four Poisson distributions.*

four models fitted. It is clear that the mixtures fit the observations much better than does a single Poisson distribution, and visually the three- and four-state models seem adequate. The better fit of the mixtures is also evident from the variances of the four models as presented in Table 1.2. In computing the means and variances of the models we have used $\mathrm{E}(X) = \sum_i \delta_i \lambda_i$ and $\mathrm{Var}(X) = \mathrm{E}(X^2) - (\mathrm{E}(X))^2$, with $\mathrm{E}(X^2) = \sum_i \delta_i(\lambda_i + \lambda_i^2)$. For comparison we also used the **R** package `flexmix` to fit the same four models. The results corresponded closely except in the case of the four-component model, where the highest likelihood value that we found by `flexmix` was 356.7759 and the component means differed somewhat.

Note, however, that the above discussion ignores the possibility of serial dependence in the earthquakes data, a point we shall take up in Chapter 2.

A mixture of normal distributions

As a very simple example of the fitting of an independent mixture of normal distributions, consider the data presented in Table 8.1 of Hastie, Tibshirani and Friedman (2009, p. 273); see our Table 1.3. Hastie *et al.* use the EM algorithm to fit a mixture model with two normal components.

Table 1.3 *Data of Hastie* et al. *(2009), plus two mixture models. The first model was fitted by direct numerical maximization in* **R**, *the second is the model fitted by EM by Hastie* et al.

−0.39	0.12	0.94	1.67	1.76	2.44	3.72	4.28	4.92	5.53
0.06	0.48	1.01	1.68	1.80	3.25	4.12	4.60	5.28	6.22

i	δ_i	μ_i	σ_i^2	$-\log L$
1	0.4454	4.656	0.8188	38.9134
2	0.5546	1.083	0.8114	
1	0.454	4.62	0.87	
2	0.546	1.06	0.77	

Our two-component model, fitted by direct numerical maximization of the log-likelihood in **R**, has log-likelihood −38.9134, and is also displayed in Table 1.3. (Here we used the continuous likelihood, i.e. the joint density of the observations, not the discrete likelihood.) The parameter estimates are close to those given by Hastie *et al.*, but not identical.

1.3 Markov chains

We now introduce Markov chains, a second building-block of hidden Markov models. Our treatment is restricted to those few aspects of discrete-time Markov chains that we need. Thus, although we shall make passing reference to properties such as irreducibility and aperiodicity, we shall not dwell on such technical issues. For a general account of the topic, see Grimmett and Stirzaker (2001, Chapter 6), or Feller's classic text (Feller, 1968).

1.3.1 Definitions and example

A sequence of discrete random variables $\{C_t : t \in \mathbb{N}\}$ is said to be a (discrete-time) **Markov chain** (MC) if, for all $t \in \mathbb{N}$, it satisfies the **Markov property**

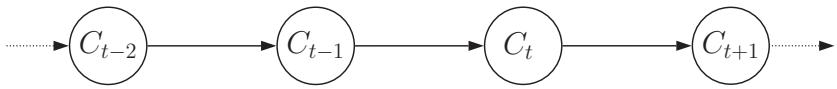
$$\Pr(C_{t+1} \mid C_t, \ldots, C_1) = \Pr(C_{t+1} \mid C_t).$$

That is, conditioning on the 'history' of the process up to time t is equivalent to conditioning only on the most recent value C_t. For compactness we define $\mathbf{C}^{(t)}$ as the history $(C_1, C_2, \ldots, C_t)$, in which case the Markov

property can be written as

$$\Pr(C_{t+1} \mid \mathbf{C}^{(t)}) = \Pr(C_{t+1} \mid C_t).$$

The Markov property can be regarded as a first relaxation of the assumption of independence. The random variables $\{C_t\}$ are dependent in a specific way that is mathematically convenient, as displayed in the following directed graph in which the past and the future are dependent only through the present.



Important quantities associated with a Markov chain are the conditional probabilities called **transition probabilities**:

$$\Pr(C_{s+t} = j \mid C_s = i).$$

If these probabilities do not depend on s, the Markov chain is called **homogeneous**, otherwise non-homogeneous. Unless there is an explicit indication to the contrary, we shall assume that the Markov chain under discussion is homogeneous, in which case the transition probabilities will be denoted by

$$\gamma_{ij}(t) = \Pr(C_{s+t} = j \mid C_s = i).$$

Notice that the notation $\gamma_{ij}(t)$ does not involve s. The matrix $\mathbf{\Gamma}(t)$ is defined as the matrix with (i, j) element $\gamma_{ij}(t)$.

An important property of all finite state-space homogeneous Markov chains is that they satisfy the **Chapman–Kolmogorov equations**:

$$\mathbf{\Gamma}(t + u) = \mathbf{\Gamma}(t)\,\mathbf{\Gamma}(u).$$

The proof requires only the definition of conditional probability and the application of the Markov property: this is Exercise 10. The Chapman–Kolmogorov equations imply that, for all $t \in \mathbb{N}$,

$$\mathbf{\Gamma}(t) = \mathbf{\Gamma}(1)^t;$$

that is, the matrix of t-step transition probabilities is the tth power of $\mathbf{\Gamma}(1)$, the matrix of one-step transition probabilities. The matrix $\mathbf{\Gamma}(1)$, which will be abbreviated as $\mathbf{\Gamma}$, is a square matrix of probabilities with row sums equal to 1:

$$\mathbf{\Gamma} = \begin{pmatrix} \gamma_{11} & \cdots & \gamma_{1m} \\ \vdots & \ddots & \vdots \\ \gamma_{m1} & \cdots & \gamma_{mm} \end{pmatrix},$$

where (throughout this text) m denotes the number of states of the Markov chain. The statement that the row sums are equal to 1 can be written as $\mathbf{\Gamma 1}' = \mathbf{1}'$; that is, the column vector $\mathbf{1}'$ is a right eigenvector of $\mathbf{\Gamma}$ and corresponds to eigenvalue 1. We shall refer to $\mathbf{\Gamma}$ as the (one-step) **transition probability matrix** (t.p.m.). Many authors use instead the term 'transition matrix'; we avoid that term because of possible confusion with a matrix of transition counts, or a matrix of transition intensities.

The **unconditional probabilities** $\Pr(C_t = j)$ of a Markov chain being in a given state at a given time t are often of interest. We denote these by the row vector

$$\mathbf{u}(t) = (\Pr(C_t = 1), \ldots, \Pr(C_t = m)), \quad t \in \mathbb{N}.$$

We refer to $u(1)$ as the **initial distribution** of the Markov chain. To deduce the distribution at time $t+1$ from that at t we postmultiply by the transition probability matrix $\mathbf{\Gamma}$:

$$\mathbf{u}(t+1) = \mathbf{u}(t)\mathbf{\Gamma}. \tag{1.3}$$

The proof of this statement is left as an exercise.

Example. Imagine that the sequence of rainy and sunny days is such that each day's weather depends only on the previous day's, and the transition probabilities are given by the following table.

	day $t+1$	
day t	rainy	sunny
rainy	0.9	0.1
sunny	0.6	0.4

That is, if today is rainy, the probability that tomorrow will be rainy is 0.9; if today is sunny, that probability is 0.6. The weather is then a two-state homogeneous Markov chain, with t.p.m. $\mathbf{\Gamma}$ given by

$$\mathbf{\Gamma} = \begin{pmatrix} 0.9 & 0.1 \\ 0.6 & 0.4 \end{pmatrix}.$$

Now suppose that today (time 1) is a sunny day. This means that the distribution of today's weather is

$$\mathbf{u}(1) = \big(\Pr(C_1 = 1), \Pr(C_1 = 2)\big) = \big(0, 1\big).$$

The distribution of the weather of tomorrow, the day after tomorrow, and so on, can be calculated by repeatedly postmultiplying $\mathbf{u}(1)$ by $\mathbf{\Gamma}$,

the t.p.m.:

$$\begin{aligned}\mathbf{u}(2) &= \big(\Pr(C_2=1),\Pr(C_2=2)\big)=\mathbf{u}(1)\mathbf{\Gamma}=(0.6,0.4),\\ \mathbf{u}(3) &= \big(\Pr(C_3=1),\Pr(C_3=2)\big)=\mathbf{u}(2)\mathbf{\Gamma}=(0.78,0.22),\ \text{etc.}\end{aligned}$$

1.3.2 Stationary distributions

A Markov chain with transition probability matrix $\mathbf{\Gamma}$ is said to have **stationary distribution** $\boldsymbol{\delta}$ (a row vector with non-negative elements) if $\boldsymbol{\delta}\mathbf{\Gamma}=\boldsymbol{\delta}$ and $\boldsymbol{\delta}\mathbf{1}'=1$. The first of these requirements expresses the stationarity, the second is the requirement that $\boldsymbol{\delta}$ is indeed a probability distribution. For instance, the Markov chain with t.p.m. given by

$$\mathbf{\Gamma}=\begin{pmatrix}1/3 & 1/3 & 1/3\\ 2/3 & 0 & 1/3\\ 1/2 & 1/2 & 0\end{pmatrix}$$

has as stationary distribution $\boldsymbol{\delta}=\frac{1}{32}(15,9,8)$.

Since $\mathbf{u}(t+1)=\mathbf{u}(t)\mathbf{\Gamma}$, a Markov chain started from its stationary distribution will continue to have that distribution at all subsequent time points, and we shall refer to such a process as a **stationary Markov chain**. It is perhaps worth stating that this assumes more than merely homogeneity. Homogeneity alone would not be sufficient to render the Markov chain a stationary process, and we prefer to reserve the adjective 'stationary' for homogeneous Markov chains that have the additional property that the initial distribution $\mathbf{u}(1)$ is the stationary distribution and are therefore stationary processes. Not all authors use this terminology, however; see, for example, McLachlan and Peel (2000, p. 328), who use the word 'stationary' of a Markov chain where we would say 'homogeneous'.

An irreducible (homogeneous, discrete-time, finite state-space) Markov chain has a unique, strictly positive, stationary distribution. Note that although the technical assumption of irreducibility is needed for this conclusion, aperiodicity is not; see Grimmett and Stirzaker (2001, Lemma 6.3.5 on p. 225 and Theorem 6.4.3 on p. 227).

If, however, one does add the assumption of aperiodicity, it follows that a unique limiting distribution exists, and is precisely the stationary distribution; see Feller (1968, p. 394). Since we shall always assume aperiodicity and irreducibility, the terms 'limiting distribution' and 'stationary distribution' are for our purposes synonymous.

A general result that can conveniently be used to compute a stationary distribution (see Exercise 9(a)) is as follows. The vector $\boldsymbol{\delta}$ with non-negative elements is a stationary distribution of the Markov chain with

t.p.m. $\mathbf{\Gamma}$ if and only if

$$\boldsymbol{\delta}(\mathbf{I}_m - \mathbf{\Gamma} + \mathbf{U}) = \mathbf{1},$$

where $\mathbf{1}$ is a row vector of ones, $\mathbf{I}_m$ is the $m \times m$ identity matrix, and $\mathbf{U}$ is the $m \times m$ matrix of ones. Alternatively, a stationary distribution can be found by deleting one of the equations in the system $\boldsymbol{\delta}\mathbf{\Gamma} = \boldsymbol{\delta}$ and replacing it by $\sum_i \delta_i = 1$.

1.3.3 Autocorrelation function

We shall have occasion, for example in Section 2.2.3 and in Exercise 4(f) in Chapter 2, to compare the autocorrelation function (ACF) of a hidden Markov model with that of its underlying Markov chain $\{C_t\}$, on the states 1, 2, ..., m. We assume that these states are quantitative and not merely categorical. The ACF of $\{C_t\}$, assumed stationary and irreducible, may be obtained as follows.

Firstly, defining $\mathbf{v} = (1, 2, \ldots, m)$ and $\mathbf{V} = \text{diag}(1, 2, \ldots, m)$, we have, for all non-negative integers k,

$$\text{Cov}(C_t, C_{t+k}) = \boldsymbol{\delta}\mathbf{V}\mathbf{\Gamma}^k\mathbf{v}' - (\boldsymbol{\delta}\mathbf{v}')^2; \tag{1.4}$$

the proof is Exercise 11. Secondly, if $\mathbf{\Gamma}$ is diagonalizable, and its eigenvalues (other than 1) are denoted by $\omega_2, \omega_3, \ldots, \omega_m$, then $\mathbf{\Gamma}$ can be written as

$$\mathbf{\Gamma} = \mathbf{U}\mathbf{\Omega}\mathbf{U}^{-1},$$

where $\mathbf{\Omega}$ is $\text{diag}(1, \omega_2, \omega_3, \ldots, \omega_m)$ and the columns of $\mathbf{U}$ are corresponding right eigenvectors of $\mathbf{\Gamma}$. We then have, for non-negative integers k,

$$\begin{aligned} \text{Cov}(C_t, C_{t+k}) &= \boldsymbol{\delta}\mathbf{V}\mathbf{U}\mathbf{\Omega}^k\mathbf{U}^{-1}\mathbf{v}' - (\boldsymbol{\delta}\mathbf{v}')^2 \\ &= \mathbf{a}\mathbf{\Omega}^k\mathbf{b}' - a_1 b_1 \\ &= \sum_{i=2}^{m} a_i b_i \omega_i^k, \end{aligned}$$

where $\mathbf{a} = \boldsymbol{\delta}\mathbf{V}\mathbf{U}$ and $\mathbf{b}' = \mathbf{U}^{-1}\mathbf{v}'$. Hence $\text{Var}(C_t) = \sum_{i=2}^{m} a_i b_i$ and, for non-negative integers k,

$$\rho(k) \equiv \text{Corr}(C_t, C_{t+k}) = \sum_{i=2}^{m} a_i b_i \omega_i^k \Big/ \sum_{i=2}^{m} a_i b_i. \tag{1.5}$$

This is a weighted average of the kth powers of the eigenvalues ω_2, ω_3, ..., ω_m, and somewhat similar to the ACF of a Gaussian autoregressive process of order $m-1$. Note that equation (1.5) implies in the case $m = 2$ that $\rho(k) = \rho(1)^k$ for all non-negative integers k, and that $\rho(1)$ is the eigenvalue other than 1 of $\mathbf{\Gamma}$.

1.3.4 Estimating transition probabilities

If we are given a realization of a Markov chain, and wish to estimate the transition probabilities, one approach – but not the only one – is to find the transition counts and estimate the transition probabilities as relative frequencies. For instance, if the MC has three states and the observed sequence is

2332111112 3132332122 3232332222 3132332212 3232132232
3132332223 3232331232 3232331222 3232132123 3132332121,

then the matrix of transition counts is

$$(f_{ij}) = \begin{pmatrix} 4 & 7 & 6 \\ 8 & 10 & 24 \\ 6 & 24 & 10 \end{pmatrix},$$

where f_{ij} denotes the number of transitions observed from state i to state j. Since the number of transitions from state 2 to state 3 is 24, and the total number of transitions from state 2 is 8+10+24, a relative frequency estimate of γ_{23} is 24/42. The t.p.m. $\mathbf{\Gamma}$ is therefore plausibly estimated by

$$\begin{pmatrix} 4/17 & 7/17 & 6/17 \\ 8/42 & 10/42 & 24/42 \\ 6/40 & 24/40 & 10/40 \end{pmatrix}.$$

We shall now show that this is in fact the conditional ML estimate of $\mathbf{\Gamma}$, conditioned on the first observation.

Suppose, then, that we wish to estimate the m^2-m parameters γ_{ij} ($i \neq j$) of an m-state Markov chain $\{C_t\}$ from a realization $c_1, c_2, \ldots, c_T$. The likelihood conditioned on the first observation is

$$L = \prod_{i=1}^{m} \prod_{j=1}^{m} \gamma_{ij}^{f_{ij}}.$$

The log-likelihood is

$$l = \sum_{i=1}^{m} \left(\sum_{j=1}^{m} f_{ij} \log \gamma_{ij} \right) = \sum_{i=1}^{m} l_i \text{ (say)},$$

and we can maximize l by maximizing each l_i separately. Substituting $1 - \sum_{k \neq i} \gamma_{ik}$ for γ_{ii}, differentiating l_i with respect to an off-diagonal transition probability γ_{ij}, and equating the derivative to zero yields

$$0 = \frac{-f_{ii}}{1 - \sum_{k \neq i} \gamma_{ik}} + \frac{f_{ij}}{\gamma_{ij}} = -\frac{f_{ii}}{\gamma_{ii}} + \frac{f_{ij}}{\gamma_{ij}}.$$

Hence, unless a denominator is zero in the above equation, $f_{ij}\gamma_{ii} = f_{ii}\gamma_{ij}$, and so $\gamma_{ii} \sum_{j=1}^{m} f_{ij} = f_{ii}$. This implies that, at a maximum of the

likelihood,

$$\gamma_{ii} = f_{ii} \Big/ \sum_{j=1}^{m} f_{ij} \quad \text{and} \quad \gamma_{ij} = f_{ij}\gamma_{ii}/f_{ii} = f_{ij} \Big/ \sum_{j=1}^{m} f_{ij}.$$

(We could instead use Lagrange multipliers to express the constraints $\sum_{j=1}^{m} \gamma_{ij} = 1$ subject to which we seek to maximize the terms l_i and therefore the likelihood; see Exercise 12.)

The estimator $\widehat{\gamma}_{ij} = f_{ij} / \sum_{k=1}^{m} f_{ik}$ $(i, j = 1, \ldots, m)$ – which is just the empirical transition probability – is thereby seen to be a conditional ML estimator of γ_{ij}. This estimator of $\mathbf{\Gamma}$ satisfies the requirement that the row sums should equal 1.

The assumption of stationarity of the Markov chain was not used in the above derivation. If we wish to assume stationarity, we may use the unconditional likelihood. This is the conditional likelihood as above, multiplied by the stationary probability δ_{c_1}. The unconditional likelihood or its logarithm may then be maximized numerically, subject to non-negativity and row-sum constraints, in order to estimate the transition probabilities γ_{ij}. Bisgaard and Travis (1991) show in the case of a two-state Markov chain that, barring some extreme cases, the unconditional likelihood equations have a unique solution. For some non-trivial special cases of the two-state chain, they also derive explicit expressions for the unconditional maximum likelihood estimates (MLEs) of the transition probabilities. Since we use one such result later (in Section 17.3.1), we state it here.

Suppose the Markov chain $\{C_t\}$ takes the values 0 and 1, and that we wish to estimate the transition probabilities γ_{ij} from a sequence of observations in which there are f_{ij} transitions from state i to state j $(i, j = 0, 1)$, and $f_{11} > 0$ but $f_{00} = 0$. So in the observations a zero is always followed by a one. Define $c = f_{10} + (1 - c_1)$ and $d = f_{11}$. Then the unconditional MLEs of the transition probabilities are given by

$$\widehat{\gamma}_{01} = 1 \quad \text{and} \quad \widehat{\gamma}_{10} = \frac{-(1+d) + \big((1+d)^2 + 4c(c+d-1)\big)^{\frac{1}{2}}}{2(c+d-1)}. \tag{1.6}$$

1.3.5 Higher-order Markov chains

This section is somewhat specialized, and the material is used only in Section 10.3 and parts of Sections 17.3.2 and 19.2.2. It will therefore not interrupt the continuity greatly if the reader should initially omit this section.

In cases where observations on a process with finite state space appear not to satisfy the Markov property, one possibility that suggests itself is to use a higher-order Markov chain, that is, a model $\{C_t\}$ satisfying the

following generalization of the Markov property for some $l \geq 2$:

$$\Pr(C_t \mid C_{t-1}, C_{t-2}, \ldots) = \Pr(C_t \mid C_{t-1}, \ldots, C_{t-l}).$$

An account of such higher-order Markov chains may be found, for instance, in Lloyd (1980, Section 19.9). Although such a model is not in the usual sense a Markov chain (i.e. not a 'first-order' Markov chain), we can redefine the model in such a way as to produce an equivalent process which is. If we let $\mathbf{Y}_t = (C_{t-l+1}, C_{t-l+2}, \ldots, C_t)$, then $\{\mathbf{Y}_t\}$ is a first-order Markov chain on M^l, where M is the state space of $\{C_t\}$. Although some properties may be more awkward to establish, no essentially new theory is involved in analysing a higher-order Markov chain rather than a first-order one.

A *second-order* Markov chain, if stationary, is characterized by the transition probabilities

$$\gamma(i, j, k) = \Pr(C_t = k \mid C_{t-1} = j, C_{t-2} = i),$$

and has stationary bivariate distribution $u(j, k) = \Pr(C_{t-1} = j, C_t = k)$ satisfying

$$u(j,k) = \sum_{i=1}^{m} u(i,j)\gamma(i,j,k) \quad \text{and} \quad \sum_{j=1}^{m}\sum_{k=1}^{m} u(j,k) = 1.$$

For example, the most general stationary second-order Markov chain $\{C_t\}$ on the two states 1 and 2 is characterized by the following four transition probabilities:

$$\begin{aligned} a &= \Pr(C_t{=}2 \mid C_{t-1}{=}1, C_{t-2}{=}1), \\ b &= \Pr(C_t{=}1 \mid C_{t-1}{=}2, C_{t-2}{=}2), \\ c &= \Pr(C_t{=}1 \mid C_{t-1}{=}2, C_{t-2}{=}1), \\ d &= \Pr(C_t{=}2 \mid C_{t-1}{=}1, C_{t-2}{=}2). \end{aligned}$$

The process $\{\mathbf{Y}_t\} = \{(C_{t-1}, C_t)\}$ is then a first-order Markov chain, on the four states (1,1), (1,2), (2,1), (2,2), with transition probability matrix

$$\begin{pmatrix} 1-a & a & 0 & 0 \\ 0 & 0 & c & 1-c \\ 1-d & d & 0 & 0 \\ 0 & 0 & b & 1-b \end{pmatrix}. \tag{1.7}$$

Notice the structural zeros appearing in this matrix. It is not possible, for instance, to make a transition directly from $(2,1)$ to $(2,2)$; hence the zero in row 3 and column 4 in the t.p.m. (1.7). The parameters a, b, c and d are bounded by 0 and 1 but are otherwise unconstrained. The stationary distribution of $\{\mathbf{Y}_t\}$ is proportional to the vector

$$\big(b(1-d), ab, ab, a(1-c)\big),$$

from which it follows that the matrix $(u(j,k))$ of stationary bivariate probabilities for $\{C_t\}$ is

$$\frac{1}{b(1-d)+2ab+a(1-c)}\begin{pmatrix} b(1-d) & ab \\ ab & a(1-c) \end{pmatrix}.$$

The use of a general higher-order Markov chain (instead of a first-order one) increases the number of parameters of the model; a general Markov chain of order l on m states has $m^l(m-1)$ independent transition probabilities. Pegram (1980) and Raftery (1985a,b) have therefore proposed certain classes of parsimonious models for higher-order chains. Pegram's models have $m+l-1$ parameters, and those of Raftery $m(m-1)+l-1$. For $m=2$ the models are equivalent, but for $m>2$ those of Raftery are more general and can represent a wider range of dependence patterns and autocorrelation structures. In both cases an increase of one in the order of the Markov chain requires only one additional parameter.

Raftery's models, which he terms 'mixture transition distribution' (MTD) models, are defined as follows. The process $\{C_t\}$ takes values in $M=\{1,2,\ldots,m\}$ and satisfies

$$\Pr(C_t=j_0 \mid C_{t-1}=j_1,\ldots,C_{t-l}=j_l)=\sum_{i=1}^{l}\lambda_i\, q(j_i,j_0), \tag{1.8}$$

where $\sum_{i=1}^{l}\lambda_i=1$, and $\mathbf{Q}=(q(j,k))$ is an $m\times m$ matrix with non-negative entries and row sums equal to one, such that the right-hand side of equation (1.8) is bounded by zero and one for all j_0, j_1, …, $j_l\in M$. This last requirement, which generates m^{l+1} pairs of nonlinear constraints on the parameters, ensures that the conditional probabilities in equation (1.8) are indeed probabilities, and the condition on the row sums of $\mathbf{Q}$ ensures that the sum over j_0 of these conditional probabilities is one. Note that Raftery does not assume that the parameters λ_i are non-negative.

A variety of applications are presented by Raftery (1985a) and Raftery and Tavaré (1994). In several of the fitted models there are negative estimates of some of the coefficients λ_i. For further accounts of this class of models, see Haney (1993), Berchtold (2001), and Berchtold and Raftery (2002).

Azzalini and Bowman (1990) report the fitting of a second-order Markov chain model to the binary series they use to represent the lengths of successive eruptions of the Old Faithful geyser. Their analysis, and some alternative models, will be discussed in Chapter 17.

Exercises

1. (a) Let X be a random variable which is distributed as a (δ_1, δ_2)-mixture of two distributions with expectations μ_1, μ_2, and variances σ_1^2, σ_2^2, respectively, where $\delta_1 + \delta_2 = 1$.

   i. Show that $\mathrm{Var}(X) = \delta_1\sigma_1^2 + \delta_2\sigma_2^2 + \delta_1\delta_2(\mu_1 - \mu_2)^2$.

   ii. Show that a (non-trivial) mixture X of two Poisson distributions with distinct means is overdispersed, that is, $\mathrm{Var}(X) > \mathrm{E}(X)$.

   (b) Now suppose that X is a mixture of $m \geq 2$ distributions, with means μ_i and variances σ_i^2, for $i = 1, 2, \ldots, m$. The mixing distribution is $\boldsymbol{\delta}$.

   i. Show that

$$\mathrm{Var}(X) = \sum_{i=1}^{m} \delta_i\sigma_i^2 + \sum_{i<j} \delta_i\delta_j(\mu_i - \mu_j)^2.$$

   Hint: use either $\mathrm{Var}(X) = \mathrm{E}(X^2) - (\mathrm{E}(X))^2$ or the conditional variance formula,

$$\mathrm{Var}(X) = \mathrm{E}(\mathrm{Var}(X \mid C)) + \mathrm{Var}(\mathrm{E}(X \mid C)).$$

   ii. Describe the circumstances in which $\mathrm{Var}(X)$ equals the linear combination $\sum_{i=1}^{m} \delta_i\sigma_i^2$.

2. A zero-inflated Poisson distribution is sometimes used as a model for unbounded counts displaying an excessive number of zeros relative to the Poisson. Such a model is a mixture of two distributions: one is a Poisson and the other is identically zero.

   (a) Is it ever possible for such a model to display underdispersion relative to Poisson?

   (b) Now consider the zero-inflated binomial. Is it possible in such a model that the variance is less than the mean?

3. Brown and Buckley (2015, p. 308) consider a Poisson mixture likelihood of the form

$$L = \prod_{i=1}^{n} \sum_{j=1}^{k} w_j f(x_i \mid \mu_j).$$

   (Here $f(\cdot \mid \mu)$ denotes a Poisson probability function with mean μ.) They write that 'Even for moderate values of n and k, this takes a long time to evaluate as there are k^n terms when the inner sums are expanded', and do not pursue maximum likelihood estimation.

   Explain why it is in fact possible to evaluate L or its logarithm in computations which are of order kn rather than k^n.

4. (a) Write an **R** function to minimize minus the log-likelihood of a normal mixture model with m components, using the nonlinear minimizer `nlm`.

   Hint: first write a function to transform the parameters ($\boldsymbol{\delta}$ and the parameters of the m normal distributions) into unconstrained parameters. You will also need a function to reverse the transformation. (For the Poisson case, see the code on p. 10.)

   (b) Use your code to fit a mixture of two normals to the data appearing in Table 1.3, and compare your model with those displayed in that table.

5. Consider the following data, which appear in Lange (1995, 2002, 2004). (There they are quoted from Titterington, Smith and Makov (1985) and Hasselblad (1969), but the trail leads back via Schilling (1947) and Thorndike (1926) to Whitaker (1914), where in all eight similar data sets appear as Table XV on p. 67.)

   Here n_i denotes the number of days in 1910–1912 on which there appeared, in *The Times* of London, i death notices in respect of women aged 80 or over at death.

i	0	1	2	3	4	5	6	7	8	9
n_i	162	267	271	185	111	61	27	8	3	1

   (a) Use `nlm` or `optim` in **R** to fit a mixture of two Poisson distributions to these observations. (The parameter estimates reported by Lange (2002, p. 36; 2004, p. 151) are, in our notation: $\widehat{\delta}_1 = 0.3599$, $\widehat{\lambda}_1 = 1.2561$ and $\widehat{\lambda}_2 = 2.6634$.)

   (b) Fit also a single Poisson distribution to these data. Is a single Poisson distribution adequate as a model?

   (c) Fit a mixture of three Poisson distributions to these observations.

   (d) How many components do you think are necessary?

   (e) Repeat (a)–(d) for some of the other seven data sets of Whitaker.

6. Consider the series of weekly sales (in integer units) of a particular soap product in a supermarket, as shown in Table 1.4. The data were taken from a database† provided by the Kilts Center for Marketing, Graduate School of Business of the University of Chicago, at: `http://gsbwww.uchicago.edu/kilts/research/db/dominicks`. The product was 'Zest White Water 15 oz.', with code 3700031165, and the store number 67.

† That database is now at `http://research.chicagobooth.edu/kilts/marketing-databases/dominicks`.

Table 1.4 *Weekly sales of the soap product; to be read across rows.*

1	6	9	18	14	8	8	1	6	7	3	3	1	3	4	12	8	10	8	2
17	15	7	12	22	10	4	7	5	0	2	5	3	4	4	7	5	6	1	3
4	5	3	7	3	0	4	5	3	3	4	4	4	4	4	3	5	5	5	7
4	0	4	3	2	6	3	8	9	6	3	4	3	3	3	3	2	1	4	5
5	2	7	5	2	3	1	3	4	6	8	8	5	7	2	4	2	7	4	15
15	12	21	20	13	9	8	0	13	9	8	0	6	2	0	3	2	4	4	6
3	2	5	5	3	2	1	1	3	1	2	6	2	7	3	2	4	1	5	6
8	14	5	3	6	5	11	4	5	9	9	7	9	8	3	4	8	6	3	5
6	3	1	7	4	9	2	6	6	4	6	6	13	7	4	8	6	4	4	4
9	2	9	2	2	2	13	13	4	5	1	4	6	5	4	2	3	10	6	15
5	9	9	7	4	4	2	4	2	3	8	15	0	0	3	4	3	4	7	5
7	6	0	6	4	14	5	1	6	5	5	4	9	4	14	2	2	1	5	2
6	4																		

Fit Poisson mixture models with one, two, three and four components. How many components do you think are necessary?

7. Consider a stationary two-state Markov chain with transition probability matrix given by

$$\mathbf{\Gamma} = \begin{pmatrix} \gamma_{11} & \gamma_{12} \\ \gamma_{21} & \gamma_{22} \end{pmatrix}.$$

(a) Show that the stationary distribution is

$$(\delta_1, \delta_2) = \frac{1}{\gamma_{12} + \gamma_{21}}(\gamma_{21}, \gamma_{12}) \ .$$

(b) Consider the case

$$\mathbf{\Gamma} = \begin{pmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{pmatrix},$$

and the following two sequences of observations that are assumed to be generated by the above Markov chain.

Sequence 1:	1	1	1	2	2	1
Sequence 2:	2	1	1	2	1	1

Compute the probability of each of the sequences. Note that each sequence contains the same number of ones and twos. Why are these sequences not equally probable?

8. Consider a two-state Markov chain with transition probability matrix given by

$$\mathbf{\Gamma} = \begin{pmatrix} \gamma_{11} & \gamma_{12} \\ \gamma_{21} & \gamma_{22} \end{pmatrix}.$$

Show that the k-step transition probability matrix, $\mathbf{\Gamma}^k$, is given by

$$\mathbf{\Gamma}^k = \begin{pmatrix} \delta_1 & \delta_2 \\ \delta_1 & \delta_2 \end{pmatrix} + w^k \begin{pmatrix} \delta_2 & -\delta_2 \\ -\delta_1 & \delta_1 \end{pmatrix},$$

where $w = 1 - \gamma_{12} - \gamma_{21}$ and δ_1 and δ_2 are as defined in Exercise 7. (Hint: one way of showing this is to diagonalize the transition probability matrix, but there is a quicker way.)

9. (a) This is one of several possible approaches to finding the stationary distribution of a Markov chain, plundered from Grimmett and Stirzaker (2001, Exercise 6.6.5).

   Suppose $\mathbf{\Gamma}$ is the transition probability matrix of a (discrete-time, homogeneous) Markov chain on m states, and that $\boldsymbol{\delta}$ is a non-negative row vector with m components. Show that $\boldsymbol{\delta}$ is a stationary distribution of the Markov chain if and only if

   $$\boldsymbol{\delta}(\mathbf{I}_m - \mathbf{\Gamma} + \mathbf{U}) = \mathbf{1},$$

   where $\mathbf{1}$ is a row vector of ones, and $\mathbf{U}$ is an $m \times m$ matrix of ones.

   (b) Write an **R** function `statdist(gamma)` that computes the stationary distribution of the Markov chain with t.p.m. `gamma`.

   (c) Use your function to find stationary distributions corresponding to the following transition probability matrices. One of them should cause a problem!

   i. $\begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0 & 0.6 & 0.4 \\ 0.5 & 0 & 0.5 \end{pmatrix}$

   ii. $\begin{pmatrix} 0 & 1 & 0 \\ \frac{1}{3} & 0 & \frac{2}{3} \\ 0 & 1 & 0 \end{pmatrix}$

   iii. $\begin{pmatrix} 0 & 0.5 & 0 & 0.5 \\ 0.75 & 0 & 0.25 & 0 \\ 0 & 0.75 & 0 & 0.25 \\ 0.5 & 0 & 0.5 & 0 \end{pmatrix}$

   iv. $\begin{pmatrix} 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.5 & 0 \\ 0 & 0 & 0.25 & 0.75 \\ 0 & 0 & 0.5 & 0.5 \end{pmatrix}$

   v. $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 \\ 0 & 0.75 & 0 & 0.25 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

10. Prove the Chapman–Kolmogorov equations.

11. Prove equation (1.4).
12. Let the quantities a_i be non-negative, with $\sum_i a_i > 0$. Using a Lagrange multiplier, maximize $S = \sum_{i=1}^{m} a_i \log \delta_i$ over $\delta_i \geq 0$, subject to $\sum_i \delta_i = 1$. (Check the second- as well as the first-derivative condition.)
13. (This exercise is based on Example 2 of Bisgaard and Travis (1991).) Consider the following sequence of 21 observations, assumed to arise from a two-state (homogeneous) Markov chain:

    11101 10111 10110 11111 1.

    (a) Estimate the transition probability matrix by ML, conditional on the first observation.
    (b) Estimate the t.p.m. by unconditional ML (assuming stationarity of the Markov chain).
    (c) Use the **R** functions `contour` and `persp` to produce contour and perspective plots of the unconditional log-likelihood (as a function of the two off-diagonal transition probabilities).

14. Consider the following two transition probability matrices, neither of which is diagonalizable:

    (a)
    $$\mathbf{\Gamma} = \begin{pmatrix} 1/3 & 1/3 & 1/3 \\ 2/3 & 0 & 1/3 \\ 1/2 & 1/2 & 0 \end{pmatrix};$$

    (b)
    $$\mathbf{\Gamma} = \begin{pmatrix} 0.9 & 0.08 & 0 & 0.02 \\ 0 & 0.7 & 0.2 & 0.1 \\ 0 & 0 & 0.7 & 0.3 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

    In each case, write $\mathbf{\Gamma}$ in Jordan canonical form, and so find an explicit expression for the t-step transition probabilities (t =1, 2, ...).
15. Consider the following (very) short DNA sequence, taken from Singh (2003, p. 358):

    AACGT CTCTA TCATG CCAGG ATCTG

    (a) Fit a homogeneous Markov chain to these data by:
        i. maximizing the likelihood conditioned on the first observation;
        ii. assuming stationarity and maximizing the unconditional likelihood of all 25 observations.
    (b) Compare your estimates of the t.p.m. with each other and with the estimate displayed as Table 1 of Singh (2003, p. 360).

(c) Now repeat (a) for the following 50-nucleotide sequence, taken from Singh (2003, p. 367):

ATTAG GCACG CATTA TAATG GGCAC
CCGGA AATAA CCAGA GTTAC GGCCA.

16. Write an **R** function `rMC(n,m,gamma,delta=NULL)` that generates a series of length `n` from an `m`-state Markov chain with t.p.m. `gamma`. If the initial state distribution is given, then it should be used; otherwise the stationary distribution should be used as the initial distribution. (Use your function `statdist` from Exercise 9(b).)

CHAPTER 2

Hidden Markov models: definition and properties

2.1 A simple hidden Markov model

Consider again the observed earthquake series displayed in Figure 1.1 on p. 4. The observations are unbounded counts, making the Poisson distribution a natural choice to describe them, but their distribution is clearly overdispersed relative to the Poisson. We saw in Chapter 1 that this feature can be accommodated by using a mixture of Poisson distributions with means $\lambda_1, \lambda_2, \ldots, \lambda_m$. The choice of mean is made by a second random process, the parameter process. The mean λ_i is selected with probability δ_i, where $i = 1, 2, \ldots, m$ and $\sum_{i=1}^{m} \delta_i = 1$.

An independent mixture model will not do for the earthquake series because – by definition – it does not allow for the serial dependence in the observations. The sample autocorrelation function (ACF), displayed in Figure 2.1, clearly indicates that the observations are serially dependent. One way of allowing for serial dependence in the observations is to relax the assumption that the parameter process is serially independent. A simple and mathematically convenient way to do so is to assume that it is a Markov chain. The resulting model for the observations is called a Poisson–hidden Markov model, a simple example of the class of

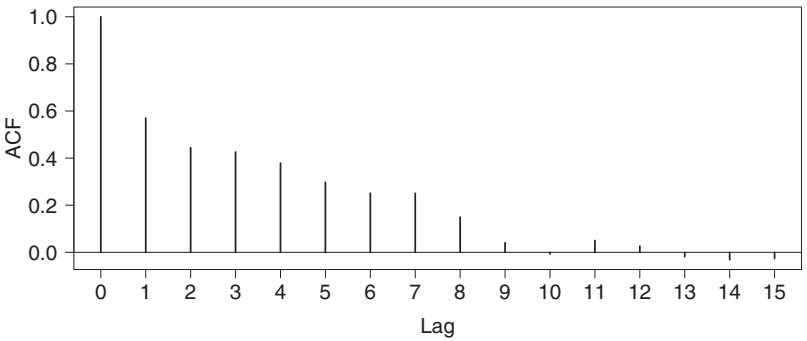


Figure 2.1 *Earthquakes series: sample autocorrelation function.*

models discussed in the rest of this book, namely hidden Markov models (HMMs).

We shall not give an account here of the (interesting) history of such models, but two valuable sources of information on HMMs that include accounts of the history are Ephraim and Merhav (2002) and Cappé *et al.* (2005).

2.2 The basics

2.2.1 Definition and notation

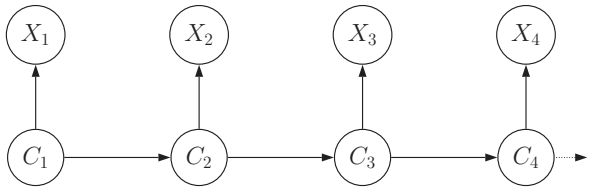


Figure 2.2 *Directed graph of basic HMM.*

A **hidden Markov model** $\{X_t : t \in \mathbb{N}\}$ is a particular kind of dependent mixture. With $\mathbf{X}^{(t)}$ and $\mathbf{C}^{(t)}$ representing the histories from time 1 to time t, one can summarize the simplest model of this kind by:

$$\Pr(C_t \mid \mathbf{C}^{(t-1)}) = \Pr(C_t \mid C_{t-1}), \quad t = 2, 3, \dots \tag{2.1}$$

$$\Pr(X_t \mid \mathbf{X}^{(t-1)}, \mathbf{C}^{(t)}) = \Pr(X_t \mid C_t), \quad t \in \mathbb{N}. \tag{2.2}$$

The model consists of two parts: firstly, an unobserved 'parameter process' $\{C_t : t = 1, 2, \dots\}$ satisfying the Markov property; and secondly, the 'state-dependent process' $\{X_t : t = 1, 2, \dots\}$, in which the distribution of X_t depends only on the current state C_t and not on previous states or observations. This structure is represented by the directed graph in Figure 2.2.

If the Markov chain $\{C_t\}$ has m states, we call $\{X_t\}$ an m-state HMM. Although it is the usual terminology in speech-processing applications, the name 'hidden Markov model' is by no means the only one used for such models or similar ones. For instance, Ephraim and Merhav (2002) argue for 'hidden Markov process', Leroux and Puterman (1992) use 'Markov-dependent mixture', and others use 'Markov-switching model' (especially for models with extra dependencies at the level of the observations X_t), 'models subject to Markov regime', 'Markov mixture model', or 'latent Markov model'. Bartolucci, Farcomeni and Pennoni (2013) use the term 'latent Markov model' specifically for models for longitudinal data, as opposed to single time series.

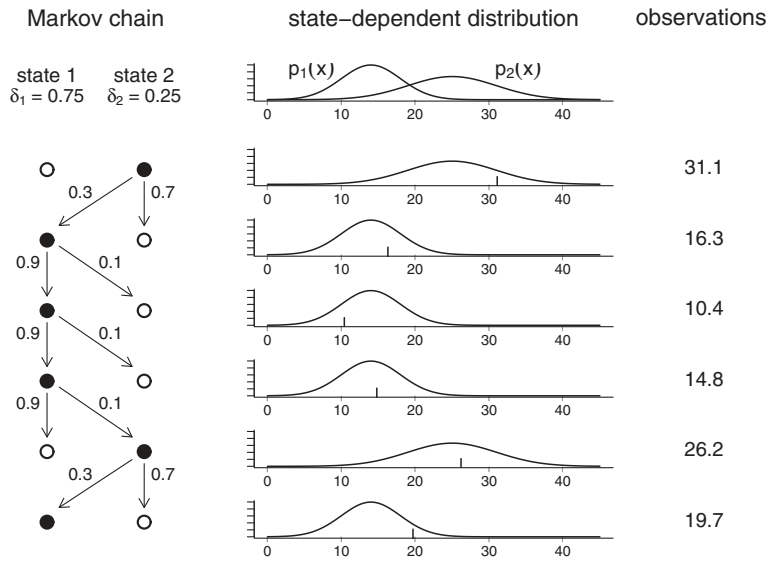


Figure 2.3 *Process generating the observations in a two-state HMM. The chain followed the path* $2, 1, 1, 1, 2, 1$*, as indicated on the left. The corresponding state-dependent distributions are shown in the middle. The observations are generated from the corresponding active distributions.*

The process generating the observations is demonstrated again in Figure 2.3, for state-dependent distributions p_1 and p_2, stationary distribution $\boldsymbol{\delta} = (0.75, 0.25)$, and t.p.m. $\boldsymbol{\Gamma} = \begin{pmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{pmatrix}$. In contrast to the case of an independent mixture, here the distribution of C_t, the state at time t, does depend on C_{t-1}. As is also true of independent mixtures, there is for each state a different distribution, discrete or continuous.

We now introduce some notation which will cover both discrete- and continuous-valued observations. In the case of discrete observations we define, for $i = 1, 2, \ldots, m$,

$$p_i(x) = \Pr(X_t = x \mid C_t = i).$$

That is, p_i is the probability mass function of X_t if the Markov chain is in state i at time t. The continuous case is treated similarly: there we define p_i to be the probability *density* function of X_t associated with state i. We refer to the m distributions p_i as the **state-dependent distributions** of the model. Many of our results are stated only for the discrete case, but, if probabilities are replaced by densities, apply also to the continuous case.

2.2.2 Marginal distributions

We shall often need the marginal distribution of X_t and also higher-order marginal distributions, such as that of (X_t, X_{t+k}). We shall derive the results for the case in which the Markov chain is homogeneous but not necessarily stationary, and then give them also for the special case in which the Markov chain is stationary. For convenience the derivation is given only for discrete state-dependent distributions; the continuous case can be derived analogously.

Univariate distributions

For discrete-valued observations X_t, defining $u_i(t) = \Pr(C_t = i)$ for $t = 1, \ldots, T$, we have

$$\begin{aligned} \Pr(X_t = x) &= \sum_{i=1}^{m} \Pr(C_t = i) \Pr(X_t = x \mid C_t = i) \\ &= \sum_{i=1}^{m} u_i(t) p_i(x). \end{aligned}$$

This expression can conveniently be rewritten in matrix notation:

$$\begin{aligned} \Pr(X_t = x) &= (u_1(t), \ldots, u_m(t)) \begin{pmatrix} p_1(x) & & 0 \\ & \ddots & \\ 0 & & p_m(x) \end{pmatrix} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \\ &= \mathbf{u}(t)\mathbf{P}(x)\mathbf{1}', \end{aligned}$$

where $\mathbf{P}(x)$ is defined as the diagonal matrix with ith diagonal element $p_i(x)$. It follows from equation (1.3) that $\mathbf{u}(t) = \mathbf{u}(1)\mathbf{\Gamma}^{t-1}$, and hence that

$$\Pr(X_t = x) = \mathbf{u}(1)\mathbf{\Gamma}^{t-1}\mathbf{P}(x)\mathbf{1}'. \tag{2.3}$$

Equation (2.3) holds if the Markov chain is merely homogeneous, and not necessarily stationary. If, as we shall often assume, the Markov chain is stationary, with stationary distribution $\boldsymbol{\delta}$, then the result is simpler: in that case $\boldsymbol{\delta}\mathbf{\Gamma}^{t-1} = \boldsymbol{\delta}$ for all $t \in \mathbb{N}$, and so

$$\Pr(X_t = x) = \boldsymbol{\delta}\mathbf{P}(x)\mathbf{1}'. \tag{2.4}$$

Bivariate distributions

The calculation of many of the distributions relating to an HMM is most easily done by first noting that, in any directed graphical model, the joint

distribution of a set of random variables V_i is given by

$$\Pr(V_1, V_2, \ldots, V_n) = \prod_{i=1}^{n} \Pr(V_i \mid \mathrm{pa}(V_i)), \tag{2.5}$$

where $\mathrm{pa}(V_i)$ denotes the set of all 'parents' of V_i in the set V_1, V_2, ..., V_n; see, for example, Davison (2003, p. 250) or Jordan (2004).

In the directed graph of the four random variables X_t, X_{t+k}, C_t, C_{t+k} (for positive integer k), C_t has no parents, $\mathrm{pa}(X_t) = \{C_t\}$, $\mathrm{pa}(C_{t+k}) = \{C_t\}$ and $\mathrm{pa}(X_{t+k}) = \{C_{t+k}\}$. It therefore follows that

$$\Pr(X_t, X_{t+k}, C_t, C_{t+k}) = \Pr(C_t)\Pr(X_t|C_t)\Pr(C_{t+k}|C_t)\Pr(X_{t+k}|C_{t+k}),$$

and hence that

$$\begin{aligned}
&\Pr(X_t = v, X_{t+k} = w) \\
&\quad = \sum_{i=1}^{m}\sum_{j=1}^{m} \Pr(X_t = v, X_{t+k} = w, C_t = i, C_{t+k} = j) \\
&\quad = \sum_{i=1}^{m}\sum_{j=1}^{m} \underbrace{\Pr(C_t = i)}_{u_i(t)}\, p_i(v)\, \underbrace{\Pr(C_{t+k} = j \mid C_t = i)}_{\gamma_{ij}(k)}\, p_j(w) \\
&\quad = \sum_{i=1}^{m}\sum_{j=1}^{m} u_i(t) p_i(v) \gamma_{ij}(k) p_j(w).
\end{aligned}$$

(Here and elsewhere, $\gamma_{ij}(k)$ denotes the (i, j) element of $\mathbf{\Gamma}^k$.) Writing the above double sum as a product of matrices yields

$$\Pr(X_t = v, X_{t+k} = w) \quad = \quad \mathbf{u}(t)\mathbf{P}(v)\mathbf{\Gamma}^k\mathbf{P}(w)\mathbf{1}'. \tag{2.6}$$

If the Markov chain is stationary, this reduces to

$$\Pr(X_t = v, X_{t+k} = w) \quad = \quad \boldsymbol{\delta}\mathbf{P}(v)\mathbf{\Gamma}^k\mathbf{P}(w)\mathbf{1}'. \tag{2.7}$$

Similarly, one can obtain expressions for the higher-order marginal distributions; in the stationary case, the formula for a trivariate distribution is, for positive integers k and l,

$$\Pr(X_t = v, X_{t+k} = w, X_{t+k+l} = z) = \boldsymbol{\delta}\mathbf{P}(v)\mathbf{\Gamma}^k\mathbf{P}(w)\mathbf{\Gamma}^l\mathbf{P}(z)\mathbf{1}'.$$

2.2.3 Moments

First, we note that

$$\mathrm{E}(X_t) = \sum_{i=1}^{m} \mathrm{E}(X_t \mid C_t = i)\Pr(C_t = i) = \sum_{i=1}^{m} u_i(t)\mathrm{E}(X_t \mid C_t = i),$$

which, in the stationary case, reduces to

$$\mathrm{E}(X_t) = \sum_{i=1}^{m} \delta_i \mathrm{E}(X_t \mid C_t = i).$$

More generally, analogous results hold for $\mathrm{E}(g(X_t))$ and $\mathrm{E}(g(X_t, X_{t+k}))$, for any functions g for which the relevant state-dependent expectations exist. In the stationary case

$$\mathrm{E}(g(X_t)) = \sum_{i=1}^{m} \delta_i \mathrm{E}(g(X_t) \mid C_t = i); \tag{2.8}$$

and

$$\mathrm{E}(g(X_t, X_{t+k})) = \sum_{i,j=1}^{m} \mathrm{E}(g(X_t, X_{t+k}) \mid C_t = i, C_{t+k} = j)\, \delta_i \gamma_{ij}(k). \tag{2.9}$$

Often we shall be interested in a function g which factorizes as

$$g(X_t, X_{t+k}) = g_1(X_t) g_2(X_{t+k}),$$

in which case equation (2.9) becomes

$$\mathrm{E}(g(X_t, X_{t+k})) = \sum_{i,j=1}^{m} \mathrm{E}(g_1(X_t) \mid C_t = i)\, \mathrm{E}(g_2(X_{t+k}) \mid C_{t+k} = j)\, \delta_i \gamma_{ij}(k). \tag{2.10}$$

These expressions enable us to find covariances and correlations; convenient explicit expressions exist in many cases. For the case of a stationary two-state Poisson–HMM:

- $\mathrm{E}(X_t) = \delta_1 \lambda_1 + \delta_2 \lambda_2$;
- $\mathrm{Var}(X_t) = \mathrm{E}(X_t) + \delta_1 \delta_2 (\lambda_2 - \lambda_1)^2 \geq \mathrm{E}(X_t)$;
- $\mathrm{Cov}(X_t, X_{t+k}) = \delta_1 \delta_2 (\lambda_2 - \lambda_1)^2 (1 - \gamma_{12} - \gamma_{21})^k$, for $k \in \mathbb{N}$.

Notice that the resulting formula for the correlation of X_t and X_{t+k} is of the form $\rho(k) = A(1 - \gamma_{12} - \gamma_{21})^k$ with $A \in [0, 1)$, and that $A = 0$ if $\lambda_1 = \lambda_2$. For more details, and for more general results, see Exercises 3 and 4.

2.3 The likelihood

The aim of this section is to develop a convenient formula for the likelihood L_T of T consecutive observations $x_1, x_2, \ldots, x_T$ assumed to be generated by an m-state HMM. That such a formula exists is indeed fortunate, but by no means obvious. We shall see that the computation of the likelihood, consisting as it does of a sum of m^T terms, each of which

is a product of $2T$ factors, appears to require $O(Tm^T)$ operations. However, it has long been known in several contexts that the likelihood is easily computable; see, for example, Baum (1972), Lange and Boehnke (1983), and Cosslett and Lee (1985). What we describe here is in fact a special case of a much more general theory; see Smyth, Heckerman and Jordan (1997) or Jordan (2004).

It is our purpose here to demonstrate that L_T can in general be computed relatively simply in $O(Tm^2)$ operations. The way will then be open to estimate parameters by numerical maximization of the likelihood. First the likelihood of a two-state model will be explored, and then the general formula will be presented.

2.3.1 The likelihood of a two-state Bernoulli–HMM

Example. Consider the stationary two-state HMM with t.p.m.

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

and state-dependent distributions given by

$$\begin{aligned} \Pr(X_t = x \mid C_t = 1) &= \frac{1}{2} \quad (\text{for } x = 0, 1), \\ \Pr(X_t = 1 \mid C_t = 2) &= 1. \end{aligned}$$

We call a model of this kind a **Bernoulli–HMM**. The stationary distribution of the Markov chain is $\boldsymbol{\delta} = \frac{1}{3}(1, 2)$. Consider the probability $\Pr(X_1 = X_2 = X_3 = 1)$. First, note that, by equation (2.5),

$$\begin{aligned} &\Pr(X_1, X_2, X_3, C_1, C_2, C_3) \\ &= \Pr(C_1) \Pr(X_1 \mid C_1) \Pr(C_2 \mid C_1) \Pr(X_2 \mid C_2) \Pr(C_3 \mid C_2) \Pr(X_3 \mid C_3); \end{aligned}$$

and then sum over the values assumed by C_1, C_2, C_3. The result is

$$\begin{aligned} &\Pr(X_1 = 1, X_2 = 1, X_3 = 1) \\ &= \sum_{i=1}^{2} \sum_{j=1}^{2} \sum_{k=1}^{2} \Pr(X_1 = 1, X_2 = 1, X_3 = 1, C_1 = i, C_2 = j, C_3 = k) \\ &= \sum_{i=1}^{2} \sum_{j=1}^{2} \sum_{k=1}^{2} \delta_i p_i(1) \gamma_{ij} p_j(1) \gamma_{jk} p_k(1). \end{aligned} \tag{2.11}$$

Notice that the triple sum (2.11) has $m^T = 2^3$ terms, each of which is a product of $2T = 2 \times 3$ factors. To evaluate the required probability, the different possibilities for the values of i, j and k can be listed and the sum (2.11) calculated as in Table 2.1.

Summation of the last column of Table 2.1 tells us that $\Pr(X_1 =$

Table 2.1 *Example of a likelihood computation.*

i	j	k	$p_i(1)$	$p_j(1)$	$p_k(1)$	δ_i	γ_{ij}	γ_{jk}	Product
1	1	1	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{2}{4}$	$\frac{2}{4}$	$\frac{1}{96}$
1	1	2	$\frac{1}{2}$	$\frac{1}{2}$	1	$\frac{1}{3}$	$\frac{2}{4}$	$\frac{2}{4}$	$\frac{1}{48}$
1	2	1	$\frac{1}{2}$	1	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{2}{4}$	$\frac{1}{4}$	$\frac{1}{96}$
1	2	2	$\frac{1}{2}$	1	1	$\frac{1}{3}$	$\frac{2}{4}$	$\frac{3}{4}$	$\frac{1}{16}$
2	1	1	1	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{2}{3}$	$\frac{1}{4}$	$\frac{2}{4}$	$\frac{1}{48}$
2	1	2	1	$\frac{1}{2}$	1	$\frac{2}{3}$	$\frac{1}{4}$	$\frac{2}{4}$	$\frac{1}{24}$
2	2	1	1	1	$\frac{1}{2}$	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{1}{4}$	$\frac{1}{16}$
2	2	2	1	1	1	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{3}{4}$	$\frac{3}{8}$
									$\frac{29}{48}$

$1, X_2 = 1, X_3 = 1) = \frac{29}{48}$. In passing we note that the largest element in that column is $\frac{3}{8}$; the state sequence that maximizes the joint probability

$$\Pr(X_1 = 1, X_2 = 1, X_3 = 1, C_1 = i, C_2 = j, C_3 = k)$$

is therefore the sequence $i = 2$, $j = 2$, $k = 2$. Equivalently, it maximizes the conditional probability $\Pr(C_1 = i, C_2 = j, C_3 = k \mid X_1 = 1, X_2 = 1, X_3 = 1)$. This is an example of 'global decoding', which will be discussed in Section 5.4.2.

But a more convenient way to present the sum is to use matrix notation. Let $\mathbf{P}(u)$ be defined (as before) as $\text{diag}(p_1(u), p_2(u))$. Then

$$\mathbf{P}(0) = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & 0 \end{pmatrix} \quad \text{and} \quad \mathbf{P}(1) = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{pmatrix},$$

and the triple sum (2.11) can be written as a matrix product:

$$\boldsymbol{\delta}\mathbf{P}(1)\boldsymbol{\Gamma}\mathbf{P}(1)\boldsymbol{\Gamma}\mathbf{P}(1)\mathbf{1}'.$$

More generally, the likelihood turns out to be a T-fold sum which can also be written as a matrix product.

2.3.2 The likelihood in general

Here we consider the likelihood of an HMM in general. We suppose there is an observation sequence $x_1, x_2, \ldots, x_T$ generated by such a model. We

seek the probability L_T of observing that sequence, as calculated under an m-state HMM which has *initial* distribution $\boldsymbol{\delta}$ and t.p.m. $\boldsymbol{\Gamma}$ for the Markov chain, and state-dependent probability (density) functions p_i. In many of our applications we shall assume that $\boldsymbol{\delta}$ is the stationary distribution implied by $\boldsymbol{\Gamma}$, but it is not necessary to make that assumption in general.

Proposition 1 *The likelihood is given by*

$$L_T = \boldsymbol{\delta}\mathbf{P}(x_1)\boldsymbol{\Gamma}\mathbf{P}(x_2)\boldsymbol{\Gamma}\mathbf{P}(x_3)\cdots\boldsymbol{\Gamma}\mathbf{P}(x_T)\mathbf{1}'. \tag{2.12}$$

If $\boldsymbol{\delta}$, the distribution of C_1, is the stationary distribution of the Markov chain, then in addition

$$L_T = \boldsymbol{\delta}\boldsymbol{\Gamma}\mathbf{P}(x_1)\boldsymbol{\Gamma}\mathbf{P}(x_2)\boldsymbol{\Gamma}\mathbf{P}(x_3)\cdots\boldsymbol{\Gamma}\mathbf{P}(x_T)\mathbf{1}'. \tag{2.13}$$

Proof. We present only the case of discrete observations. First, note that

$$L_T = \Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)}) = \sum_{c_1,c_2,\ldots,c_T=1}^{m} \Pr(\mathbf{X}^{(T)} = \mathbf{x}^{(T)}, \mathbf{C}^{(T)} = \mathbf{c}^{(T)}),$$

and that, by equation (2.5),

$$\Pr(\mathbf{X}^{(T)}, \mathbf{C}^{(T)}) = \Pr(C_1)\prod_{k=2}^{T}\Pr(C_k \mid C_{k-1})\prod_{k=1}^{T}\Pr(X_k \mid C_k). \tag{2.14}$$

It follows that

$$\begin{aligned} L_T &= \sum_{c_1,\ldots,c_T=1}^{m} \left(\delta_{c_1}\gamma_{c_1,c_2}\gamma_{c_2,c_3}\cdots\gamma_{c_{T-1},c_T}\right)\left(p_{c_1}(x_1)p_{c_2}(x_2)\cdots p_{c_T}(x_T)\right) \\ &= \sum_{c_1,\ldots,c_T=1}^{m} \delta_{c_1}p_{c_1}(x_1)\gamma_{c_1,c_2}p_{c_2}(x_2)\gamma_{c_2,c_3}\cdots\gamma_{c_{T-1},c_T}p_{c_T}(x_T) \\ &= \boldsymbol{\delta}\mathbf{P}(x_1)\boldsymbol{\Gamma}\mathbf{P}(x_2)\boldsymbol{\Gamma}\mathbf{P}(x_3)\cdots\boldsymbol{\Gamma}\mathbf{P}(x_T)\mathbf{1}', \end{aligned}$$

which is equation (2.12). The last equality above exploits the fact that a multiple sum of terms having a certain simple multiplicative form can in general be written as a matrix product. Exercise 7(b) provides the detailed justification.

If $\boldsymbol{\delta}$ is the stationary distribution of the Markov chain, we have

$$\boldsymbol{\delta}\mathbf{P}(x_1) = \boldsymbol{\delta}\boldsymbol{\Gamma}\mathbf{P}(x_1),$$

hence equation (2.13), which involves an extra factor of $\boldsymbol{\Gamma}$ but may be slightly simpler to code. □

A very simple but crucial consequence of the matrix expression for the likelihood is the 'forward algorithm' for recursive computation of

the likelihood. Such recursive computation plays a key role, not only in likelihood evaluation and hence parameter estimation, but also in forecasting, decoding and model checking. The recursive nature of likelihood evaluation via either (2.12) or (2.13) is computationally much more efficient than brute-force summation over all possible state sequences. The fact that such computationally inexpensive recursive schemes can be used to address various questions of interest is a key feature of HMMs. Recursive evaluation of such multiple sums has been discussed by Lange and Boehnke (1983) and Lange (2002, p. 120).

To state the forward algorithm we define the vector $\boldsymbol{\alpha}_t$, for $t = 1, 2, \ldots, T$, by

$$\boldsymbol{\alpha}_t = \boldsymbol{\delta}\mathbf{P}(x_1)\boldsymbol{\Gamma}\mathbf{P}(x_2)\boldsymbol{\Gamma}\mathbf{P}(x_3)\cdots\boldsymbol{\Gamma}\mathbf{P}(x_t) = \boldsymbol{\delta}\mathbf{P}(x_1)\prod_{s=2}^{t}\boldsymbol{\Gamma}\mathbf{P}(x_s), \quad (2.15)$$

with the convention that an empty product is the identity matrix. It follows immediately from this definition that

$$L_T = \boldsymbol{\alpha}_T\mathbf{1}', \quad \text{and} \quad \boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1}\boldsymbol{\Gamma}\mathbf{P}(x_t) \quad \text{for } t \geq 2.$$

Accordingly, we can conveniently set out as follows the computations involved in the likelihood formula (2.12):

$$\boldsymbol{\alpha}_1 = \boldsymbol{\delta}\mathbf{P}(x_1);$$
$$\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1}\boldsymbol{\Gamma}\mathbf{P}(x_t) \quad \text{for } t = 2, 3, \ldots, T;$$
$$L_T = \boldsymbol{\alpha}_T\mathbf{1}'.$$

That the number of operations involved is of order Tm^2 can be deduced thus. For each of the values of t in the loop, there are m elements of $\boldsymbol{\alpha}_t$ to be computed, and each of those elements is a sum of m products of three quantities: an element of $\boldsymbol{\alpha}_{t-1}$, a transition probability γ_{ij}, and a state-dependent probability (or density) $p_j(x_t)$.

The corresponding scheme for computation of (2.13) (i.e. if $\boldsymbol{\delta}$, the distribution of C_1, is the stationary distribution of the Markov chain) is

$$\boldsymbol{\alpha}_0 = \boldsymbol{\delta};$$
$$\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1}\boldsymbol{\Gamma}\mathbf{P}(x_t) \quad \text{for } t = 1, 2, \ldots, T;$$
$$L_T = \boldsymbol{\alpha}_T\mathbf{1}'.$$

The elements of the vector $\boldsymbol{\alpha}_t$ are usually referred to as **forward probabilities**; the reason for this name will be seen in Section 4.1.1, where we show that the jth element of $\boldsymbol{\alpha}_t$ is $\Pr(\mathbf{X}^{(t)} = \mathbf{x}^{(t)}, C_t = j)$.

We show here **R** code that uses the forward algorithm to evaluate the likelihood of observations $x_1, \ldots, x_T$ under a Poisson–HMM with at least two states, t.p.m. $\boldsymbol{\Gamma}$, vector of state-dependent means $\boldsymbol{\lambda}$, and initial distribution $\boldsymbol{\delta}$ (not necessarily the stationary distribution). Note,

however, that, unless the series is short, one needs to guard against underflow and evaluate the log-likelihood rather than the likelihood; see p. 49 for code that does so.

```
alpha               <- delta*dpois(x[1],lambda)
for (i in 2:T) alpha <- alpha %*% Gamma*dpois(x[i],lambda)
sum(alpha)
```

In the above discussion we have used the multiple-sum expression for the likelihood in order to arrive at the matrix expression, and then used the matrix expression to arrive at the forward recursion. An alternative route, which anticipates some of the material of Chapter 4, is to *define* the vector of forward probabilities $\boldsymbol{\alpha}_t$ by

$$\alpha_t(j) = \Pr(\mathbf{X}^{(t)} = \mathbf{x}^{(t)}, C_t = j), \quad j = 1, 2, \ldots, m,$$

and then to deduce the forward recursion:

$$\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1}\boldsymbol{\Gamma}\mathbf{P}(x_t).$$

The matrix expression is then a simple consequence of the forward recursion. This alternative route is described in Exercise 8.

2.3.3 HMMs are not Markov processes

HMMs do not in general satisfy the Markov property. This we can now establish via a simple counterexample. Let X_t and C_t be as defined in the example in Section 2.3.1. We already know that

$$\Pr(X_1 = 1, X_2 = 1, X_3 = 1) = \frac{29}{48},$$

and from equations (2.4) and (2.7) it can be established that $\Pr(X_2 = 1) = \frac{5}{6}$ and that

$$\Pr(X_1 = 1, X_2 = 1) = \Pr(X_2 = 1, X_3 = 1) = \frac{17}{24}.$$

It therefore follows that

$$\begin{aligned} \Pr(X_3 = 1 \mid X_1 = 1, X_2 = 1) &= \frac{\Pr(X_1 = 1, X_2 = 1, X_3 = 1)}{\Pr(X_1 = 1, X_2 = 1)} \\ &= \frac{29/48}{17/24} = \frac{29}{34} \end{aligned}$$

and that

$$\begin{aligned} \Pr(X_3 = 1 \mid X_2 = 1) &= \frac{\Pr(X_2 = 1, X_3 = 1)}{\Pr(X_2 = 1)} \\ &= \frac{17/24}{5/6} = \frac{17}{20}. \end{aligned}$$

Hence $\Pr(X_3 = 1 \mid X_2 = 1) \neq \Pr(X_3 = 1 \mid X_1 = 1, X_2 = 1)$; this HMM does not satisfy the Markov property. That some HMMs do satisfy the property, however, is clear. For instance, a two-state Bernoulli–HMM can degenerate in obvious fashion to the underlying Markov chain; one simply identifies each of the two observable values with one of the two underlying states. For the conditions under which an HMM will itself satisfy the Markov property, see Spreij (2001).

2.3.4 The likelihood when data are missing

In a time series context it is potentially awkward if some of the data are missing. In the case of hidden Markov time series models, however, the adjustment that needs to be made to the likelihood computation if data are missing turns out to be a simple one.

Suppose, for example, that one has available the observations x_1, x_2, x_4, x_7, x_8, ..., x_T of an HMM, but x_3, x_5 and x_6 are missing. Then the likelihood of the observations is given by

$$\begin{aligned}\Pr(X_1 = x_1, X_2 = x_2, X_4 = x_4, X_7 = x_7, \ldots, X_T = x_T) \\ = \sum \delta_{c_1}\gamma_{c_1,c_2}\gamma_{c_2,c_4}(2)\gamma_{c_4,c_7}(3)\gamma_{c_7,c_8}\cdots\gamma_{c_{T-1},c_T} \\ \times p_{c_1}(x_1)p_{c_2}(x_2)p_{c_4}(x_4)p_{c_7}(x_7)\cdots p_{c_T}(x_T),\end{aligned}$$

where (as before) $\gamma_{ij}(k)$ denotes a k-step transition probability, and the sum is taken over all indices c_t other than c_3, c_5 and c_6. But this is just

$$\begin{aligned}\sum \delta_{c_1}p_{c_1}(x_1)\gamma_{c_1,c_2}p_{c_2}(x_2)\gamma_{c_2,c_4}(2)p_{c_4}(x_4)\gamma_{c_4,c_7}(3)p_{c_7}(x_7) \\ \times\cdots\times\gamma_{c_{T-1},c_T}p_{c_T}(x_T) \\ = \boldsymbol{\delta}\mathbf{P}(x_1)\boldsymbol{\Gamma}\mathbf{P}(x_2)\boldsymbol{\Gamma}^2\mathbf{P}(x_4)\boldsymbol{\Gamma}^3\mathbf{P}(x_7)\cdots\boldsymbol{\Gamma}\mathbf{P}(x_T)\mathbf{1}'.\end{aligned}$$

With $L_T^{-(3,5,6)}$ denoting the likelihood of the observations (other than x_3, x_5 and x_6), it follows that

$$L_T^{-(3,5,6)} = \boldsymbol{\delta}\mathbf{P}(x_1)\boldsymbol{\Gamma}\mathbf{P}(x_2)\boldsymbol{\Gamma}^2\mathbf{P}(x_4)\boldsymbol{\Gamma}^3\mathbf{P}(x_7)\cdots\boldsymbol{\Gamma}\mathbf{P}(x_T)\mathbf{1}'.$$

In general, in the expression for the likelihood the diagonal matrices $\mathbf{P}(x_t)$ corresponding to missing observations x_t are replaced by the identity matrix; that is, the corresponding state-dependent probabilities $p_i(x_t)$ are replaced by 1 for all states i. If one can assume that the missingness is ignorable, this 'ignorable likelihood' is a reasonable basis for estimating parameters (Little, 2009, p. 411).

The fact that, even if some observations are missing, the likelihood of an HMM can be computed easily is especially useful in the derivation of conditional distributions, as will be shown in Section 5.2.

2.3.5 The likelihood when observations are interval-censored

Suppose that we wish to fit a Poisson–HMM to a series of counts, some of which are interval-censored. For instance, the value of x_t may be known only for $4 \leq t \leq T$, with the information $x_1 \leq 5$, $2 \leq x_2 \leq 3$ and $x_3 > 10$ available about the remaining observations. For simplicity, let us first assume that the Markov chain has only two states. In that case, one replaces the diagonal matrices $\mathbf{P}(x_i)$ $(i = 1, 2, 3)$ in the likelihood expression (2.12) by the matrices

$$\begin{aligned}
&\text{diag}(\Pr(X_1 \leq 5 \mid C_1 = 1), \Pr(X_1 \leq 5 \mid C_1 = 2)),\\
&\text{diag}(\Pr(2 \leq X_2 \leq 3 \mid C_2 = 1), \Pr(2 \leq X_2 \leq 3 \mid C_2 = 2)), \text{and}\\
&\text{diag}(\Pr(X_3 > 10 \mid C_3 = 1), \Pr(X_3 > 10 \mid C_3 = 2)).
\end{aligned}$$

More generally, suppose that $a \leq x_t \leq b$, where a may be $-\infty$ (although that is not relevant to the Poisson case), b may be ∞, and the Markov chain has m states. One replaces $\mathbf{P}(x_t)$ in the likelihood by the $m \times m$ diagonal matrix of which the ith diagonal element is $\Pr(a \leq X_t \leq b \mid C_t = i)$. See Exercise 12. It is worth noting that missing data can be regarded as an extreme case of such interval-censoring.

Exercises

1. Consider a *stationary* two-state Poisson–HMM with parameters

$$\boldsymbol{\Gamma} = \begin{pmatrix} 0.1 & 0.9 \\ 0.4 & 0.6 \end{pmatrix} \quad \text{and} \quad \boldsymbol{\lambda} = (1, 3).$$

   In each of the following ways, compute the probability that the first three observations from this model are 0, 2, 1.

   (a) Consider all possible sequences of states of the Markov chain that could have occurred. Compute the probability of each sequence, and the probability of the observations given each sequence.

   (b) Apply the formula

$$\Pr(X_1 = 0, X_2 = 2, X_3 = 1) = \boldsymbol{\delta}\mathbf{P}(0)\boldsymbol{\Gamma}\mathbf{P}(2)\boldsymbol{\Gamma}\mathbf{P}(1)\mathbf{1}',$$

   where

$$\mathbf{P}(s) = \begin{pmatrix} \lambda_1^s e^{-\lambda_1}/s! & 0 \\ 0 & \lambda_2^s e^{-\lambda_2}/s! \end{pmatrix} = \begin{pmatrix} 1^s e^{-1}/s! & 0 \\ 0 & 3^s e^{-3}/s! \end{pmatrix}.$$

2. Consider again the model defined in Exercise 1. In that question you were asked to compute $\Pr(X_1 = 0, X_2 = 2, X_3 = 1)$. Now compute $\Pr(X_1 = 0, X_3 = 1)$ in each of the following ways.

   (a) Consider all possible sequences of states of the Markov chain that

could have occurred. Compute the probability of each sequence, and the probability of the observations given each sequence.

(b) Apply the formula

$$\Pr(X_1=0, X_3=1) = \boldsymbol{\delta}\mathbf{P}(0)\boldsymbol{\Gamma}\mathbf{I}_2\boldsymbol{\Gamma}\mathbf{P}(1)\mathbf{1}' = \boldsymbol{\delta}\mathbf{P}(0)\boldsymbol{\Gamma}^2\mathbf{P}(1)\mathbf{1}',$$

and check that this probability is equal to your answer in (a).

3. Consider an m-state HMM $\{X_t : t = 1, 2, \ldots\}$, based on a stationary Markov chain with transition probability matrix $\boldsymbol{\Gamma}$ and stationary distribution $\boldsymbol{\delta} = (\delta_1, \delta_2, \ldots, \delta_m)$, and having (univariate) state-dependent distributions $p_i(x)$. Let μ_i and σ_i^2 denote the mean and variance of the distribution p_i, $\boldsymbol{\mu}$ the vector $(\mu_1, \mu_2, \ldots, \mu_m)$, and $\mathbf{M}$ the matrix $\text{diag}(\boldsymbol{\mu})$.

Derive the following results for the moments of $\{X_t\}$.

(a) $\text{E}(X_t) = \sum_{i=1}^m \delta_i\mu_i = \boldsymbol{\delta}\boldsymbol{\mu}'$.
(b) $\text{E}(X_t^2) = \sum_{i=1}^m \delta_i(\sigma_i^2 + \mu_i^2)$.
(c) $\text{Var}(X_t) = \sum_{i=1}^m \delta_i(\sigma_i^2 + \mu_i^2) - (\boldsymbol{\delta}\boldsymbol{\mu}')^2$.
(d) If $m = 2$, $\text{Var}(X_t) = \delta_1\sigma_1^2 + \delta_2\sigma_2^2 + \delta_1\delta_2(\mu_1 - \mu_2)^2$.
(e) For $k \in \mathbb{N}$, i.e. for positive integers k,

$$\text{E}(X_tX_{t+k}) = \sum_{i=1}^m\sum_{j=1}^m \delta_i\mu_i\gamma_{ij}(k)\mu_j = \boldsymbol{\delta}\mathbf{M}\boldsymbol{\Gamma}^k\boldsymbol{\mu}'.$$

(f) For $k \in \mathbb{N}$,

$$\rho(k) = \text{Corr}(X_t, X_{t+k}) = \frac{\boldsymbol{\delta}\mathbf{M}\boldsymbol{\Gamma}^k\boldsymbol{\mu}' - (\boldsymbol{\delta}\boldsymbol{\mu}')^2}{\text{Var}(X_t)}.$$

Note that, if the eigenvalues of $\boldsymbol{\Gamma}$ are distinct, this is a linear combination of the kth powers of those eigenvalues.

(g) If the state-dependent means μ_i are all equal, X_t and X_{t+k} are uncorrelated for $k \in \mathbb{N}$.

Timmermann (2000) and Frühwirth-Schnatter (2006, pp. 308–312) are useful references for moments. See also Exercise 1 of Chapter 1.

4. (Marginal moments and autocorrelation function of a Poisson–HMM: special case of Exercise 3.) Consider a stationary m-state Poisson–HMM $\{X_t : t = 1, 2, \ldots\}$ with transition probability matrix $\boldsymbol{\Gamma}$ and state-dependent means $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_m)$. Let $\boldsymbol{\delta} = (\delta_1, \delta_2, \ldots, \delta_m)$ be the stationary distribution of the Markov chain. Let $\boldsymbol{\Lambda} = \text{diag}(\boldsymbol{\lambda})$.

Derive the following results.

(a) $\text{E}(X_t) = \boldsymbol{\delta}\boldsymbol{\lambda}'$.
(b) $\text{E}(X_t^2) = \sum_{i=1}^m (\lambda_i^2 + \lambda_i)\delta_i = \boldsymbol{\delta}\Lambda\boldsymbol{\lambda}' + \boldsymbol{\delta}\boldsymbol{\lambda}'$.

(c) $\text{Var}(X_t) = \boldsymbol{\delta}\boldsymbol{\Lambda}\boldsymbol{\lambda}' + \boldsymbol{\delta}\boldsymbol{\lambda}' - (\boldsymbol{\delta}\boldsymbol{\lambda}')^2 = \text{E}(X_t) + \boldsymbol{\delta}\boldsymbol{\Lambda}\boldsymbol{\lambda}' - (\boldsymbol{\delta}\boldsymbol{\lambda}')^2 \geq \text{E}(X_t)$.

(d) For $k \in \mathbb{N}$, $\text{E}(X_t X_{t+k}) = \boldsymbol{\delta}\boldsymbol{\Lambda}\boldsymbol{\Gamma}^k\boldsymbol{\lambda}'$.

(e) For $k \in \mathbb{N}$,

$$\rho(k) = \text{Corr}(X_t, X_{t+k}) = \frac{\boldsymbol{\delta}\boldsymbol{\Lambda}\boldsymbol{\Gamma}^k\boldsymbol{\lambda}' - (\boldsymbol{\delta}\boldsymbol{\lambda}')^2}{\boldsymbol{\delta}\boldsymbol{\Lambda}\boldsymbol{\lambda}' + \boldsymbol{\delta}\boldsymbol{\lambda}' - (\boldsymbol{\delta}\boldsymbol{\lambda}')^2}.$$

(f) In the case $m = 2$, $\rho(k) = Aw^k$ for $k \in \mathbb{N}$, where

$$A = \frac{\delta_1\delta_2(\lambda_2 - \lambda_1)^2}{\delta_1\delta_2(\lambda_2 - \lambda_1)^2 + \boldsymbol{\delta}\boldsymbol{\lambda}'}$$

and $w = 1 - \gamma_{12} - \gamma_{21}$. Notice that the extra level of randomness in the HMM, as compared with the underlying Markov chain, has reduced the autocorrelations by the factor $A \in [0, 1)$.

5. (A serially dependent process with zero autocorrelation.) In finance, time-series models consisting of serially uncorrelated but dependent random variables are often of interest. We consider here a stationary HMM $\{X_t\}$, with normal state-dependent distributions, that is such a process. Suppose that

$$\boldsymbol{\Gamma} = \begin{pmatrix} 0.990 & 0.005 & 0.005 \\ 0.010 & 0.980 & 0.010 \\ 0.015 & 0.015 & 0.970 \end{pmatrix}$$

and that, given $C_t = i$, $X_t \sim \text{N}(1, \sigma_i^2)$, with $(\sigma_1, \sigma_2, \sigma_3) = (1, 10, 20)$. By Exercise 3(g), X_t and X_{t+k} are uncorrelated for $k \in \mathbb{N}$.

(a) Simulate (say) 10 000 observations $\{x_t\}$ from this model. One way of doing so is to modify the code in Section A.1.5, which applies to the case of Poisson state-dependent distributions.

(b) Using the **R** function `acf`, plot the sample ACF of:

i. $\{x_t\}$;
ii. $\{|x_t|\}$;
iii. $\{x_t^2\}$.

What can you conclude from these three sample ACFs?

(c) Find the ACF of $\{X_t^2\}$ under the model, and superimpose it on your plot of the ACF of $\{x_t^2\}$. You should get a plot similar to that shown in Figure 2.4.

6. We have the general expression

$$L_T = \boldsymbol{\delta}\mathbf{P}(x_1)\boldsymbol{\Gamma}\mathbf{P}(x_2)\cdots\boldsymbol{\Gamma}\mathbf{P}(x_T)\mathbf{1}'$$

for the likelihood of an HMM, e.g. of Poisson type.

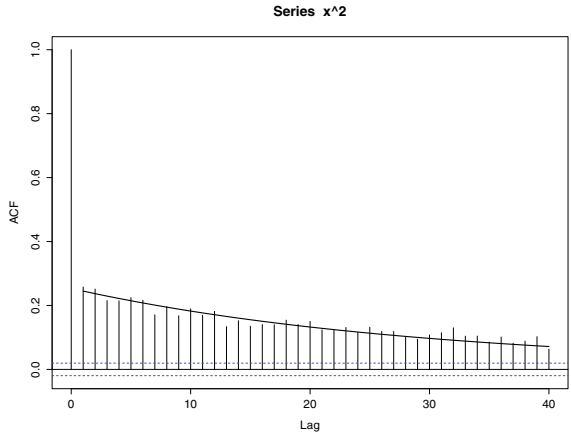


Figure 2.4 *Exercise 5: Sample autocorrelation function of the squares of 10 000 simulated observations, plus ACF of* $\{X_t^2\}$ *under the model (continuous line).*

(a) Consider the special case in which the Markov chain degenerates to a sequence of independent, identically-distributed random variables, i.e. an independent mixture model. Show that, in this case, the likelihood simplifies to the expression given in equation (1.1) for the likelihood of an *independent* mixture.

(b) Suppose instead that, for all i and x, $p_i(x) = p(x)$. What does the likelihood expression

$$\boldsymbol{\delta}\mathbf{P}(x_1)\boldsymbol{\Gamma}\mathbf{P}(x_2)\boldsymbol{\Gamma}\mathbf{P}(x_3)\cdots\boldsymbol{\Gamma}\mathbf{P}(x_T)\mathbf{1}'$$

now reduce to, and what do you conclude?

7. This exercise shows that a sum of m^T terms of a certain simple multiplicative form can (perhaps surprisingly) be computed efficiently, in $O(Tm^2)$ operations.

Consider a multiple sum S of the following general form:

$$S = \sum_{i_1=1}^{m}\sum_{i_2=1}^{m}\cdots\sum_{i_T=1}^{m} h(i_1)\prod_{t=2}^{T} f_t(i_{t-1}, i_t).$$

For $i_1 = 1, 2, \ldots, m$, define the (row) vector $\boldsymbol{\alpha}_1$ by

$$\alpha_1(i_1) \equiv h(i_1);$$

and for $r = 1, 2, \ldots, T-1$ and $i_{r+1} = 1, 2, \ldots, m$, define

$$\alpha_{r+1}(i_{r+1}) \equiv \sum_{i_r=1}^{m} \alpha_r(i_r)\, f_{r+1}(i_r, i_{r+1}).$$

That is, the vector $\boldsymbol{\alpha}_{r+1}$ is defined by, and can be computed as, $\boldsymbol{\alpha}_{r+1} = \boldsymbol{\alpha}_r\, \mathbf{F}_{r+1}$, where the $m \times m$ matrix $\mathbf{F}_t$ has (i, j) element equal to $f_t(i, j)$.

(a) Show by induction on T that $\alpha_T(i_T)$ is precisely the sum over all but i_T, i.e. that

$$\alpha_T(i_T) = \sum_{i_1} \sum_{i_2} \cdots \sum_{i_{T-1}} h(i_1) \prod_{t=2}^{T} f_t(i_{t-1}, i_t).$$

(b) Hence show that $S = \sum_{i_T} \alpha_T(i_T) = \boldsymbol{\alpha}_T \mathbf{1}' = \boldsymbol{\alpha}_1 \mathbf{F}_2 \mathbf{F}_3 \cdots \mathbf{F}_T \mathbf{1}'$.

8. Consider an m-state HMM with the basic dependence structure as depicted in Figure 2.2.

(a) Consider the vector $\boldsymbol{\alpha}_t = (\alpha_t(1), \ldots, \alpha_t(m))$ defined by

$$\alpha_t(j) = \Pr(\mathbf{X}^{(t)} = \mathbf{x}^{(t)}, C_t = j), \quad j = 1, 2, \ldots, m.$$

Use conditional probability and the conditional independence assumptions to show that

$$\alpha_t(j) = \sum_{i=1}^{m} \alpha_{t-1}(i) \gamma_{ij} p_j(x_t).$$

(b) Verify for yourself that the result from (a), written in matrix notation, yields the forward recursion

$$\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1} \boldsymbol{\Gamma} \mathbf{P}(x_t), \quad t = 2, \ldots, T.$$

(c) Hence derive the matrix expression for the likelihood.

9. Write a function `pois-HMM.moments(m,lambda,gamma,lag.max=10)` that computes the expectation, variance and autocorrelation function (for lags 0 to `lag.max`) of an `m`-state stationary Poisson–HMM with t.p.m. `gamma` and state-dependent means `lambda`.

Hint: when finding the autocorrelation function, use the **R** package `expm` to compute the necessary powers of the t.p.m.

10. Write the three functions listed below, relating to the marginal distribution of an `m`-state Poisson–HMM with parameters `lambda`, `gamma`, and possibly `delta`. In each case, if `delta` is specified as `NULL`, the stationary distribution should be used. You can use your function `statdist` (see Exercise 9(b) of Chapter 1) to provide the stationary distribution.

```
dpois.HMM(x, m, lambda, gamma, delta=NULL)
ppois.HMM(x, m, lambda, gamma, delta=NULL)
qpois.HMM(p, m, lambda, gamma, delta=NULL)
```

The function `dpois.HMM` computes the probability function at the arguments specified by the vector `x`, `ppois.HMM` the distribution function, and `qpois.HMM` the inverse distribution function.

11. Consider the function `pois.HMM.generate_sample` in Section A.1.5 that generates observations from a stationary m-state Poisson–HMM. Test the function by generating a long sequence of observations (10 000, say), and then check whether the sample mean, variance, ACF and relative frequencies correspond to what you expect.

12. (Interval-censored observations.)

    (a) Suppose that, in a series of unbounded counts $x_1, \ldots, x_T$, only the observation x_t is interval-censored, and $a \leq x_t \leq b$, where b may be ∞. Prove the statement made in Section 2.3.5 that the likelihood of a Poisson–HMM with m states is obtained by replacing $\mathbf{P}(x_t)$ in the expression (2.12) by the $m \times m$ diagonal matrix of which the ith diagonal element is $\Pr(a \leq X_t \leq b \mid C_t = i)$.

    (b) Extend part (a) to allow for any number of interval-censored observations.

CHAPTER 3

Estimation by direct maximization of the likelihood

3.1 Introduction

We saw in equation (2.12) that the likelihood of an HMM is given by

$$L_T = \Pr\left(\mathbf{X}^{(T)} = \mathbf{x}^{(T)}\right) = \boldsymbol{\delta}\mathbf{P}(x_1)\boldsymbol{\Gamma}\mathbf{P}(x_2)\cdots\boldsymbol{\Gamma}\mathbf{P}(x_T)\mathbf{1}',$$

where $\boldsymbol{\delta}$ is the initial distribution (that of C_1) and $\mathbf{P}(x)$ the $m \times m$ diagonal matrix with ith diagonal element the state-dependent probability or density $p_i(x)$. In principle, we can therefore compute $L_T = \boldsymbol{\alpha}_T\mathbf{1}'$ recursively via

$$\boldsymbol{\alpha}_1 = \boldsymbol{\delta}\mathbf{P}(x_1)$$

and

$$\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1}\boldsymbol{\Gamma}\mathbf{P}(x_t), \quad \text{for } t = 2, 3, \ldots, T.$$

If the Markov chain is assumed stationary (in which case $\boldsymbol{\delta} = \boldsymbol{\delta}\boldsymbol{\Gamma}$), we can choose to use instead

$$\boldsymbol{\alpha}_0 = \boldsymbol{\delta}$$

and

$$\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1}\boldsymbol{\Gamma}\mathbf{P}(x_t), \quad \text{for } t = 1, 2, \ldots, T.$$

We shall first consider the stationary case.

The number of operations involved is of order Tm^2, making the evaluation of the likelihood quite feasible even for large T. Parameter estimation can therefore be performed by numerical maximization of the likelihood with respect to the parameters.

But there are several problems that need to be addressed when parameters are estimated in this way. The main problems are numerical underflow, constraints on the parameters, and multiple local maxima in the likelihood function. In this chapter we first discuss how to overcome these problems, in order to arrive at a general strategy for computing MLEs. Then we discuss the estimation of standard errors for parameters. We defer to the next chapter the EM algorithm, which necessitates some discussion of the forward and backward probabilities.

3.2 Scaling the likelihood computation

In the case of discrete state-dependent distributions, the elements of $\boldsymbol{\alpha}_t$, being made up of products of probabilities, become progressively smaller as t increases, and are eventually rounded to zero. In fact, with probability 1 the likelihood approaches 0 (or possibly ∞ in the continuous case) exponentially fast; see Leroux and Puterman (1992). The remedy is, however, the same for over- and underflow, and we confine our attention to underflow.

Since the likelihood is a product of matrices, not of scalars, it is not possible to circumvent numerical underflow simply by computing the log of the likelihood as the sum of logs of its factors. In this respect the computation of the likelihood of an independent mixture model is simpler than that of an HMM.

To solve the problem, Durbin *et al.* (1998, p. 78) suggest (*inter alia*) a method of computation that relies on the following approximation. Suppose we wish to compute $\log(p+q)$, where $p > q$. Write $\log(p+q)$ as

$$\log p + \log(1 + q/p) = \log p + \log(1 + \exp(\tilde{q} - \tilde{p})),$$

where $\tilde{p} = \log p$ and $\tilde{q} = \log q$. The function $\log(1 + e^x)$ is then approximated by interpolation from a table of its values; apparently quite a small table will give a reasonable degree of accuracy.

We prefer to compute the logarithm of L_T by using a strategy of scaling the vector of forward probabilities $\boldsymbol{\alpha}_t$. Effectively we scale the vector $\boldsymbol{\alpha}_t$ at each time t so that its elements add to 1, keeping track of the sum of the logs of the scale factors thus applied.

Define, for $t = 0, 1, \ldots, T$, the vector

$$\boldsymbol{\phi}_t = \boldsymbol{\alpha}_t / w_t,$$

where $w_t = \sum_i \alpha_t(i) = \boldsymbol{\alpha}_t \mathbf{1}'$. First, we note certain immediate consequences of the definitions of $\boldsymbol{\phi}_t$ and w_t:

$$\begin{aligned} w_0 = \boldsymbol{\alpha}_0 \mathbf{1}' &= \boldsymbol{\delta}\mathbf{1}' = 1; \\ \boldsymbol{\phi}_0 &= \boldsymbol{\delta}; \\ w_t \boldsymbol{\phi}_t &= w_{t-1} \boldsymbol{\phi}_{t-1} \boldsymbol{\Gamma} \mathbf{P}(x_t); \\ L_T = \boldsymbol{\alpha}_T \mathbf{1}' &= w_T (\boldsymbol{\phi}_T \mathbf{1}') = w_T. \end{aligned} \tag{3.1}$$

Hence $L_T = w_T = \prod_{t=1}^{T} (w_t / w_{t-1})$. From (3.1) it follows that

$$w_t = w_{t-1} (\boldsymbol{\phi}_{t-1} \boldsymbol{\Gamma} \mathbf{P}(x_t) \mathbf{1}'),$$

and so we conclude that

$$\log L_T = \sum_{t=1}^{T} \log(w_t / w_{t-1}) = \sum_{t=1}^{T} \log(\boldsymbol{\phi}_{t-1} \boldsymbol{\Gamma} \mathbf{P}(x_t) \mathbf{1}').$$

The computation of the log-likelihood is summarized below in the form of an algorithm. Note that $\mathbf{\Gamma}$ and $\mathbf{P}(x_t)$ are $m \times m$ matrices, $\mathbf{v}$ and $\boldsymbol{\phi}_t$ are vectors of length m, u is a scalar, and l is the scalar in which the log-likelihood is accumulated.

$$
\begin{aligned}
&\boldsymbol{\phi}_0 \leftarrow \boldsymbol{\delta};\ l \leftarrow 0 \\
&\text{for } t = 1, 2, \ldots, T \\
&\qquad \mathbf{v} \leftarrow \boldsymbol{\phi}_{t-1}\mathbf{\Gamma}\mathbf{P}(x_t) \\
&\qquad u \leftarrow \mathbf{v}\mathbf{1}' \\
&\qquad l \leftarrow l + \log u \\
&\qquad \boldsymbol{\phi}_t \leftarrow \mathbf{v}/u \\
&\text{return } l
\end{aligned}
$$

The required log-likelihood, $\log L_T$, is then given by the final value of l. This procedure will almost always prevent underflow. Clearly, minor variations of the technique are possible: the scale factor w_t could be chosen instead to be the largest element of the vector being scaled, or the mean of its elements (as opposed to the sum).

The algorithm is easily modified to compute the log-likelihood without assuming stationarity of the Markov chain. With $\boldsymbol{\delta}$ denoting the initial distribution, the more general algorithm is

$$
\begin{aligned}
&w_1 \leftarrow \boldsymbol{\delta}\mathbf{P}(x_1)\mathbf{1}';\ \boldsymbol{\phi}_1 \leftarrow \boldsymbol{\delta}\mathbf{P}(x_1)/w_1;\ l \leftarrow \log w_1 \\
&\text{for } t = 2, 3, \ldots, T \\
&\qquad \mathbf{v} \leftarrow \boldsymbol{\phi}_{t-1}\mathbf{\Gamma}\mathbf{P}(x_t) \\
&\qquad u \leftarrow \mathbf{v}\mathbf{1}' \\
&\qquad l \leftarrow l + \log u \\
&\qquad \boldsymbol{\phi}_t \leftarrow \mathbf{v}/u \\
&\text{return } l
\end{aligned}
$$

If the initial distribution happens to be the stationary distribution, the more general algorithm still applies.

The following code implements this last version of the algorithm in order to compute the log-likelihood of observations $x_1, \ldots, x_T$ under a Poisson–HMM with at least two states, transition probability matrix $\mathbf{\Gamma}$, vector of state-dependent means $\boldsymbol{\lambda}$, and initial distribution $\boldsymbol{\delta}$.

```
alpha        <- delta*dpois(x[1],lambda)
lscale       <- log(sum(alpha))
alpha        <- alpha/sum(alpha)
for (i in 2:T) {
  alpha      <- alpha %*% Gamma*dpois(x[i],lambda)
  lscale     <- lscale+log(sum(alpha))
  alpha      <- alpha/sum(alpha)
}
lscale
```

This code improves on that shown on p. 39 in that the vector of forward probabilities is scaled to have sum 1 at all times. But it is probably

unnecessary to scale the forward probabilities at time 1, and if one omits that part of the scaling, the algorithm and code simplify slightly.

3.3 Maximization of the likelihood subject to constraints

3.3.1 Reparametrization to avoid constraints

The elements of $\mathbf{\Gamma}$ and those of $\boldsymbol{\lambda}$, the vector of state-dependent means in a Poisson–HMM, are subject to non-negativity and other constraints. In particular, the row sums of $\mathbf{\Gamma}$ equal 1. Estimates of parameters should also satisfy such constraints. Thus, when maximizing the likelihood we need to solve a constrained optimization problem, not an unconstrained one.

Special-purpose software, such as NPSOL (Gill *et al.*, 1986) or the corresponding NAG routine E04UCF, can be used to maximize a function of several variables which are subject to constraints. The advice of Gill, Murray and Wright (1981, p. 267) is that it is 'rarely appropriate to alter linearly constrained problems'. However, depending on the implementation and the nature of the data, constrained optimization can be slow. For example, the constrained optimizer `constrOptim` available in **R** is acknowledged to be slow if the optimum lies on the boundary of the parameter space. We shall focus on the use of the unconstrained optimizer `nlm`. Exercise 3 explores the use of `constrOptim`, which can minimize a function subject to linear inequality constraints.

In general, there are two groups of constraints: those that apply to the parameters of the state-dependent distributions and those that apply to the parameters of the Markov chain. The first group of constraints depends on which state-dependent distribution(s) are chosen; for example, the 'success probability' of a binomial distribution lies between 0 and 1.

In the case of a Poisson–HMM the relevant constraints are:

- the means λ_i of the state-dependent distributions must, for $i = 1, \ldots, m$, be non-negative;

- the rows of the transition probability matrix $\mathbf{\Gamma}$ must add to 1, and all the parameters γ_{ij} must be non-negative.

Here the constraints can be imposed by making transformations. The transformation of the parameters λ_i is easy. Define $\eta_i = \log \lambda_i$, for $i = 1, \ldots, m$. Then $\eta_i \in \mathbb{R}$. After we have maximized the likelihood with respect to the unconstrained parameters, the constrained parameter estimates can be obtained by transforming back: $\widehat{\lambda}_i = \exp \widehat{\eta}_i$.

The reparametrization of the matrix $\mathbf{\Gamma}$ requires more work, but can be accomplished quite elegantly. Note that $\mathbf{\Gamma}$ has m^2 entries but only

$m(m-1)$ free parameters, as there are m row-sum constraints

$$\sum_{j=1}^{m} \gamma_{ij} = 1 \quad (i = 1, \ldots, m).$$

We shall show one possible transformation between the m^2 constrained probabilities γ_{ij} and $m(m-1)$ unconstrained real numbers $\tau_{ij}, i \neq j$.

For the sake of readability we display the case $m = 3$. We begin by defining the matrix

$$\mathbf{T} = \begin{pmatrix} - & \tau_{12} & \tau_{13} \\ \tau_{21} & - & \tau_{23} \\ \tau_{31} & \tau_{32} & - \end{pmatrix},$$

a matrix with $m(m-1)$ entries $\tau_{ij} \in \mathbb{R}$. Now let $g : \mathbb{R} \to \mathbb{R}^+$ be a strictly increasing function, for example,

$$g(x) = e^x \quad \text{or} \quad g(x) = \begin{cases} e^x & x \leq 0 \\ x + 1 & x \geq 0. \end{cases}$$

Define

$$\nu_{ij} = \begin{cases} g(\tau_{ij}) & \text{for } i \neq j \\ 1 & \text{for } i = j. \end{cases}$$

We then set $\gamma_{ij} = \nu_{ij} / \sum_{k=1}^{m} \nu_{ik}$ (for $i, j = 1, 2, \ldots, m$) and $\mathbf{\Gamma} = (\gamma_{ij})$. It is left to the reader as an exercise to verify that the resulting matrix $\mathbf{\Gamma}$ satisfies the constraints of a t.p.m. We shall refer to the parameters η_i and τ_{ij} as **working parameters**, and to the parameters λ_i and γ_{ij} as **natural parameters**.

Using the above transformations of $\mathbf{\Gamma}$ and $\boldsymbol{\lambda}$, we can perform the calculation of the likelihood-maximizing parameters in two steps.

1. Maximize L_T with respect to the working parameters $\mathbf{T} = \{\tau_{ij}\}$ and $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_m)$. These are all unconstrained.
2. Transform the estimates of the working parameters to estimates of the natural parameters:

$$\widehat{\mathbf{T}} \to \widehat{\mathbf{\Gamma}}, \quad \widehat{\boldsymbol{\eta}} \to \widehat{\boldsymbol{\lambda}}.$$

Consider $\mathbf{\Gamma}$ for the case $g(x) = e^x$ and general m. Here we have

$$\gamma_{ij} = \frac{\exp(\tau_{ij})}{1 + \sum_{k \neq i} \exp(\tau_{ik})}, \quad \text{for } i \neq j,$$

and the diagonal elements of $\mathbf{\Gamma}$ follow from the row sums of 1. The transformation in the opposite direction is

$$\tau_{ij} = \log\left(\frac{\gamma_{ij}}{1 - \sum_{k \neq i} \gamma_{ik}}\right) = \log\left(\gamma_{ij} / \gamma_{ii}\right), \quad \text{for } i \neq j.$$

This generalization of the logit and inverse logit transforms has long been used in the context of compositional data; see Aitchison (1982), where several other transforms are described as well.

We now display some relatively simple code that will transform natural parameters to working and vice versa. The code refers to a Poisson–HMM with $m \geq 2$ states, in which the Markov chain may, if appropriate, be assumed stationary. In that case the stationary distribution $\boldsymbol{\delta}$ is not supplied, but is computed when needed from the t.p.m. $\boldsymbol{\Gamma}$ by solving $\boldsymbol{\delta}(\mathbf{I}_m - \boldsymbol{\Gamma} + \mathbf{U}) = \mathbf{1}$; see p. 18 and Exercise 9(b) of Chapter 1. Otherwise $\boldsymbol{\delta}$ is treated as a (natural) parameter and transformed in order to remove the constraints $\delta_i \geq 0$ and $\sum_i \delta_i = 1$ (although there is a simpler route; see Section 4.2.4).

```
# Transform Poisson natural parameters to working parameters
pois.HMM.pn2pw <- function(m,lambda,gamma,delta=NULL,stationary=TRUE)
{
 tlambda <- log(lambda)
 foo     <- log(gamma/diag(gamma))
 tgamma  <- as.vector(foo[!diag(m)])
 if(stationary) {tdelta <-NULL} else {tdelta<-log(delta[-1]/delta[1])}
 parvect <- c(tlambda,tgamma,tdelta)
 return(parvect)
}

# Transform Poisson working parameters to natural parameters
pois.HMM.pw2pn <- function(m,parvect,stationary=TRUE)
{
 lambda        <- exp(parvect[1:m])
 gamma         <- diag(m)
 gamma[!gamma] <- exp(parvect[(m+1):(m*m)])
 gamma         <- gamma/apply(gamma,1,sum)
 if(stationary) {delta<-solve(t(diag(m)-gamma+1),rep(1,m))} else
                 {foo<-c(1,exp(parvect[(m*m+1):(m*m+m-1)]))
                  delta<-foo/sum(foo)}
 return(list(lambda=lambda,gamma=gamma,delta=delta))
}
```

For code which includes and uses these functions, and for some discussion thereof, see Sections A.1.1–A.1.4 and A.2.1.

3.3.2 Embedding in a continuous-time Markov chain

A different reparametrization is discussed by Zucchini and MacDonald (1998). In a continuous-time Markov chain on a finite state space, the transition probability matrix $\mathbf{P}_t$ over t time units is given by $\mathbf{P}_t = \exp(t\mathbf{Q})$, where $\mathbf{Q}$ is the matrix of transition intensities. The row sums of $\mathbf{Q}$ are 0, but the only constraint on the off-diagonal elements of $\mathbf{Q}$ is that they be non-negative. It is not in general the case that a discrete-time Markov chain is embeddable in a continuous-time Markov chain; see Exercise 11. But if one is prepared to assume that the discrete-time

Markov chain of interest is thus embeddable, the one-step transition probabilities of the discrete-time chain can then be parametrized via $\mathbf{\Gamma} = \exp(\mathbf{Q})$. This is effectively what one is doing if one uses the **R** package `msm` (Jackson *et al.*, 2003) to fit HMMs.

3.4 Other problems

3.4.1 Multiple maxima in the likelihood

The likelihood of an HMM is a complicated function of the parameters and frequently has several local maxima. The goal of course is to find the global maximum, but there is no simple method of determining in general whether a numerical maximization algorithm has reached the global maximum. Depending on the starting values, it can easily happen that the algorithm identifies a local, but not the global, maximum. This applies also to the main alternative method of estimation, the EM algorithm, which is discussed in Chapter 4. A sensible strategy is therefore to use a range of starting values for the maximization, and to see whether the same maximum is identified in each case.

3.4.2 Starting values for the iterations

It is often easy to find plausible starting values for some of the parameters of an HMM: for instance, if one seeks to fit a Poisson–HMM with two states, and the sample mean is 10, one could try 8 and 12, or 5 and 15, for the values of the two state-dependent means. More systematic strategies based on the quantiles of the observations are possible, however. For example, if the model has three states, use as the starting values of the state-dependent means the lower quartile, median and upper quartile of the observed counts.

It is less easy to guess values of the transition probabilities γ_{ij}. One strategy is to assign a common starting value (e.g. 0.01 or 0.05) to all the off-diagonal transition probabilities. A consequence of such a choice, perhaps convenient, is that the corresponding stationary distribution is uniform over the states; this follows by symmetry. Choosing good starting values for parameters tends to steer one away from numerical instability.

3.4.3 Unbounded likelihood

In the case of HMMs with continuous state-dependent distributions, just as in the case of independent mixtures (see Section 1.2.3), it may happen that the likelihood is unbounded in the vicinity of certain parameter combinations. As before, we suggest that, if this creates difficulties, one

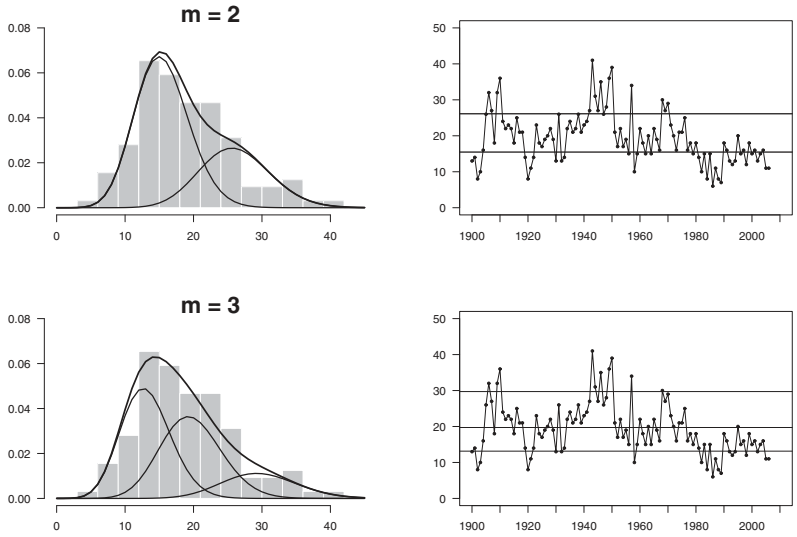


Figure 3.1 *Earthquakes series. Left: marginal distributions of Poisson–HMMs with two and three states, and their components, compared with a histogram of the observations. Right: the state-dependent means (horizontal lines) compared to the observations.*

maximizes the discrete likelihood instead of the joint density. This has the advantage in any case that it applies more generally to interval-censored data. Applications of this kind are described in Sections 17.4 and 17.5.

3.5 Example: earthquakes

Figure 3.1 shows the result of fitting (stationary) Poisson–hidden Markov models with two and three states to the earthquakes series by means of the unconstrained optimizer `nlm`. The relevant code (with starting values) appears in Section A.2.1. The two-state model is

$$\mathbf{\Gamma} = \begin{pmatrix} 0.9340 & 0.0660 \\ 0.1285 & 0.8715 \end{pmatrix},$$

with $\boldsymbol{\delta} = (0.6608, 0.3392)$, $\boldsymbol{\lambda} = (15.472, 26.125)$, and log-likelihood given by $l = -342.3183$. It is clear that the fitted (Markov-dependent) mixture of two Poisson distributions provides a much better fit to the marginal distribution of the observations than does a single Poisson distribution, but the fit can be further improved by using a mixture of three or four Poisson distributions.

The three-state model is

$$\mathbf{\Gamma} = \begin{pmatrix} 0.955 & 0.024 & 0.021 \\ 0.050 & 0.899 & 0.051 \\ 0.000 & 0.197 & 0.803 \end{pmatrix},$$

with $\boldsymbol{\delta} = (0.4436, 0.4045, 0.1519)$, $\boldsymbol{\lambda} = (13.146, 19.721, 29.714)$ and $l = -329.4603$. The four-state is

$$\mathbf{\Gamma} = \begin{pmatrix} 0.805 & 0.102 & 0.093 & 0.000 \\ 0.000 & 0.976 & 0.000 & 0.024 \\ 0.050 & 0.000 & 0.902 & 0.048 \\ 0.000 & 0.000 & 0.188 & 0.812 \end{pmatrix},$$

with $\boldsymbol{\delta} = (0.0936, 0.3983, 0.3643, 0.1439)$, $\boldsymbol{\lambda} = (11.283,\ 13.853,\ 19.695,\ 29.700)$, and $l = -327.8316$.

The means and variances of the marginal distributions of the four models compare as follows with those of the observations. By a one-state Poisson–HMM we mean a model that assumes that the observations are realizations of independent Poisson random variables with common mean.

	mean	variance
observations:	19.364	51.573
'one-state HMM':	19.364	19.364
two-state HMM:	19.086	44.523
three-state HMM:	18.322	50.709
four-state HMM:	18.021	49.837

As regards the autocorrelation functions of the models, that is, $\rho(k) = \mathrm{Corr}(X_{t+k}, X_t)$, we have the following results, valid for all $k \in \mathbb{N}$, based on the conclusions of Exercise 4 of Chapter 2:

- two states, $\rho(k) = 0.5713 \times 0.8055^k$;
- three states, $\rho(k) = 0.4447 \times 0.9141^k + 0.1940 \times 0.7433^k$;
- four states, $\rho(k) = 0.2332 \times 0.9519^k + 0.3682 \times 0.8174^k + 0.0369 \times 0.7252^k$.

In all these cases the ACF is just a linear combination of the kth powers of the eigenvalues other than 1 of the transition probability matrix.

For model selection, for example, choosing between competing models such as HMMs and independent mixtures, or choosing the number of components in either, see Section 6.1.

A phenomenon that is noticeable when one fits models with three or more states to relatively short series is that the estimates of one or more of the transition probabilities turn out to be very close to zero; see the three-state model above (one such probability, γ_{13}) and the four-state model (six of the 12 off-diagonal transition probabilities).

This phenomenon can be explained as follows. In a stationary Markov chain, the expected number of transitions from state i to state j in a series of T observations is $(T-1)\delta_i\gamma_{ij}$. For $\delta_3 = 0.152$ and $T = 107$ (as in our three-state model), this expectation will be less than 1 if $\gamma_{31} < 0.062$. In such a series, therefore, it is likely that if γ_{31} is fairly small there will be no transitions from state 3 to state 1, and so when we seek to estimate γ_{31} in an HMM the estimate is likely to be effectively zero. As m increases, the probabilities δ_i and γ_{ij} get smaller on average; this makes it increasingly likely that at least one estimated transition probability is effectively zero.

3.6 Standard errors and confidence intervals

Relatively little is known about the properties of the maximum likelihood estimators of HMMs; only asymptotic results are available. To exploit these results one requires estimates of the variance–covariance matrix of the estimators of the parameters. One can estimate the standard errors from the Hessian of the log-likelihood at the maximum, but this approach runs into difficulties when some of the parameters are on the boundary of their parameter space, which occurs quite often when HMMs are fitted. An alternative here is the parametric bootstrap, for which see Section 3.6.2. The algorithm is easy to code (see Section A.1.5), but the computations are time-consuming.

3.6.1 Standard errors via the Hessian

Although the point estimates $\widehat{\boldsymbol{\Theta}} = (\widehat{\boldsymbol{\Gamma}}, \widehat{\boldsymbol{\lambda}})$ are easy to compute, exact interval estimates are not available. Cappé *et al.* (2005, Chapter 12) show that, under certain regularity conditions, the MLEs of HMM parameters are consistent, asymptotically normal and efficient. Thus, if we can estimate the standard errors of the MLEs, then, using the asymptotic normality, we can also compute approximate confidence intervals. However, as pointed out by Frühwirth-Schnatter (2006, p. 53) in the context of independent mixture models, 'The regularity conditions are often violated, including cases of great practical concern, among them small data sets, mixtures with small component weights, and overfitting mixtures with too many components.' Furthermore, McLachlan and Peel (2000, p. 68) warn: 'In particular for mixture models, it is well known that the sample size n has to be very large before the asymptotic theory of maximum likelihood applies.'

With the above caveats in mind we can, in order to estimate the standard errors of the MLEs of an HMM, use the approximate Hessian of minus the log-likelihood at the minimum (e.g. as supplied by `nlm`). We

can invert it and so estimate the asymptotic variance–covariance matrix of the estimators of the parameters. A problem with this suggestion is that, if the parameters have been transformed, the Hessian available will be that which refers to the working parameters ϕ_i, not the original, more readily interpretable, natural parameters θ_i ($\boldsymbol{\Gamma}$ and $\boldsymbol{\lambda}$ in the case of a Poisson–HMM).

The situation is therefore that we have, at the minimum of $-l$, the Hessian with respect to the working parameters,

$$\mathbf{H} = -\left(\frac{\partial^2 l}{\partial\phi_i \partial\phi_j}\right),$$

but what we really need is the Hessian with respect to the natural parameters,

$$\mathbf{G} = -\left(\frac{\partial^2 l}{\partial\theta_i \partial\theta_j}\right).$$

There is, however, the following relationship between the two Hessians at the minimum:

$$\mathbf{H} = \mathbf{M}\mathbf{G}\mathbf{M}' \quad \text{and} \quad \mathbf{G}^{-1} = \mathbf{M}'\mathbf{H}^{-1}\mathbf{M}, \tag{3.2}$$

where $\mathbf{M}$ is defined by $m_{ij} = \partial\theta_j/\partial\phi_i$. See also Monahan (2011, p. 247) for this relation between the Hessians. (Note that all the derivatives appearing here are as evaluated at the minimum.) In the case of a Poisson–HMM, the elements of $\mathbf{M}$ are quite simple; see Exercise 7 for details.

With $\mathbf{M}$ at our disposal, we can use (3.2) to deduce $\mathbf{G}^{-1}$ from $\mathbf{H}^{-1}$, and use $\mathbf{G}^{-1}$ to find standard errors for the natural parameters, provided such parameters are not on the boundary of the parameter space. An alternative route to the standard errors with respect to the natural parameters which often works well, and is less laborious, is this. First find the MLE by solving the constrained optimization problem, then rerun the optimization without constraints, starting at or very close to the MLE. If the resulting estimate is the same as the MLE already found, the corresponding Hessian then directly supplies the standard errors with respect to the natural parameters. But if one is to make a normality assumption and base a confidence interval on it, such a normality assumption is more likely, but not guaranteed, to be reasonable on the working-parameter scale than on the (constrained) natural-parameter scale.

Furthermore, it is true in many applications that some of the estimated (natural) parameters lie on or very close to the boundary; this limits the usefulness of the above results. As already pointed out on p. 55, for series of moderate length the estimates of some transition probabilities are expected to be close to zero. This is true of $\widehat{\gamma}_{13}$ in the three-state model for the earthquakes series. An additional example of this type can be found in Section 17.3.2. In Section 19.2.1, several of the estimates of

the parameters in the state-dependent distributions are practically zero, their lower bound; see Table 19.1. The same phenomenon is apparent in Section 23.9.2; see Table 23.1.

Recursive computation of the Hessian

An alternative method of computing the Hessian is that of Lystig and Hughes (2002). They present the forward algorithm $\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1}\boldsymbol{\Gamma}\mathbf{P}(x_t)$ in a form which incorporates automatic or 'natural' scaling, and then extend that approach in order to compute (in a single pass, along with the log-likelihood) its Hessian and gradient with respect to the natural parameters, those we have denoted above by θ_i. Turner (2008) has used this approach in order to find the analytical derivatives needed to maximize HMM likelihoods directly by the Levenberg–Marquardt algorithm.

Although this may be a more efficient and more accurate method of computing the Hessian than the use of (3.2), it does not solve the fundamental problem that the use of the Hessian to compute standard errors (and thence confidence intervals) is unreliable if some of the parameters are on or near the boundary of their parameter space.

3.6.2 Bootstrap standard errors and confidence intervals

As an alternative to the techniques described in Section 3.6.1 one may use the **parametric bootstrap** (Efron and Tibshirani, 1993). Roughly speaking, the idea of the parametric bootstrap is to assess the properties of the model with parameters $\boldsymbol{\Theta}$ by using those of the model with parameters $\widehat{\boldsymbol{\Theta}}$. The following steps are performed to estimate the variance–covariance matrix of $\widehat{\boldsymbol{\Theta}}$.

1. Fit the model, i.e. compute $\widehat{\boldsymbol{\Theta}}$.

2.(a) Generate a sample, called a bootstrap sample, of observations from the fitted model, i.e. the model with parameters $\widehat{\boldsymbol{\Theta}}$. The length should be the same as the original number of observations.

(b) Estimate the parameters $\boldsymbol{\Theta}$ by $\widehat{\boldsymbol{\Theta}}^*$ for the bootstrap sample.

(c) Repeat steps (a) and (b) B times (with B 'large') and record the values $\widehat{\boldsymbol{\Theta}}^*$.

The variance–covariance matrix of $\widehat{\boldsymbol{\Theta}}$ is then estimated by the sample variance–covariance matrix of the bootstrap estimates $\widehat{\boldsymbol{\Theta}}^*(b)$, $b = 1, 2, \ldots, B$:

$$\widehat{\text{Var-Cov}}(\widehat{\boldsymbol{\Theta}}) = \frac{1}{B-1}\sum_{b=1}^{B}\left(\widehat{\boldsymbol{\Theta}}^*(b) - \widehat{\boldsymbol{\Theta}}^*(\cdot)\right)'\left(\widehat{\boldsymbol{\Theta}}^*(b) - \widehat{\boldsymbol{\Theta}}^*(\cdot)\right),$$

where $\widehat{\boldsymbol{\Theta}}^*(\cdot) = B^{-1}\sum_{b=1}^{B}\widehat{\boldsymbol{\Theta}}^*(b)$.

The parametric bootstrap requires code to generate realizations from a fitted model; for a Poisson–HMM this is given in Section A.1.5. Since code to fit models is available, that same code can be used to fit models to the bootstrap sample.

The bootstrap method can be used to estimate confidence intervals directly. In the example given in the next section we use the well-known 'percentile method' (Efron and Tibshirani, 1993); other options are available.

3.7 Example: the parametric bootstrap applied to the three-state model for the earthquakes data

Table 3.1 *Earthquakes data: bootstrap confidence intervals for the parameters of the three-state HMM.*

Parameter	MLE	90% conf. limits	
λ_1	13.146	11.463	14.253
λ_2	19.721	13.708	21.142
λ_3	29.714	20.929	33.160
γ_{11}	0.954	0.750	0.988
γ_{12}	0.024	0.000	0.195
γ_{13}	0.021	0.000	0.145
γ_{21}	0.050	0.000	0.179
γ_{22}	0.899	0.646	0.974
γ_{23}	0.051	0.000	0.228
γ_{31}	0.000	0.000	0.101
γ_{32}	0.197	0.000	0.513
γ_{33}	0.803	0.481	0.947
δ_1	0.444	0.109	0.716
δ_2	0.405	0.139	0.685
δ_3	0.152	0.042	0.393

A bootstrap sample of size 500 was generated from the three-state model for the earthquakes data, which appears on p. 55. In fitting models to the bootstrap samples, we noticed that, in two cases out of the 500, the starting values which we were in general using caused numerical instability or convergence problems. By choosing better starting values for these two cases we were able to fit models successfully and complete the exercise. The resulting sample of parameter values then produced the 90% confidence intervals for the parameters that are displayed in Table 3.1, and the estimated parameter correlations that are displayed in Table 3.2. What is noticeable is that the intervals for the state-dependent

Table 3.2 *Earthquakes data: bootstrap estimates of the correlations of the estimators of λ_i, for $i = 1, 2, 3$.*

	λ_1	λ_2	λ_3
λ_1	1.000	0.483	0.270
λ_2		1.000	0.688
λ_3			1.000

means λ_i overlap, the intervals for the stationary probabilities δ_i are very wide, and the estimators $\widehat{\lambda}_i$ are quite strongly correlated.

These results, in particular the correlations shown in Table 3.2, should make one wary of over-interpreting a model with nine parameters based on only 107 (dependent) observations. In particular, they suggest that the states are not well defined, and one should be cautious of attaching a substantive interpretation to them.

Exercises

1. Consider the following parametrization of the t.p.m. of an m-state Markov chain. Let $\tau_{ij} \in \mathbb{R}$ $(i, j = 1, 2, \ldots, m;\ i \neq j)$ be $m(m-1)$ arbitrary real numbers. Let $g : \mathbb{R} \to \mathbb{R}^+$ be some strictly increasing function, e.g. $g(x) = e^x$. Define ν_{ij} and γ_{ij} as on p. 51.

   (a) Show that the matrix $\mathbf{\Gamma}$ with entries γ_{ij} that are constructed in this way is a t.p.m., i.e. show that $0 \leq \gamma_{ij} \leq 1$ for all i and j, and that the row sums of $\mathbf{\Gamma}$ are equal to 1.

   (b) Given an $m \times m$ t.p.m. $\mathbf{\Gamma} = (\gamma_{ij})$, derive an expression for the parameters τ_{ij}, for $i, j = 1, 2, \ldots, m;\ i \neq j$.

2. The purpose of this exercise is to investigate the numerical behaviour of an 'unscaled' evaluation of the likelihood of an HMM, and to compare this with the behaviour of an alternative algorithm that applies scaling.

   Consider the stationary two-state Poisson–HMM with parameters

   $$\mathbf{\Gamma} = \begin{pmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{pmatrix}, \quad (\lambda_1, \lambda_2) = (1, 5).$$

   Compute the likelihood, L_{10}, of the following sequence of ten observations in two ways: 2, 8, 6, 3, 6, 1, 0, 0, 4, 7.

   (a) Use the unscaled method $L_{10} = \boldsymbol{\alpha}_{10}\mathbf{1}'$, where $\boldsymbol{\alpha}_0 = \boldsymbol{\delta}$ and $\boldsymbol{\alpha}_t =$

$\boldsymbol{\alpha}_{t-1}\mathbf{B}_t$;

$$\mathbf{B}_t = \boldsymbol{\Gamma} \begin{pmatrix} p_1(x_t) & 0 \\ 0 & p_2(x_t) \end{pmatrix};$$

and

$$p_i(x_t) = \lambda_i^{x_t} e^{-\lambda_i} / x_t!, \quad i = 1, 2;\ t = 1, 2, \ldots, 10.$$

Examine the numerical values of the vectors $\boldsymbol{\alpha}_0, \boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_{10}$.

(b) Use the first algorithm given in Section 3.2 to compute $\log L_{10}$. Examine the numerical values of the vectors $\boldsymbol{\phi}_0, \boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_{10}$. (It is easiest to store these vectors as rows in an 11×2 matrix.)

3. Use the **R** function `constrOptim` to fit HMMs with two to four states to the earthquakes data, and compare your models with those given in Section 3.5.

4. Another approach to the non-negativity and row-sum constraints on $\boldsymbol{\Gamma}$ is to convert them into 'box constraints', i.e. constraints of the form $a \leq \theta_i \leq b$. A box-constrained optimizer, such as `optim` in **R** with method `L-BFGS-B`, can then be used.

   Consider therefore the following transformation:

$$\begin{aligned} w_1 &= \sin^2 \theta_1, \\ w_i &= \left(\prod_{j=1}^{i-1} \cos^2 \theta_j \right) \sin^2 \theta_i, \quad i = 2, \ldots, m-1, \\ w_m &= \prod_{i=1}^{m-1} \cos^2 \theta_i. \end{aligned}$$

   Show how this transformation can be used to convert the constraints

$$\sum_{i=1}^{m} w_i = 1, \quad w_i \geq 0;\ i = 1, \ldots, m,$$

   into box constraints.

5. (a) Consider a stationary Markov chain, with t.p.m. $\boldsymbol{\Gamma}$ and stationary distribution $\boldsymbol{\delta}$. Show that the expected number of transitions from state i to state j in a series of T observations (i.e. in $T-1$ transitions) is $(T-1)\delta_i \gamma_{ij}$.

   Hint: this expectation is $\sum_{t=2}^{T} \Pr(X_{t-1} = i, X_t = j)$.

   (b) Show that, for $\delta_3 = 0.152$ and $T = 107$, this expectation is less than 1 if $\gamma_{31} < 0.062$.

6. Prove the relation (3.2) between the Hessian $\mathbf{H}$ of $-l$ with respect to the working parameters and the Hessian $\mathbf{G}$ of $-l$ with respect to the natural parameters, both being evaluated at the minimum of $-l$.

7. (See Section 3.6.1.) Consider an m-state Poisson–HMM, with natural parameters γ_{ij} and λ_i, and working parameters τ_{ij} and η_i defined as in Section 3.3.1, with $g(x) = e^x$.

(a) Show that

$$\begin{aligned}
\partial\gamma_{ij}/\partial\tau_{ij} &= \gamma_{ij}(1-\gamma_{ij}), \quad \text{for all } i,j;\\
\partial\gamma_{ij}/\partial\tau_{il} &= -\gamma_{ij}\gamma_{il}, \quad \text{for } j \neq l;\\
\partial\gamma_{ij}/\partial\tau_{kl} &= 0, \quad \text{for } i \neq k;\\
\partial\lambda_i/\partial\eta_i &= e^{\eta_i} = \lambda_i, \quad \text{for all } i.
\end{aligned}$$

(b) Hence find the matrix $\mathbf{M}$ in this case.

8. Modify the **R** code in Sections A.1.1–A.1.4 in order to fit a Poisson–HMM to interval-censored observations. (Assume that the observations are available as a $T \times 2$ matrix of which the first column contains the lower bound of the observation and the second the upper bound, possibly `Inf`.)

9. Verify the autocorrelation functions given on p. 55 for the two-, three- and four-state models for the earthquakes data. (Hint: use the **R** function `eigen` to find the eigenvalues and eigenvectors of the relevant transition probability matrices.)

10. Consider again the soap sales series introduced in Exercise 6 of Chapter 1.

    (a) Fit stationary Poisson–HMMs with two, three and four states to these data.

    (b) Find the marginal means and variances, and the ACFs, of these models, and compare them with their sample equivalents.

11. (Embeddability of discrete-time Markov chain in continuous-time.) It is not always possible to embed a discrete-time Markov chain uniquely in a continuous-time chain. That is, given a t.p.m. $\mathbf{\Gamma}$, there does not always exist a unique generator matrix $\mathbf{Q}$ such that $\mathbf{\Gamma} = \exp(\mathbf{Q})$. The following examples show that there may, even in simple cases, be more than one corresponding generator matrix, or there may be none.

    (a) (Example taken from Israel, Rosenthal and Wei (2001, p. 256).) Consider the matrices $\mathbf{\Gamma}$, $\mathbf{Q}_1$ and $\mathbf{Q}_2$ given by

$$\mathbf{\Gamma} = \frac{1}{5}\begin{pmatrix} 2 & 2 & 1 \\ 2 & 2 & 1 \\ 2 & 2 & 1 \end{pmatrix} - \frac{e^{-4\pi}}{5}\begin{pmatrix} -3 & 2 & 1 \\ 2 & -3 & 1 \\ 2 & 2 & -4 \end{pmatrix},$$

$$\mathbf{Q}_1 = 2\pi\begin{pmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \\ 2 & 0 & -2 \end{pmatrix}, \qquad \mathbf{Q}_2 = \frac{4\pi}{5}\begin{pmatrix} -3 & 2 & 1 \\ 2 & -3 & 1 \\ 2 & 2 & -4 \end{pmatrix}.$$

    Verify that $\exp(\mathbf{Q}_1) = \exp(\mathbf{Q}_2) = \mathbf{\Gamma}$.

(b) Theorem 3.1 of Israel *et al.* (2001, p. 249) states the following.

Let $\mathbf{P}$ *be a transition [probability] matrix, and suppose that*

i. $\det(\mathbf{P}) \leq 0$, *or*
ii. $\det(\mathbf{P}) > \prod_i p_{ii}$, *or*
iii. there are states i *and* j *such that* j *is accessible from* i, *but* $p_{ij} = 0$.

Then there does not exist an exact generator for $\mathbf{P}$.

Use this theorem to conclude that there is no corresponding generator matrix for the following t.p.m.s:

$$\mathbf{\Gamma} = \begin{pmatrix} 0.4 & 0.6 \\ 0.5 & 0.5 \end{pmatrix}, \qquad \mathbf{\Gamma} = \begin{pmatrix} 0.9 & 0.1 & 0 \\ 0.1 & 0.8 & 0.1 \\ 0.1 & 0.1 & 0.8 \end{pmatrix}.$$

APPENDIX A

Examples of R code

In this Appendix we present **R** functions which will perform basic analyses in respect of a Poisson–HMM, plus several examples of code which uses these functions to analyse the earthquakes series. It is intended that this code will be available on the web pages of the book, currently `www.hmms-for-time-series.de`, along with further examples of the use of these functions.

A.1 The functions

In Sections A.1.1–A.1.4 we give four functions which, for a Poisson–HMM:

- implement the transformations described in Section 3.3.1, and their inverse;
- compute minus the log-likelihood of a Poisson–HMM for given values of the working parameters; and
- estimate the (natural) parameters of the model by using numerical minimization of minus the log-likelihood.

The purpose of the transformation of the natural parameters (Poisson means, transition probabilities and, if appropriate, the initial distribution of the Markov chain) to working parameters is simply to convert a constrained optimization problem to an unconstrained one. The separation of the transformation functions from the estimation functions is not essential, but has the advantage of providing simple, easily checkable tools that can readily be modified to cope with constraints of a different form.

The code will cope with both the stationary and the non-stationary cases. The former case is when one assumes that the Markov chain is not merely homogeneous but stationary, and is the default. Note that, in the non-stationary case, the function `pois.HMM.pn2pw` cannot be expected to work if any component of `delta` is exactly zero. In practice this is not a problem, unless one explicitly provides such a `delta` as starting value.

These functions allow for the special case of a 'one-state HMM', and for the possibility that there are observations missing; missing data are treated as described in Section 2.3.4.

Notice, in particular, the form of the output of the code in Section A.1.4: a list consisting of

- the number of states (m);
- the estimates of $\boldsymbol{\lambda}$, $\boldsymbol{\Gamma}$ and $\boldsymbol{\delta}$;
- the 'convergence code' of the optimizer `nlm`, an integer from 1 to 5; and
- the values of minus the log-likelihood, AIC and BIC.

This is the form in which all the functions given in Sections A.1.5–A.1.14 expect the input `mod` to be supplied.

A.1.1 Transforming natural parameters to working

```
pois.HMM.pn2pw <- function(m,lambda,gamma,delta=NULL,
                           stationary=TRUE)
{
 tlambda <- log(lambda)
 if(m==1) return(tlambda)
 foo     <- log(gamma/diag(gamma))
 tgamma  <- as.vector(foo[!diag(m)])
 if(stationary) {tdelta  <- NULL}
   else {tdelta <- log(delta[-1]/delta[1])}
 parvect <- c(tlambda,tgamma,tdelta)
 return(parvect)
}
```

A.1.2 Transforming working parameters to natural

```
pois.HMM.pw2pn <- function(m,parvect,stationary=TRUE)
{
 lambda        <- exp(parvect[1:m])
 gamma         <- diag(m)
 if (m==1) return(list(lambda=lambda,gamma=gamma,delta=1))
 gamma[!gamma] <- exp(parvect[(m+1):(m*m)])
 gamma         <- gamma/apply(gamma,1,sum)
 if(stationary){delta<-solve(t(diag(m)-gamma+1),rep(1,m))}
   else {foo<-c(1,exp(parvect[(m*m+1):(m*m+m-1)]))
   delta<-foo/sum(foo)}
 return(list(lambda=lambda,gamma=gamma,delta=delta))
}
```

A.1.3 Computing minus the log-likelihood from the working parameters

Notice that the vector `foo` is scaled at each iteration to have sum 1. When scaling is done in this way, the scaled likelihood that emerges at the end of the loop is automatically 1. Hence the log-likelihood we seek is just the final value of `lscale`, the sum of the logs of all the quantities by which the vector `foo` has been divided.

```
pois.HMM.mllk <- function(parvect,x,m,stationary=TRUE,...)
{
 if(m==1) return(-sum(dpois(x,exp(parvect),log=TRUE)))
 n         <- length(x)
 pn        <- pois.HMM.pw2pn(m,parvect,stationary=stationary)
 foo       <- pn$delta*dpois(x[1],pn$lambda)
 sumfoo    <- sum(foo)
 lscale    <- log(sumfoo)
 foo       <- foo/sumfoo
 for (i in 2:n)
   {
   if(!is.na(x[i])){P<-dpois(x[i],pn$lambda)}
     else {P<-rep(1,m)}
   foo     <- foo %*% pn$gamma*P
   sumfoo  <- sum(foo)
   lscale  <- lscale+log(sumfoo)
   foo     <- foo/sumfoo
   }
 mllk      <- -lscale
 return(mllk)
}
```

A.1.4 Computing the MLEs, given starting values for the natural parameters

```
pois.HMM.mle <-
function(x,m,lambda0,gamma0,delta0=NULL,stationary=TRUE,...)
{
 parvect0  <- pois.HMM.pn2pw(m,lambda0,gamma0,delta0,
         stationary=stationary)
 mod         <- nlm(pois.HMM.mllk,parvect0,x=x,m=m,
         stationary=stationary)
 pn          <- pois.HMM.pw2pn(m=m,mod$estimate,stationary=stationary)
 mllk        <- mod$minimum
 np          <- length(parvect0)
 AIC         <- 2*(mllk+np)
 n           <- sum(!is.na(x))
 BIC         <- 2*mllk+np*log(n)
 list(m=m,lambda=pn$lambda,gamma=pn$gamma,delta=pn$delta,
         code=mod$code,mllk=mllk,AIC=AIC,BIC=BIC)
}
```

A.1.5 Generating a sample

This function generates a realization, of length `ns`, of the HMM `mod`.

```
pois.HMM.generate_sample  <- function(ns,mod)
{
 mvect                    <- 1:mod$m
 state                    <- numeric(ns)
 state[1]                 <- sample(mvect,1,prob=mod$delta)
 for (i in 2:ns) state[i] <- sample(mvect,1,
                                 prob=mod$gamma[state[i-1],])
 x                        <- rpois(ns,lambda=mod$lambda[state])
 return(x)
}
```

A.1.6 Global decoding by the Viterbi algorithm

Given the model `mod` and observed series `x`, this function performs global decoding as described in Section 5.4.2.

```
pois.HMM.viterbi<-function(x,mod)
 {
 n              <- length(x)
 xi             <- matrix(0,n,mod$m)
 foo            <- mod$delta*dpois(x[1],mod$lambda)
 xi[1,]         <- foo/sum(foo)
 for (i in 2:n)
  {
  foo<-apply(xi[i-1,]*mod$gamma,2,max)*dpois(x[i],mod$lambda)
  xi[i,] <- foo/sum(foo)
  }
 iv<-numeric(n)
 iv[n]     <-which.max(xi[n,])
 for (i in (n-1):1)
   iv[i] <- which.max(mod$gamma[,iv[i+1]]*xi[i,])
 return(iv)
}
```

A.1.7 Computing log(forward probabilities)

Given data `x` and model `mod`, this function uses the recursion $\boldsymbol{\alpha}_{t+1} = \boldsymbol{\alpha}_t \boldsymbol{\Gamma} \mathbf{P}(x_{t+1})$ (see Section 4.1.1) to find all the vectors of forward probabilities, in logarithmic form. A matrix (`lalpha`) is returned. Scaling of the same kind as used for likelihood computations is implemented.

```
pois.HMM.lforward<-function(x,mod)
{
 n              <- length(x)
 lalpha         <- matrix(NA,mod$m,n)
 foo            <- mod$delta*dpois(x[1],mod$lambda)
 sumfoo         <- sum(foo)
 lscale         <- log(sumfoo)
 foo            <- foo/sumfoo
 lalpha[,1]     <- lscale+log(foo)
 for (i in 2:n)
  {
  foo           <- foo%*%mod$gamma*dpois(x[i],mod$lambda)
  sumfoo        <- sum(foo)
  lscale        <- lscale+log(sumfoo)
  foo           <- foo/sumfoo
  lalpha[,i]    <- log(foo)+lscale
  }
 return(lalpha)
}
```

A.1.8 Computing log(backward probabilities)

Similarly, this function uses the recursion $\boldsymbol{\beta}'_t = \boldsymbol{\Gamma} \mathbf{P}(x_{t+1}) \boldsymbol{\beta}'_{t+1}$ (see Section 4.1.2) to find all the vectors of backward probabilities, in logarithmic form.

```
pois.HMM.lbackward<-function(x,mod)
{
 n          <- length(x)
 m          <- mod$m
 lbeta      <- matrix(NA,m,n)
 lbeta[,n]  <- rep(0,m)
 foo        <- rep(1/m,m)
 lscale     <- log(m)
 for (i in (n-1):1)
  {
  foo        <- mod$gamma%*%(dpois(x[i+1],mod$lambda)*foo)
  lbeta[,i]  <- log(foo)+lscale
  sumfoo     <- sum(foo)
  foo        <- foo/sumfoo
  lscale     <- lscale+log(sumfoo)
  }
 return(lbeta)
}
```

A.1.9 Conditional probabilities

Equation (5.3) is used here to find, for all t, the conditional probabilities $\Pr\left(X_t = x \mid \mathbf{X}^{(-t)}\right)$, given the data `x` and model `mod`. The input `xc` specifies the range of x-values for which these probabilities are required. The use of the shifts `lafact` and `lbfact` is intended to prevent underflow in the exponentiations. The row vector `foo` is then normalized to have sum 1.

```
#==Conditional probability that observation at time t equals
#  xc, given all observations other than that at time t.
#  Note: xc is a vector and the result (dxc) is a matrix.
pois.HMM.conditional <- function(xc,x,mod)
{
 n          <- length(x)
 m          <- mod$m
 nxc        <- length(xc)
 dxc        <- matrix(NA,nrow=nxc,ncol=n)
 Px         <- matrix(NA,nrow=m,ncol=nxc)
 for (j in 1:nxc) Px[,j] <-dpois(xc[j],mod$lambda)
 la         <- pois.HMM.lforward(x,mod)
 lb         <- pois.HMM.lbackward(x,mod)
 la         <- cbind(log(mod$delta),la)
 lafact     <- apply(la,2,max)
 lbfact     <- apply(lb,2,max)
 for (i in 1:n)
  {
  foo       <-
   (exp(la[,i]-lafact[i])%*%mod$gamma)*exp(lb[,i]-lbfact[i])
  foo       <- foo/sum(foo)
  dxc[,i]   <- foo%*%Px
  }
 return(dxc)
}
```

A.1.10 Pseudo-residuals

The function `pois.HMM.conditional` in Section A.1.9 is now used to find ordinary normal pseudo-residuals as described in Sections 6.2.1 and 6.2.2. The form of the output is this (in the notation and terminology of those sections). For each time t from 1 to `n`, the function provides the lower and upper normal pseudo-residuals, z_t^- and z_t^+, and the mid-pseudo-residual $z_t^m = \Phi^{-1}\left((u_t^- + u_t^+)/2\right)$. Plots such as those in the top row of Figure 6.5 can then be produced.

```
pois.HMM.pseudo_residuals <- function(x,mod)
{
 n         <- length(x)
 cdists    <- pois.HMM.conditional(xc=0:max(x),x,mod)
 cumdists <- rbind(rep(0,n),apply(cdists,2,cumsum))
 ulo <- uhi <- rep(NA,n)
 for (i in 1:n)
  {
  ulo[i]  <- cumdists[x[i]+1,i]
  uhi[i]  <- cumdists[x[i]+2,i]
  }
 umi        <- 0.5*(ulo+uhi)
 npsr      <- qnorm(rbind(ulo,umi,uhi))
 return(npsr)
}
```

A.1.11 State probabilities

Here we compute probabilities $\Pr(C_t = i \mid \mathbf{X}^{(T)} = \mathbf{x}^{(T)})$, for $t \in \{1, 2, \ldots, T\}$. See Section 5.4.1, equation (5.6) in particular. The functions `pois.HMM.lforward` and `pois.HMM.lbackward` are used and, as elsewhere, a shift (by `c`) is used to counteract underflow in the exponentiation.

```
pois.HMM.state_probs <- function(x,mod)
{
 n           <- length(x)
 la          <- pois.HMM.lforward(x,mod)
 lb          <- pois.HMM.lbackward(x,mod)
 c           <- max(la[,n])
 llk         <- c+log(sum(exp(la[,n]-c)))
 stateprobs <- matrix(NA,ncol=n,nrow=mod$m)
 for (i in 1:n) stateprobs[,i]<-exp(la[,i]+lb[,i]-llk)
 return(stateprobs)
}
```

A.1.12 State prediction

Here we use equation (5.12) and the function `pois.HMM.lforward` to compute probabilities $\Pr(C_{T+h} = i \mid \mathbf{X}^{(T)} = \mathbf{x}^{(T)})$, for a range of values $h \in \mathbb{N}$.

```
# Note that state output 'statepreds' is a matrix even if h=1.
pois.HMM.state_prediction <- function(h=1,x,mod)
{
 n          <- length(x)
 la         <- pois.HMM.lforward(x,mod)
 c          <- max(la[,n])
 llk        <- c+log(sum(exp(la[,n]-c)))
 statepreds <- matrix(NA,ncol=h,nrow=mod$m)
 foo <- exp(la[,n]-llk)
 for (i in 1:h){
  foo<-foo%*%mod$gamma
  statepreds[,i]<-foo
  }
 return(statepreds)
}
```

A.1.13 Local decoding

See Section 5.4.1. This is a straightforward application of the function `pois.HMM.state_probs` in Section A.1.11.

```
pois.HMM.local_decoding <- function(x,mod)
{
 n          <- length(x)
 stateprobs <- pois.HMM.state_probs(x,mod)
 ild        <- rep(NA,n)
 for (i in 1:n) ild[i]<-which.max(stateprobs[,i])
 ild
}
```

A.1.14 Forecast probabilities

Given `x` and `mod`, equation (5.5) is used here to find forecast probabilities $\Pr(X_{T+h} = x \mid \mathbf{X}^{(T)} = \mathbf{x}^{(T)})$. The range of x-values for which these probabilities are required is specified by the input `xf` (e.g. 0:50 or 0:45, as in Section A.2.2) and the range of times for which they are required by `h`.

```
# Note that the output 'dxf' is a matrix.
pois.HMM.forecast <- function(xf,h=1,x,mod)
{
 n        <- length(x)
 nxf      <- length(xf)
 dxf      <- matrix(0,nrow=h,ncol=nxf)
 foo      <- mod$delta*dpois(x[1],mod$lambda)
 sumfoo   <- sum(foo)
 lscale   <- log(sumfoo)
 foo      <- foo/sumfoo
 for (i in 2:n)
   {
   foo    <- foo%*%mod$gamma*dpois(x[i],mod$lambda)
   sumfoo <- sum(foo)
   lscale <- lscale+log(sumfoo)
   foo    <- foo/sumfoo
   }
```

```
  for (i in 1:h)
   {
   foo     <- foo%*%mod$gamma
   for (j in 1:mod$m) dxf[i,] <- dxf[i,] +
           foo[j]*dpois(xf,mod$lambda[j])
    }
 return(dxf)
}
```

A.2 Examples of code using the above functions

The purpose of this section is to provide a few examples of code which uses the functions in Section A.1, so it will be convenient to load those functions first. Further examples of the use of those functions will be provided on the web pages.

A.2.1 Fitting Poisson–HMMs to the earthquakes series

This code uses all the functions in Sections A.1.1–A.1.4 in order to fit to the earthquakes series both stationary and non-stationary models, with two, three and four states.

```
dat <- read.table(
 "http://www.hmms-for-time-series.de/second/data/earthquakes.txt")
#(or set your own path)
x   <-dat[,2]
d   <-dat[,1]
n   <-length(x)
#====================================== fit 2-state HMM
m<-2
lambda0<-c(15,25)
gamma0<-matrix(
c(
0.9,0.1,
0.1,0.9
),m,m,byrow=TRUE)
mod2s<-pois.HMM.mle(x,m,lambda0,gamma0,stationary=TRUE)
delta0<-c(1,1)/2
mod2h<-pois.HMM.mle(x,m,lambda0,gamma0,delta=delta0,stationary=FALSE)
mod2s; mod2h
#====================================== fit 3-state HMM
m<-3
lambda0<-c(10,20,30)
gamma0<-matrix(
c(
 0.8,0.1,0.1,
 0.1,0.8,0.1,
 0.1,0.1,0.8
),m,m,byrow=TRUE)
mod3s<-pois.HMM.mle(x,m,lambda0,gamma0,stationary=TRUE)
delta0 <- c(1,1,1)/3
mod3h<-pois.HMM.mle(x,m,lambda0,gamma0,delta=delta0,stationary=FALSE)
mod3s; mod3h
#====================================== fit 4-state HMM
m<-4
lambda0<-c(10,15,20,30)
gamma0<-matrix(
```

```
  c(
    0.85,0.05,0.05,0.05,
    0.05,0.85,0.05,0.05,
    0.05,0.05,0.85,0.05,
    0.05,0.05,0.05,0.85
  ),m,m,byrow=TRUE)
mod4s<-pois.HMM.mle(x,m,lambda0,gamma0,stationary=TRUE)
delta0<-c(1,1,1,1)/4
mod4h<-pois.HMM.mle(x,m,lambda0,gamma0,delta=delta0,stationary=FALSE)
mod4s; mod4h
```

A.2.2 Forecast probabilities

Here we demonstrate the use of `pois.HMM.forecast`, and plot the forecast distributions. Note that the output of that function is returned as a matrix.

```
#=== Use it for 1-step-ahead and plot the forecast distribution.
h<-1
xf<-0:50
forecasts<-pois.HMM.forecast(xf,h,x,mod3s)
fc<-forecasts[1,]
par(mfrow=c(1,1),las=1)
plot(xf,fc,type="h",
main=paste("Earthquake series: forecast distribution for", d[n]+1),
xlim=c(0,max(xf)),ylim=c(0,0.12),xlab="count",ylab="probability",lwd=3)

#=== Forecast 1-4 steps ahead and plot these.
h<-4
xf<-0:45
forecasts<-pois.HMM.forecast(xf,h,x,mod3s)

par(mfrow=c(2,2),las=1)
for (i in 1:4)
  {
fc<-forecasts[i,]
plot(xf,fc,type="h",main=paste("Forecast distribution for", d[n]+i),
xlim=c(0,max(xf)),ylim=c(0,0.12),xlab="count",ylab="probability",lwd=3)
}

#=== Compute the marginal distribution (called "dstat" below)
#    for mod3h.
#=== This is also the long-term forecast.
m<-3.

lambda<-mod3h$lambda
delta<-solve(t(diag(m)-mod3h$gamma+1),rep(1,m))
dstat<-numeric(length(xf))
for (j in 1:m) dstat <- dstat + delta[j]*dpois(xf,lambda[j])

#=== Compare the 50-year-ahead forecast with the long-term forecast.
h<-50
xf<-0:45
forecasts<-pois.HMM.forecast(xf,h,x,mod3h)
fc<-forecasts[h,]
par(mfrow=c(1,1),las=1)
plot(xf,fc,type="h",
main=paste("Forecast distribution for", d[n]+h),
xlim=c(0,max(xf)),ylim=c(0,0.12),xlab="count",ylab="probability",lwd=3)
lines(xf,dstat,col="gray",lwd=3)
```