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

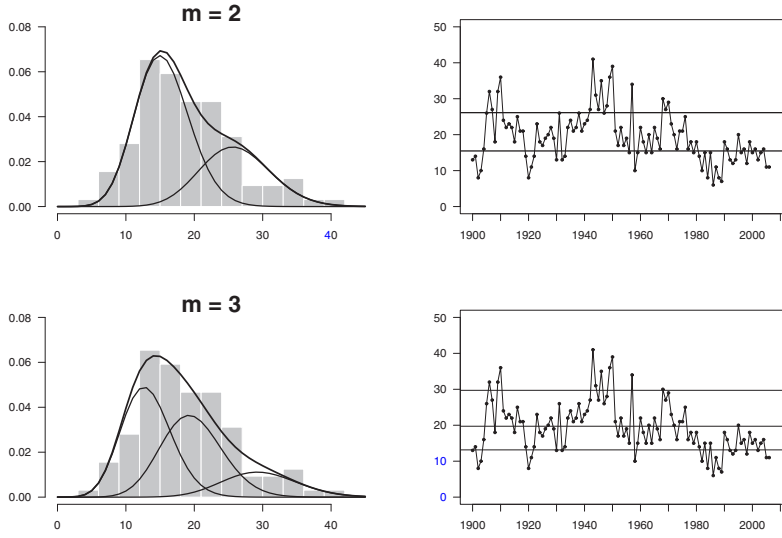

Figure 3.1 *Earthquakes series. Left: marginal distributions of Poisson–HMMs with two and three states, and their components, compared with a histogram of the observations. Right: the state-dependent means (horizontal lines) compared to the observations.*

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:

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