BAYESIAN MIXTURE MODELS

BAYESIAN MIXTURE MODELS

By

Zhihui Liu, B.Sc.

A Thesis

Submitted to the School of Graduate Studies

in Partial Fulfilment of the Requirements

for the Degree

Master of Science

McMaster University

MASTER OF SCIENCE (2010) McMaster University

(Statistics) Hamilton, Ontario

TITLE: Bayesian Mixture Models

AUTHOR: Zhihui Liu

B.Sc. (McMaster University)

SUPERVISOR: Professor Peter D. M. Macdonald

NUMBER OF PAGES: cvii, 107

Abstract

Mixture distributions are typically used to model data in which each observation belongs to one of some number of different groups. They also provide a convenient and flexible class of models for density estimation. When the number of components k is assumed known, the Gibbs sampler can be used for Bayesian estimation of the component parameters. We present the implementation of the Gibbs sampler for mixtures of Normal distributions and show that spurious modes can be avoided by introducing a Gamma prior in the Kiefer-Wolfowitz example.

Adopting a Bayesian approach for mixture models has certain advantages; it is not without its problems. One typical problem associated with mixtures is non-identifiability of the component parameters. This causes label switching in the Gibbs sampler output and makes inference for the individual components meaningless. We show that the usual approach to this problem by imposing simple identifiability constraints on the mixture parameters is sometimes inadequate, and present an alternative approach by arranging the mixture components in order of non-decreasing means

whilst choosing priors that are slightly more informative. We illustrate the success of our approach on the fishery example.

When the number of components k is considered unknown, more sophisticated methods are required to perform the Bayesian analysis. One method is the Reversible Jump MCMC algorithm described by Richardson and Green (1997), which they applied to univariate Normal mixtures. Alternatively, selection of k can be based on a comparison of models fitted with different numbers of components by some joint measures of model fit and model complexity. We review these methods and illustrate how to use them to compare competing mixture models using the acidity data.

We conclude with some suggestions for further research.

Acknowledgement

My warmest and unreserved thanks go to my supervisor Professor Peter Macdonald, for giving me the freedom to make my own mistakes and discoveries, whilst ensuring that I remained on the right track. His vast knowledge of mixture models has constantly guided me, his enthusiasm for applied statistics has always inspired me, and his faith in me was vital for the completion of my MSc.

I would like to thank Professors Aaron Childs and Eleanor Pullenayegum for serving on my examination committee, providing valuable advice, and being extremely supportive. These were particularly instrumental in the final revision of this thesis.

Still at McMaster, I must thank Professors Angelo Canty, Roman Viveros, Shui Feng and N. Balakrishnan for their kindness and help. The financial support from the Department of Mathematics and Statistics is gratefully acknowledged.

I cannot forget to thank my friends for providing me with good food, entertainment and companionship along the way, in particular Feng and Pete. I thank Jason, Yu Qing, Grant, and Chuang for helping me move in August, as well as David and Dan for many wonderful sessions on Sunday.

My parents deserve not only thanks for unconditional support during all phases of my life, but also apologies for pursuing my studies so far away from home for so many years. To them I dedicate this thesis.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introduction

While based on elementary distributions, mixture models provide a much wider range of modeling possibilities than their components. They date back to the work of Newcomb (1886) and Pearson (1894), but advances in computational methods such as maximum likelihood (Baum *et al.*, 1970), the EM algorithm (Dempster *et al.*, 1977) and Markov chain Monte Carlo (MCMC) Bayesian methods (Diebolt and Robert, 1994) have substantially expanded the areas of their applications. These areas include genomics (Broët *et al.*, 2002; Fraley and Raftery, 2002), epidemiology (Schlattmann and Böhning, 1993; Green and Richardson, 2002), econometrics (Jedidi *et al.*, 1997; Allenby *et al.*, 1998), macroeconomics (Hamilton, 1989; Lesage, 1992), finance (Lam-

oureux and Lastsrapes, 1994; Robert *et al.*, 2000; Kaufman and Frühwirth-Schnatter, 2002), and so on.

A number of recent books on mixtures that deserve mentioning are Lindsay (1995), Böhning (1999), McLachlan and Peel (2000), Frühwirth-Schnatter (2006) and Schlattmann (2009), which update the previous books by Everitt and Hand (1981), Titterington *et al.* (1985), and McLachlan and Basford (1988). A diversity of publications on mixtures before and after 2003 are reviewed in Böhning and Seidel (2003) and Böhning *et al.* (2007). As the list of references indicates, there is a very large literature on methodology for and applications of finite mixture models. We try to refer to as many of the technical papers as possible at appropriate points in this thesis.

1.2 Basic Definitions

Definition 1.2.1. *Suppose that a random variable or vector X takes values in a sample space $\mathcal{X}$, and that its distribution can be represented by a probability density function (or mass function in the case of discrete $\mathcal{X}$) of the form*

$$g(x) = \sum_{j=1}^{k} w_j f_j(x) \quad (x \in \mathcal{X}), \tag{1.1}$$

where $0 \leq w_j \leq 1, \ \sum_{j=1}^{k} w_j = 1, \ j = 1, \ldots, k, \ k > 1.$

We say that X has a finite mixture distribution and that $g(\cdot)$ defined by (1.1) is

a finite mixture density function. The parameters $(w_1, \ldots, w_k)$ *are called the mixing weights or mixing proportions, and* $f_1(x), \ldots, f_k(x)$ *the component densities of the mixture.*

If the components $f_j(\cdot)$ come from a parametric family, with unknown parameters $\boldsymbol{\theta}_j$, then the parametric mixture model is

$$g(x|\boldsymbol{\Psi}) = \sum_{j=1}^{k} w_j f_j(x|\boldsymbol{\theta}_j), \tag{1.2}$$

We denote the collection of all distinct parameters occurring in the component densities by $\boldsymbol{\theta}$, and the complete collection of all distinct parameters occurring in the mixture model by $\boldsymbol{\Psi}$, following the notation in Titterington *et al.* (1985, §1.1).

Consider a two-component mixture model of the form

$$g(x|\boldsymbol{\Psi}) = w\phi(x|\mu_1, \sigma_1^2) + (1-w)\phi(x|\mu_2, \sigma_2^2), \tag{1.3}$$

where $\phi(x|\mu_j, \sigma_j^2)$, $j = 1, 2$, denotes a univariate Normal density with mean μ_j and variance σ_j^2. In this case, $w_1 = w$, $w_2 = (1-w)$, $\boldsymbol{\theta}_1 = (\mu_1, \sigma_1^2)$, $\boldsymbol{\theta}_2 = (\mu_2, \sigma_2^2)$, $\boldsymbol{\theta} = (\mu_1, \mu_2, \sigma_1^2, \sigma_2^2)$, and $\boldsymbol{\Psi} = (w, \mu_1, \mu_2, \sigma_1^2, \sigma_2^2)$.

In many applications, the component densities in (1.2) belong to the same para-

metric family, and the finite mixture density function can be written as

$$g(x|\Psi) = \sum_{j=1}^{k} w_j f(x|\theta_j), \tag{1.4}$$

where $f(\cdot|\theta)$ denotes a generic member of the parametric family, k is called the *number of components*, and $(\theta_1, \ldots, \theta_k)$ are called the *component parameters*. Note however, it is not required that all the components belong to the same parametric family.

Such formulations are of interest in a context where there is a reason to assume that g is genuinely a mixture of a fixed number of components. For instance, we may believe that the x_i's are drawn from a heterogeneous population which is composed of subsets, and the distribution of members of the jth subset is $f_j(x|\theta_j)$. Consider an example where the x_i's are measures of size of a species of fish caught in the fall. If these fish spawn only in the spring, then the population will be composed of one group that are around 6 months old, another that are about 18 months old, and so on. Each age cohort defines a subset of the population with its own size distribution. Even where the population is known to be heterogeneous in this way, the number of mixture components may be unknown and even potentially unbounded. In this case, one may be interested in developing a nonparametric formulation, allowing the mixture to have an arbitrary and unlimited number of components.

A difference between the two motivations for mixture is that in the nonparametric context we are primarily interested in inference about g, and the components of the

mixture are not in themselves important. The mixture is simply a convenient way to achieve a flexible representation for g, without constraining it to belong to a parametric family. In contrast, where there is a scientific basis for a mixture model, the components $f_j(\cdot|\boldsymbol{\theta}_j)$, the mixing weights $\boldsymbol{w} = (w_1, \ldots, w_k)$ and even the number of components k are of intrinsic interest.

An important concept associated with the mixture model is identifiability. In order to estimate Ψ, it is necessary that they should be identifiable. In general, a parametric distribution family is said to be identifiable if distinct parametric values determine distinct members of the family. This is defined similarly for mixture models.

Definition 1.2.2. *Let $f(x|\Psi) = \sum_{j=1}^{k} w_j f(x|\boldsymbol{\theta}_j)$ be the member of a parametric family of finite mixture models. This class of finite mixtures is said to be identifiable if for any two members $f(x|\Psi)$ and $f(x|\Psi^\star)$,*

$$\sum_{j=1}^{k} w_j f(x|\boldsymbol{\theta}_j) = \sum_{j=1}^{k^\star} w_j^\star f(x|\boldsymbol{\theta}_j^\star)$$

if and only if $k = k^\star, w_j = w_j^\star$ and $\boldsymbol{\theta}_j = \boldsymbol{\theta}_j^\star$ after permuting the component labels.

Teicher (1963) showed that except for mixtures of uniform distributions, many finite mixtures of continuous densities are identifiable. These results were extended to multivariate families such as multivariate mixtures of Normals (Yakowitz and Spragins, 1968). While mixtures of discrete distributions need not be identifiable, finite mixtures

of Poisson distributions (Teicher, 1960) and Negative Binomial distributions (Yakowitz and Spragins, 1968) are identifiable. See Titterington *et al.* (1985, §3.1) for a detailed account of the identifiability of finite mixture models.

When the number of components k is large, some of the mixing weights can become so close to 0 that the mixture models are close to non-identifiable; or if two components are very close to each other, a mixture density with k components can be empirically indistinguishable from a mixture with fewer than k components.

1.3 Estimation in Mixture Models

Various methods have been developed for estimating the parameters in finite mixture models. We mention four of them that are widely used in practice and cited in the literature: method of moments, minimum distance method, maximum likelihood method and Bayesian method.

Dating back to the work of Pearson (1984), the method of moments is one of the earliest methods for estimating the parameters in finite mixture models. It was widely used in applications when computers were not fast enough to find the maximum of the log-likelihood function. Some developments of moment estimators can be found in Lindsay and Basak (1993), Furman and Lindsay (1994a, 1994b), Lindsay (1995), Withers (1996), as well as Craigmile and Titterington (1998). Even today, they are

still useful serving as initial values for iterative numerical methods to compute the maximum likelihood estimates (Lindsay, 1995).

Minimum distance estimation, introduced by Wolfowitz (1957), is another general method for estimating $\boldsymbol{\Psi}$ in a finite mixture. It minimizes the distance between the empirical distribution and the mixture distribution, or between the kernel density and the mixture density. Titterington *et al.* (1985, §4.5) gave a detailed review of the minimum distance estimators. Note that the maximum likelihood estimator (MLE) can be viewed as a special case of minimum distance estimators, since it minimizes the Kullback-Leibler (1951) distance between the empirical distribution and the mixture distribution.

Since finding numerical solutions of a likelihood equation became feasible, likelihood-based inference has enjoyed fast development and played an important role in the scope of finite mixture models. Let the data take the form of a random sample of observations $X_1 = x_1, \ldots, X_n = x_n$, where the distributions of each X is described by a parametric finite mixture density of the form (1.4). The likelihood function and log-likelihood function of $\boldsymbol{\Psi}$ are given by

$$L_n(\boldsymbol{\Psi}) = \prod_{i=1}^{n} [\sum_{j=1}^{k} w_j f(x_i|\boldsymbol{\theta}_j)] \tag{1.5}$$

and

$$l_n(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \log\Big\{ \sum_{j=1}^{k} w_j f(x_i|\boldsymbol{\theta}_j)\Big\}. \tag{1.6}$$

The maximum likelihood estimator of $\boldsymbol{\Psi}$ is defined to be

$$\hat{\boldsymbol{\Psi}} = \arg\max_{\boldsymbol{\Psi}\in\Omega} l_n(\boldsymbol{\Psi}).$$

when this exists. Because the explicit expression for the MLE's are typically not available, a number of numerical algorithms have been developed for maximizing the log-likelihood function. Among them, the expectation-maximization (EM) algorithm (Dempster *et al*, 1977) is one of the most popular methods. More details can be found in McLachlan and Krishnan (1997), as well as McLachlan and Peel (2000). An `R` package `mixdist` by Macdonald and Du (2010) deserves mentioning here. It contains functions for fitting finite mixture models to grouped data and conditional data by the method of maximum likelihood using a combination of a Newton-type algorithm and the EM algorithm; the components can be Normal, Lognormal, Gamma, Exponential, Weibull, Binomial, Negative Binomial or Poisson distributions.

Under some classes of mixture models, ordinary MLE's are inconsistent or not well defined. For example, the MLE is not well defined in the two-component Normal mixture (1.3), because $l_n(\boldsymbol{\Psi}) \to \infty$ when $\mu_1 = X_1$, $\sigma_1^2 \to 0$ and the other parameters are

held fixed. To account for this case, Hathaway (1985) and Tan *et al.* (2006) discussed the use of constrained MLE, and Chen *et al.* (2008) investigated the properties of penalized MLE.

The fourth method for estimating $\boldsymbol{\Psi}$ is the Bayesian method. Let $L_n(X_1, \ldots, X_n|\boldsymbol{\Psi})$ be the likelihood function of $\boldsymbol{\Psi}$. Assuming that a prior distribution $p(\boldsymbol{\Psi})$ on $\boldsymbol{\Psi}$ is available, then the posterior density $p(\boldsymbol{\Psi}|X_1, \ldots, X_n)$ can be obtained by

$$p(\boldsymbol{\Psi}|X_1, \ldots, X_n) \propto L_n(X_1, \ldots, X_n|\boldsymbol{\Psi})p(\boldsymbol{\Psi}),$$

using Bayes' theorem.

There are various reasons why people may be interested in using Bayesian methods in finite mixture models (Frühwirth-Schnatter, 2006, §2.4.5). First, introducing a suitable prior distribution for $\boldsymbol{\Psi}$ can avoid spurious modes when maximizing the log-likelihood function. Secondly, when the posterior distribution for the unknown parameters is available, Bayesian methods provide valid inference without relying on the asymptotic normality. This is an advantage especially when the sample size n is small, because the asymptotic theory of the MLE can apply only when n is very large.

What can we do if we want to compare different models for a particular set of data? In frequentist statistics, three methods for model comparison are popular: the likelihood ratio test (LRT), Akaike's information criterion (AIC) and the Bayesian

information criterion (BIC). The LRT is used to compare the fit of two models one of which is nested within the other. Both AIC and BIC are based on the log-likelihood evaluated at the MLE and penalized for the number of parameters in the model. There is a danger of over-parameterization, so only those parameters that substantially improve the model should be included. Spiegelhalter *et al.* (2002) reviewed a number of Bayesian approaches to tackling the question of model fit and model comparison including Bayes Factors, the deviance information criterion (DIC) and the penalized expected deviance (PED). In addition, Reversible Jump MCMC (RJMCMC) method (Green, 1995) enables us to get a handle on both model selection and parameter estimation in one single algorithm.

1.4 Data Sets

1.4.1 The Galaxy Data

The galaxy data, first given by Roeder (1990), consist of 82 velocities of distant galaxies diverging from our own, sampled from 6 well-separated conic sections of the *corona borealis*. It was believed that galaxies further from us are moving at greater velocities, due to the expansion of the Universe. Thus, distance is proportional to and can be estimated from velocity. This data set has been analyzed under a variety of mixture models by many researchers, including Crawford (1994), Chib (1995), Carlin and Chib

(1995), Escobar and West (1995), Phillips and Smith (1996), Richardson and Green (1997), and Stephens (2000a).

1.4.2 The Fishery Data

The fishery data, given in Cassie (1954) and Titterington *et al.* (1985, §2.1), consist of lengths of 256 snappers. The underlying categories are the possible age groups to which an individual fish may belong. Thus the component densities describe the length distributions for fish of different ages and the mixing weights indicate the age distribution of snappers in the total population. For a given age group, the length of a fish is assumed to follow a Normal distribution.

1.4.3 The Acidity Data

The acidity data concern an acidity index measured in a sample of 155 lakes in north-central Wisconsin. This index describes the capability of a lake to absorb acid; low values can lead to a loss of biological resources. It was believed that seepage lakes, which have neither inlets or outlets, tend to have lower acidity index, and drainage lakes with both inlets and outlets tend to have higher values. This data set has been analyzed as a mixture of Normal distributions on the log-scale by Crawford *et al.* (1992, 1994), as well as Richard and Green (1997) using reversible jump algorithms.

1.5 Purpose and Motivation

This thesis is a review and discussion of recent published work in Bayesian modeling and inference for finite mixture distributions and an attempt to find methods that will succeed in applications of practical interest.

So far, Bayesian methods that have been proposed and developed for mixture models are not simple to implement. We bring together several approaches for mixtures with both known and unknown number of components to see if any of them could work in problems of practical importance. Although complicated methods have been proposed for issues such as label switching, few of them have been found to be both natural and effective. We also review some of the computer software that is available.

The rest of my thesis is organized as follows. Chapter 2 is a review of Bayesian framework for mixture distributions; we discuss issues on choices of priors, non-identifiability and convergence monitoring. Chapter 3 concerns mixtures with a known number of components; we illustrate the implementation and applications of Gibbs sampling to mixtures of Normal distributions. In Chapter 4 we consider the case when the number of components is unknown; a fully Bayesian analysis to the acidity data is presented. Finally, we offer some concluding remarks and some directions for further research in Chapter 5.

Chapter 2

Bayesian Framework for Mixture Distributions

2.1 Completion

Like the EM algorithm, practical Bayesian estimation using MCMC methods is based on the work of Dempster *et al.* (1977) who pointed out that a finite mixture model can always be expressed as an incomplete-data problem by introducing the allocations as missing data. That is, every observation x_i can be associated with a latent variable $z_i \in \{1, \ldots, k\}$ that indicates the allocation of x_i. The corresponding completion of

the mixture model is then

$$z_i|\boldsymbol{w} \sim M_k(w_1, \ldots, w_k), \quad x_i|z_i, \boldsymbol{\theta} \sim f(\cdot|\boldsymbol{\theta}_{z_i}),$$

where $M_k(w_1, \ldots, w_k)$ denotes a multinomial distribution with parameters k and $\boldsymbol{w} = (w_1, \ldots, w_k)$. Therefore, the density of the complete data $y_i = (x_i, z_i)$ is

$$\prod_{j=1}^{k} w_j^{z_{ij}} f_j^{z_{ij}}(x_i|\boldsymbol{\theta}_j),$$

and the likelihood function (1.5) becomes

$$L_n(\boldsymbol{\theta}, \boldsymbol{w}|\boldsymbol{x}, \boldsymbol{z}) = \prod_{i}^{n} w_{z_i} f(x_i|\boldsymbol{\theta}_{z_i}),$$

and

$$\pi(\boldsymbol{\theta}, \boldsymbol{w}|\boldsymbol{x}, \boldsymbol{z}) \propto \Big[\prod_{i=1}^{n} w_{z_i} f(x_i|\boldsymbol{\theta}_{z_i})\Big] \pi(\boldsymbol{\theta}, \boldsymbol{w}),$$

where $\boldsymbol{z} = (z_1, \ldots, z_n)$.

One may wonder why such completion is useful in this setting since the observed marginal likelihood can be computed in closed form. The reason is that using latent indicator variables usually leads to an efficient simulation algorithm that quickly focuses on the modes of the posterior distribution. Diebolt and Robert (1994) first constructed MCMC inference for this model with fixed number of components k. If the

prior distribution of $\boldsymbol{w}$ is Dirichlet, then its posterior distribution given $\boldsymbol{z} = (z_1, ..., z_n)$ is also Dirichlet; the posterior conditional distribution of $\boldsymbol{\theta}_j$ given $\boldsymbol{w}$ is simply obtained by combining its prior distribution with the sample values x_i for which $z_{i=j}$. Generalization to unknown k was given by Richardson and Green (1997), using the Reversible Jump MCMC algorithm. For a different Bayesian approach to computation for mixture models, see Fraley and Raftery (2002).

2.2 Choices of Priors

2.2.1 Objective and Subjective Priors

Objective priors should be used when we have no prior information about the parameters in the model. However, there exist no general rules about how this ignorance should be expressed in terms of a probability distribution. Therefore, improper priors which are not integrable over the parameter space are often used, in the hope that the data themselves are informative enough to turn the improper prior into a proper posterior distribution. The choice of objective priors is particularly difficult for finite mixture models, since some commonly-used improper priors can lead to improper posteriors.

Subjective priors bring prior knowledge into the analysis and offer the advantage of being proper. They are usually obtained by choosing priors that are conjugate for

the complete-data likelihood function. It is common to assume that the parameters $(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k)$ are independent of the weight distribution $\boldsymbol{w}$. The standard prior for the weight distribution $\boldsymbol{w}$ is the Dirichlet distribution, and the priors on the component parameters $(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k)$ depend on the distribution family underlying the mixture distribution.

The parameters of a subjective prior are called hyper-parameters. Results from Bayesian analyses of finite mixture models using subjective prior information often highly depends on particular choices of hyper-parameters. However, it is not always easy to assess these hyper-parameters. To reduce their sensitivity, hierarchical priors are often used, where the hyper-parameter is equipped with a prior of its own. In any case, in a Bayesian analysis of finite mixture models, the prior distribution has to be selected carefully.

2.2.2 Proper and Improper Priors

The possibility of getting few or no observations from a given component in the sample has a direct drawback. It prohibits the use of independent priors

$$\pi(\boldsymbol{\theta}) = \prod_{j=1}^{k} \pi(\boldsymbol{\theta}_j),$$

since if

$$\int \pi(\boldsymbol{\theta}_j) d\boldsymbol{\theta}_j = \infty$$

then for every sample size n and any sample $\boldsymbol{x}$,

$$\int \pi(\boldsymbol{\theta}, \boldsymbol{w}|\boldsymbol{x}) d\boldsymbol{\theta} d\boldsymbol{w} = \infty.$$

Note that using improper priors in mixture models is not impossible, by adding some degree of dependence between the component parameters, as demonstrated in Mengersen and Robert (1996). Alternatively, Marin *et al.* (2005) proposed to introduce first a common reference parameter (μ, τ), and to define the original parameters in terms of departure from those references. For other approaches to the use of default or non-informative priors in the setting of mixture models, see Wasserman (2000), Pérez and Berger (2002), as well as Moreno and Liseo (2003).

2.3 Nonidentifiability

In Bayesian mixture models, parameter estimation is not always straightforward. The common practice of estimating parameters by their posterior mean and summarizing joint posterior distributions by marginal distributions often leads to nonsensical answers. This is due to the so-called "label-switching" problem, caused by symmetry in the likelihood of the model parameters. For a k-component mixture, the parameter

space has $k!$ regions over which the likelihood is identical, that is, the component parameters are not marginally identifiable. Thus, if $(\theta_1, \ldots, \theta_k)$ is a local maximum, so is $(\boldsymbol{\theta}_{\sigma_1}, \ldots, \boldsymbol{\theta}_{\sigma_k})$ for every permutation $\sigma \in \Sigma_n$. This makes maximization and exploration of the posterior surface difficult. Moreover, if an exchangeable prior is used on $\boldsymbol{\theta} = (\boldsymbol{\theta}_{\sigma_1}, \ldots, \boldsymbol{\theta}_{\sigma_k})$, all the marginals on $\boldsymbol{\theta}$ will be identical.

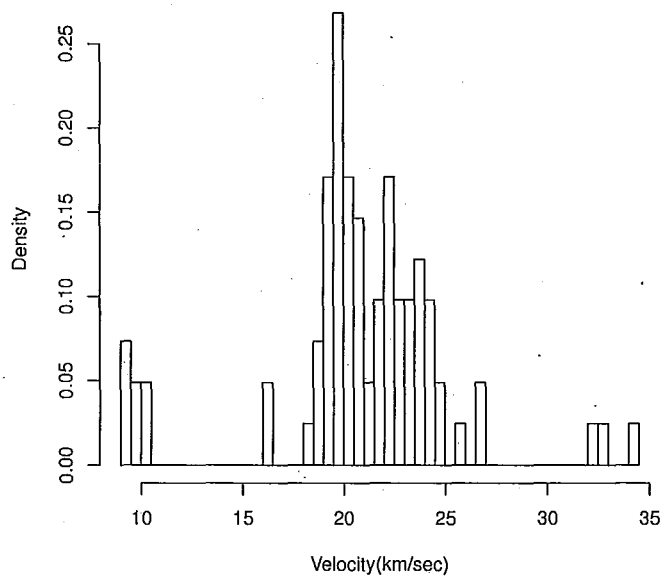

Figure 2.1: Histogram of the galaxy data.

We illustrate this label switching problem using the galaxy data introduced in Chapter 1. A histogram of the 82 velocities is shown in Figure 2.1. We model the data

as independent observations from a mixture of $k = 6$ univariate Normal distributions:

$$\sum_{j=1}^{6} w_j N(x|\mu_j, \sigma_j^2), \tag{2.1}$$

where $N(\cdot|\mu, \sigma^2)$ denotes the density function of a Normal distribution with mean μ and variance σ^2.

We fitted model (2.1) using Gibbs sampling, with non-informative priors proposed by Raftery (1996b). Figure 2.2 shows the effects of label switching in the sampled values of the component means. As the MCMC scheme moves between relatively well-separated regions of parameter space, we can see distinct jumps in the traces of the means. Intuitively, these regions correspond to some of the 6! ways of labeling the mixture components.

Figure 2.3 shows that the estimates of the marginal posterior distributions of the means are very similar to each other, therefore the estimation of the means based on the MCMC output will not be straightforward. For the mixing weights and variances, the traces and estimates of their marginal posterior distributions behave in the same way (not shown here).

If the galaxies are clumped, the distribution of their velocities will be multimodal, each mode representing a cluster that moves away at its own speed. Due to label switching, inference for the individual cluster is no longer meaningful.

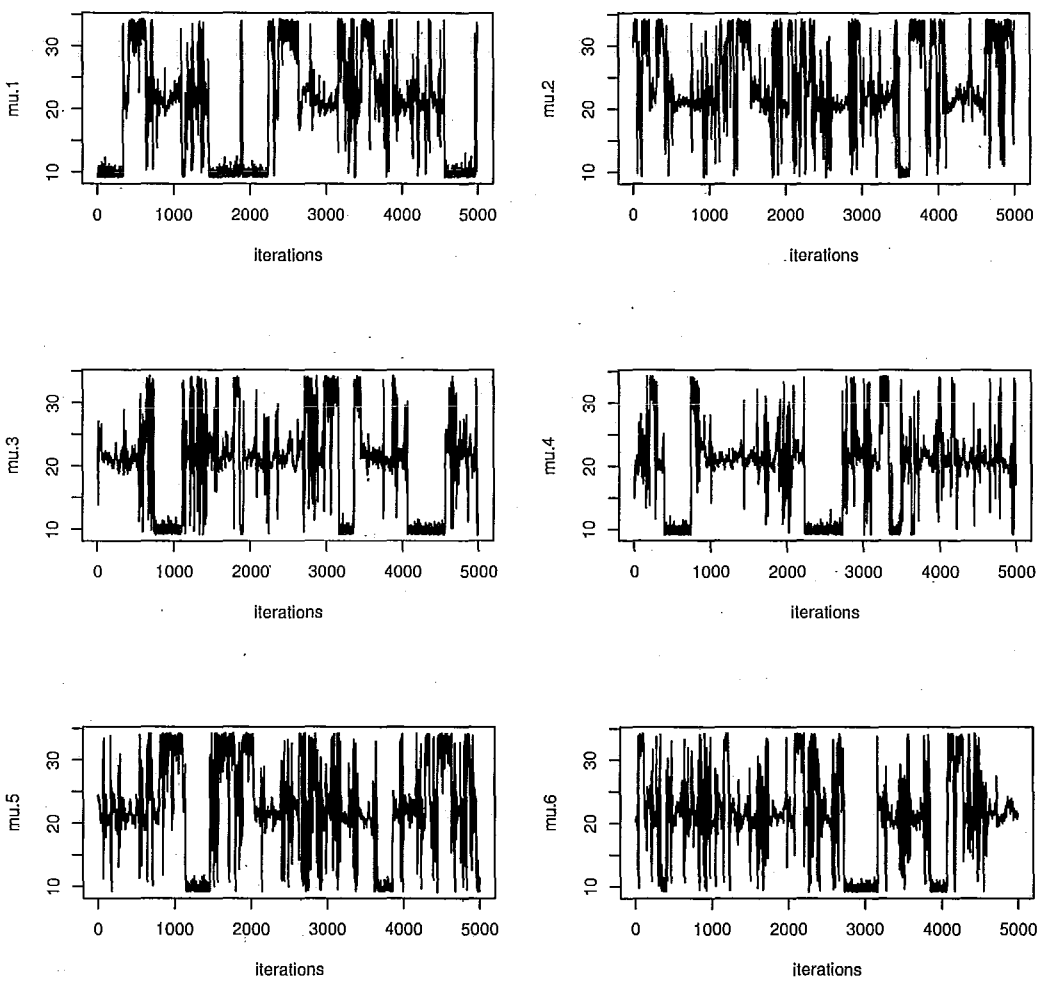

Figure 2.2: Traces of component means to illustrate the effects of label switching in the raw output of the Gibbs sampler when fitting a mixture of Normal distributions to the galaxy data.

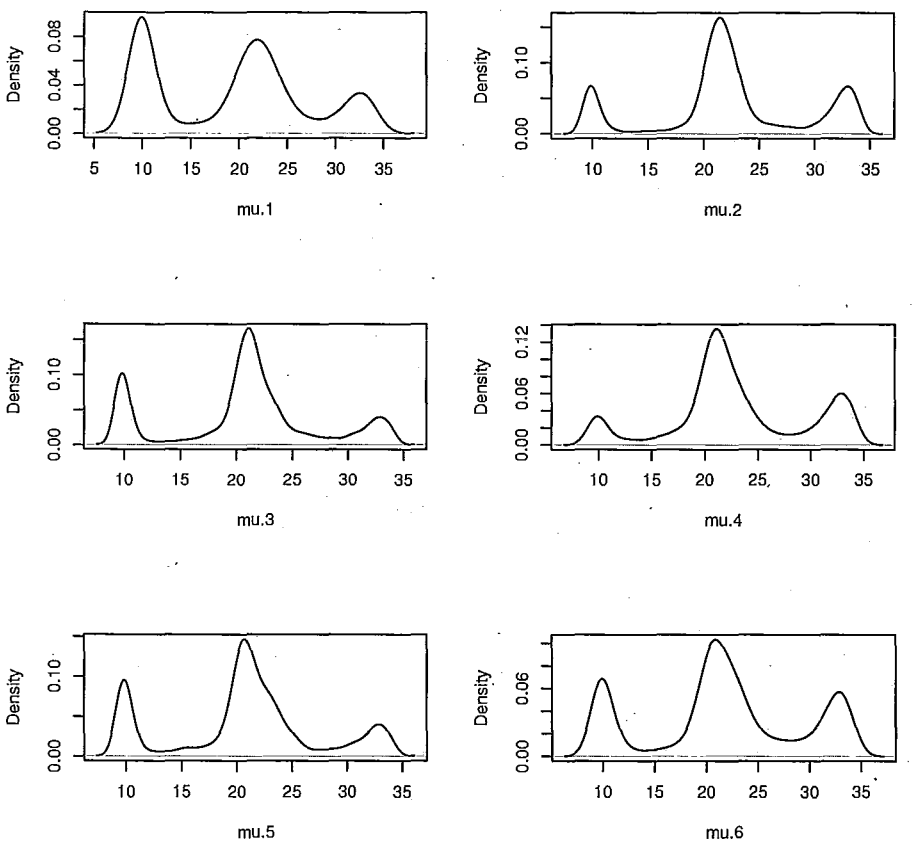

Figure 2.3: Estimated marginal posterior densities of component means to illustrate the effects of label switching in the raw output of the Gibbs sampler when fitting a mixture of Normal distributions to the galaxy data.

A frequent response to this problem is to remove the symmetry by imposing an artificial identifiability constraint on the parameters. For instance, we could arrange the mixture components in order of non-decreasing means $\mu_1 \leq \mu_2 \leq \cdots \leq \mu_k$ and increasing variances $\sigma_1 < \sigma_2 < \cdots < \sigma_k$ when some means are equal. However, imposing a constraint on one and only one of the different types of parameters (weights, locations, scales) may fail to discriminate between some components of the mixture. We illustrate this in Table 2.1 and Figure 2.4. While one of the estimations is close to the true density, the other two both miss at least one of the three modes.

Table 2.1: Parameter estimates obtained from simulated sample of 500 observations from a three-component Normal mixture, by re-ordering according to one of three constraints, $w : w_1 < w_2 < w_3, \mu : \mu_1 < \mu_2 < \mu_3$, or $\sigma : \sigma_1 < \sigma_2 < \sigma_3$.

Order	p_1	p_2	p_3	μ_1	μ_2	μ_3	σ_1	σ_2	σ_3
True	0.50	0.30	0.20	-1.50	0	1.50	0.5	0.30	0.20
On p	0.22	0.33	0.45	0.01	-1.49	1.48	0.31	0.50	0.26
On μ	0.44	0.33	0.23	-1.50	0.00	1.45	0.49	0.34	0.27
On σ	0.15	0.20	0.65	-1.62	1.50	-0.59	0.23	0.29	0.91

Marin *et al.* (2005) pointed out that the introduction of an identifiability constraint has severe consequences on the resulting inference. When reducing the parameter space to its constrained part, the imposed truncation has no reason to respect the topology of either the prior or the likelihood. The constrained parameter space may include parts of several modes and the resulting posterior mean may lie in a very low probability region, while the high posterior probability areas are located at the boundaries of this space. In addition, the constraint may radically modify the prior modeling and

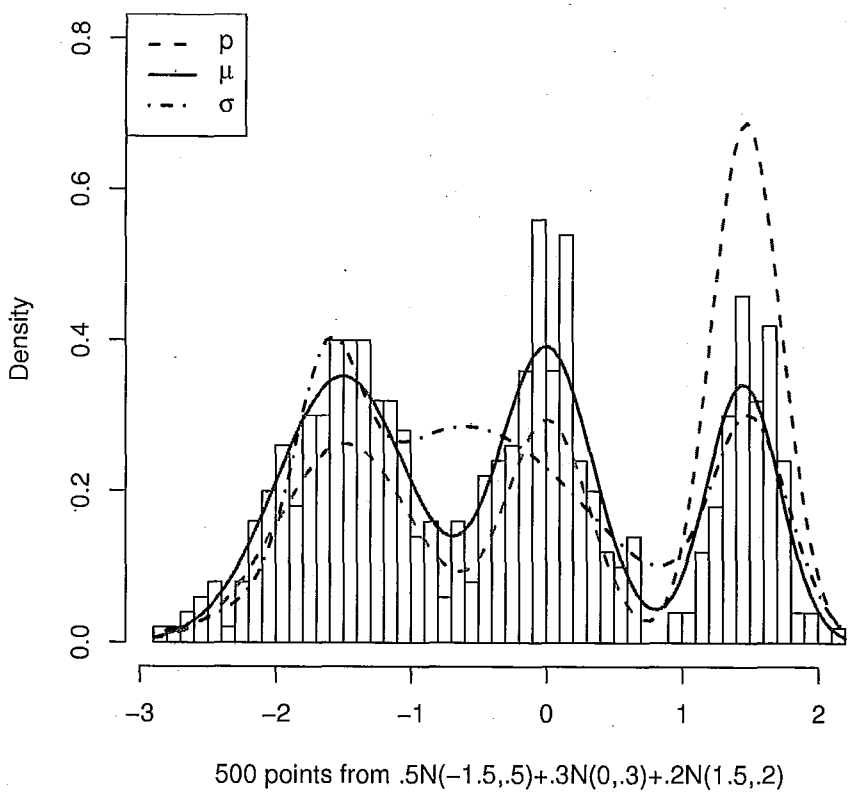

Figure 2.4: Comparison of the plug-in densities for the estimations of Table 2.1 superimposed on the histogram of the data.

contradict the prior information.

For other approaches to this problem, see Roeder and Wasserman (1997b), Celeux *et al.*(2000), Stephens (2000b), Jasra *et al.* (2005), Marin *et al.* (2005), Yao and Lindsay (2009), and Sperrin *et al.* (2010). Note that label switching is only a problem when we wish to make inference about individual components of the mixture. When the mixture is being used as a nonparametric representation, the focus is on inference about g, and label switching is then irrelevant.

2.4 Issues on Convergence

2.4.1 The Burning-in Period

For an MCMC sampler, the number of steps until the chain approaches stationarity is called the length of the burn-in period. Typically the first 1000 to 5000 elements can be discarded. Since a poor choice of starting values or proposal distribution can greatly increase the required burn-in period, an area of current research is whether optimal starting points and proposal distribution can be found. One suggestion is to start the chain as close to the center of the distribution as possible.

A chain is called poorly mixing if it stays in small regions of the parameter space for long periods of time, while a well mixing chain happily explores the space. If the target distribution is multimodal and our choice of starting values traps us near one

of the modes, then it is easy to obtain a poorly mixing chain. Two approaches have been suggested for situations where the target distribution may have multiple peaks: one is to use multiple highly dispersed initial values to start several chains (Gelman and Rubin 1992), and the other is to use a simulated annealing algorithm (Metropolis *et al.*, 1953).

2.4.2 Convergence Diagnostics

In the previous sections, we have assumed that under general conditions the MCMC algorithms are convergent, because the chains they produce are ergodic (Tierney, 1994). We wish that there existed some clear convergence markers so that no sequential processing would be needed. Unfortunately this is almost impossible in practice. It is illusional to think that we can assess the convergence behavior of a Markov chain based on a few thousand realizations of it. No theories tell us when to stop and make estimations with enough confidence. In this subsection, we describe several techniques that are mostly empirical to assess the convergence behavior.

The minimal requirement for the convergence of an MCMC algorithm is that the distribution of the chain $(x^{(t)})$ should be the stationary target g. To detect non-stationarity, a first approach is to plot the evolution of the output from the simulated chains, componentwise and jointly. This is not as straightforward as it seems. For a given time t, it is difficult for $(x^{(t)})$ to be exactly distributed from g when g is a complex

distribution. Moreover, it is theoretically impossible if we only consider a single realization of $(x^{(t)})$. Another useful plot is to draw the empirical cumulative distribution functions (cdf) derived from the Markov chains and check for their stability.

What if we want to access convergence in a more formal way? One can use the Geweke test (Geweke, 1992), which splits the sample into two parts, the first 10% and last 50%, after removing the burn-in's. If the chain has reached stationarity, the means of the two samples should be equal. A modified z-test can be used to compare the two samples, and the resulting test statistic is often referred to as the Geweke z-score. A value larger than 2 indicates that the mean of the series is still drifting, and a longer burn-in is required. Additional diagnostic checks for stationarity are discussed by Geyer (1992), Gelman and Rubin (1992), as well as Raftery and Lewis (1992b).

2.4.3 The coda Package

An R package coda written by Plummer *et al.* (2006) contains a number of tools for convergence diagnostics of an MCMC algorithm. The majority of diagnostics are based on the review of Cowles and Carlin (1996), as well as Brooks and Roberts (1998). While coda was primarily intended for processing the output of a BUGS run (Lunn *et al.*, 2000), it can also be used to handle an arbitrary output from users' own MCMC programs. Some of the generic functions in this package are listed in Table 2.2.

Consider an example of simulating samples from a bivariate Normal with zero

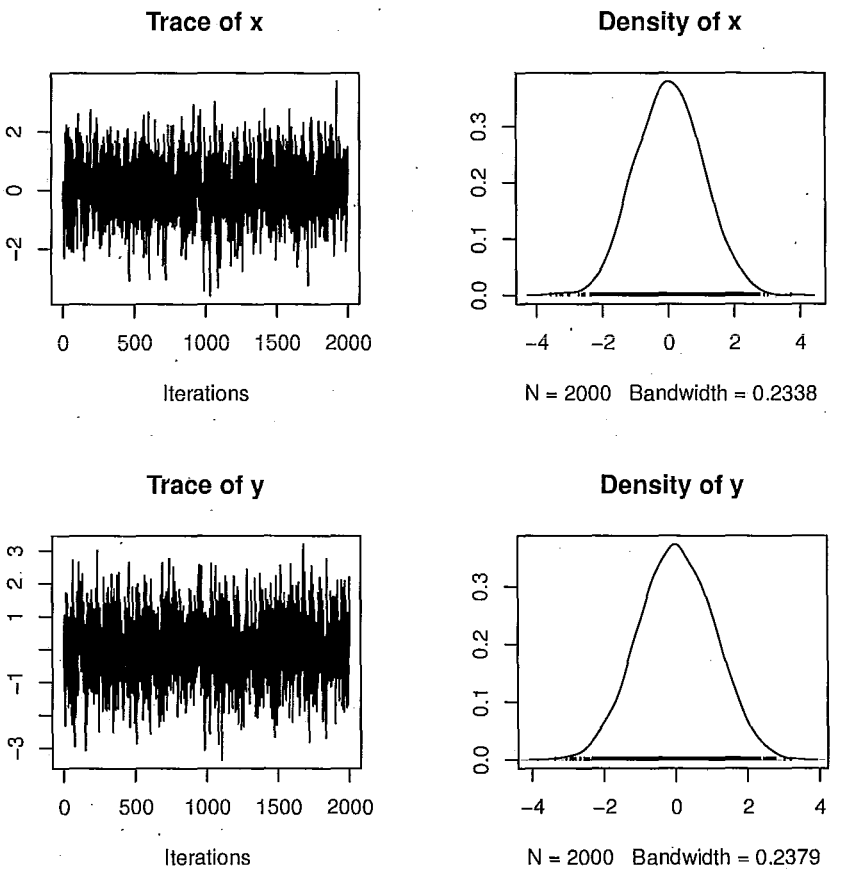

Figure 2.5: Outcome of the `plot.mcmc` function applied to a sample of 2000 values produced by a Gibbs sampler for the bivariate Normal example.

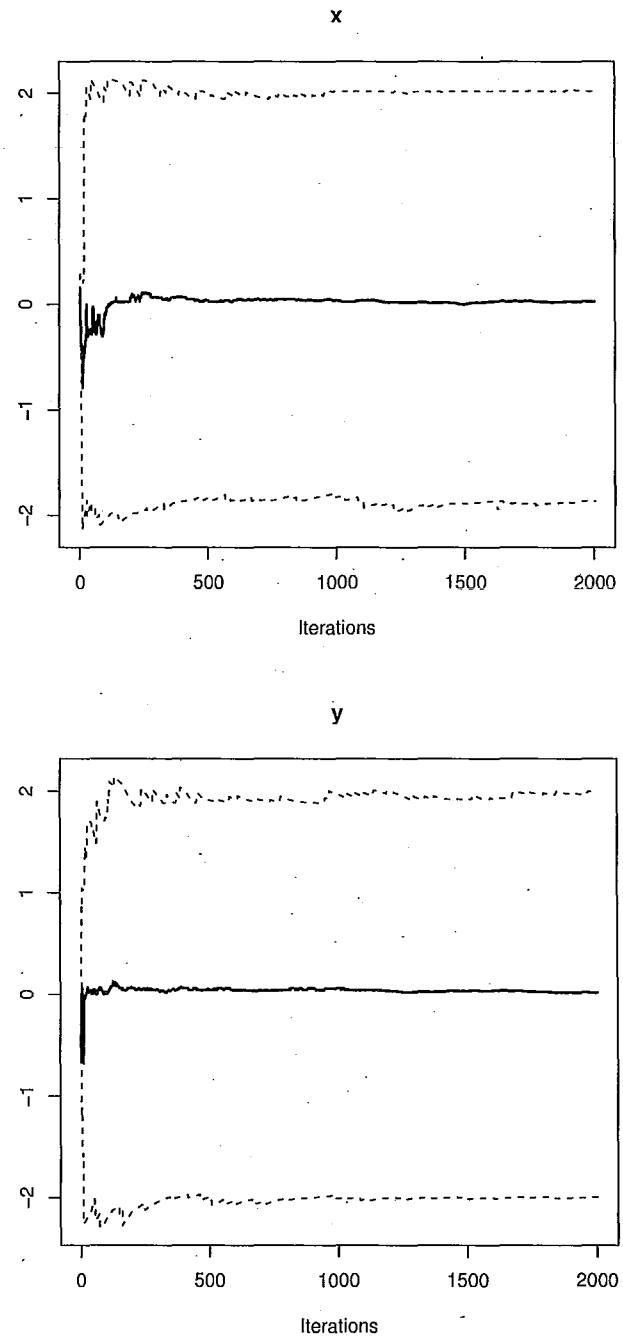

Figure 2.6: Outcome of the `cumuplot` function applied to the same MCMC sample as in Figure 2.5. The lower (upper) plot corresponds to the 2.5% (97.5%) quantile, the central plot to the median.

```
Fraction in 2nd window = 0.5

     x      y
0.5463 0.2600
```

For multiple chains with possibly different initial values, we can use the convergence tool of Gelman and Rubin (1992), implemented as `gelman.diag` and `gelman.plot`. Its stopping rule is based on the difference between a weighted estimator of the variance and the variance of estimators from the different chains. Figure 2.7 describes the evolution of the criterion (called shrink factor) for two parallel chains. It shows a clear stabilization around the target value 1 as early as 1200 iterations. The conclusion is thus in agreement with the above Geweke test.

Ideally, the approximation to g provided by MCMC algorithms should extend to the approximate production of iid samples from g. Note that even within a stationary regime, there exists a difference between the number of iterations and the size of an iid sample from g that would lead to the same variability. Using the same example, the following result returned by `autocorr.diag` shows that at least one out of 5 points should be considered for an iid sample based on this output.

```
> autocorr.diag(mcmc2)
                  x            y
Lag 0   1.00000000  1.000000000
Lag 1   0.24408941  0.252013506
Lag 5  -0.00838563  0.023634908
Lag 10 -0.03563237 -0.048437273
Lag 50  0.02242023 -0.002957836
```

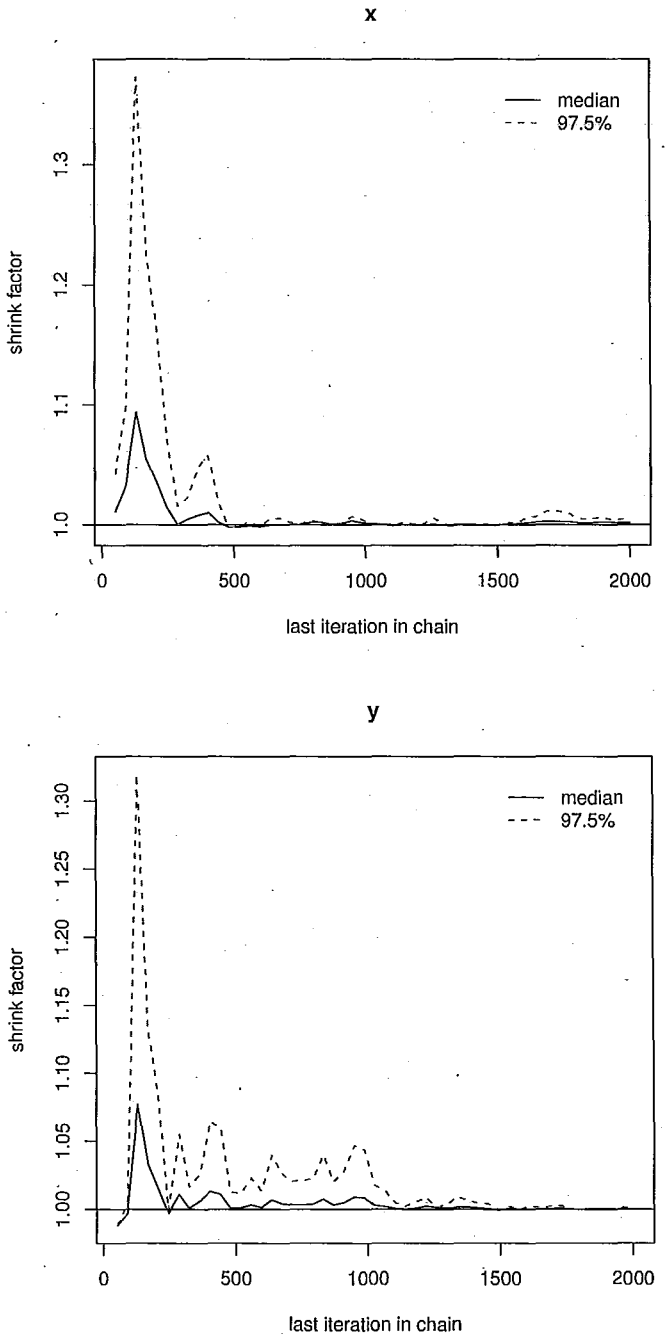

Figure 2.7: Outcome of the `gelman.plot` function applied to the same MCMC sample as in Figure 2.5, using two chains.

The effective sample size, available as `effectiveSize`, provides an indication of the loss in efficiency due to the use of a Markov chain instead of an iid sample. The outcome below leads to an effective sample size of approximately 1215 for x and 1195 for y, which again agrees with the message from Figure 2.7.

```
> effectiveSize(mcmc2)
       x        y
1214.595 1194.256
```

Another useful function is `summary.mcmc`. It produces summary statistics for each variable; these statistics are the mean, standard deviation, naive standard error of the mean, time-series standard error, and quantiles of the sample distribution. More functions and their usage can be found in the latest `coda` manual on CRAN.

```
> summary(mcmc2)

Iterations = 1:2000
Thinning interval = 1
Number of chains = 1
Sample size per chain = 2000

1. Empirical mean and standard deviation for each variable,
   plus standard error of the mean:

     Mean    SD Naive SE Time-series SE
x 0.02979 1.009  0.02255        0.02847
y 0.02547 1.026  0.02295        0.03010

2. Quantiles for each variable:

    2.5%     25%     50%    75% 97.5%
x -1.867 -0.6611 0.02595 0.7180 2.019
y -1.995 -0.6838 0.01638 0.7485 2.004
```

2.4.4 One Long Chain or Many Shorter Chains?

One can either use a single long chain (Geyer 1992, Raftery and Lewis 1992b) or multiple chains each starting from different initial values (Gelman and Rubin 1992). Note that with parallel processing machines, using multiple chains may be computationally more efficient than a single long chain. However, Geyer (1992) argues that using a single longer chain is the better approach. If long burn-in periods are required, or if the chains are highly autocorrelated, using a number of smaller chains may result in each not being long enough to be useful.

Chapter 3

Bayesian Modeling and Inference for Mixture Distributions with Known Number of Components

3.1 Gibbs Sampling

3.1.1 Introduction

Gibbs sampling or the Gibbs sampler, named after the physicist J. W. Gibbs, is an algorithm that generates a sequence of samples from the joint probability distribution of two or more random variables. It was first described in a statistical paper by

brothers Stuart and Donald Geman (1984), whose work built on that of Metropolis *et al.* (1953), Hastings (1970) and Peskun (1973). Influenced by Geman and Geman (1984), Gelfand and Smith (1990) wrote a paper that had an enormous impact on Bayesian methods, statistical computing, and stochastic processes. Since then, the Gibbs sampler has become one of the most popular methods for summarizing complex posterior distributions. Although Tanner and Wong (1987) and Besag and Clifford (1989) had proposed similar solutions, they did not receive the same response from the statistical community.

Let us first consider a bivariate random variable $(x; y)$, and suppose we wish to compute one or both marginals, $p(x)$ and $p(y)$. The idea behind the Gibbs sampler is that it is much easier to consider a sequence of conditional distributions, $p(x_j|y)$ and $p(y_j|x)$, than it is to obtain the marginal by integration of the joint density $p(x; y)$. The sampler starts with an initial value y_0 for y and obtains x_0 by generating a random variable from the conditional distribution $p(x_j|y = y_0)$. It then generate y_1 based on the value of x_0, drawing from the conditional distribution $p(y_j|x = x_0)$.

Repeating this

$$x_i \sim p(x|y = y_{i-1}),$$

$$y_i \sim p(y|x = x_i)$$

k times, we obtain a Gibbs sequence of length k. A subset of this sequence $(x_j; y_j)$, $1 \leq$

$j \leq m < k$, can be taken as our simulated draws from the full joint distribution $p(x; y)$. After a sufficient burn-in to remove the effects of the initial sampling values, one can sample m points from the chain for estimation and inference purposes. The Gibbs sequence converges to a stationary or equilibrium distribution that is independent of the starting values, and this stationary distribution is the target distribution we are trying to simulate (Tierney 1994).

The implementation of Gibbs sampling is straightforward in this case. We illustrate this using the example mentioned in subsection 2.4.3. Simulating samples from a bivariate Normal with zero mean, unit variance for the marginals and a correlation of $\rho = 0.5$ between the two variables, the core of the R code is

```
> bn_gibbs <- function (n = 500, rho = 0.5, startx = 0, starty = 0)
{
    x <- rep(NA, n)
    y <- x
    x[1] <- startx
    y[1] <- starty
    for (i in 2:n) {
        x[i] <- rnorm(1, y[i - 1] * rho, sqrt(1 - rho^2))
        y[i] <- rnorm(1, x[i] * rho, sqrt(1 - rho^2))
    }
    data.frame(cbind(x, y))
}
```

When more than two variables are involved, the sampler is extended in an obvious fashion. The value of the kth variable is drawn from the distribution $p(\theta^{(k)}|\boldsymbol{\theta}^{(-k)}))$, where $\boldsymbol{\theta}^{(-k)}$ denotes a vector containing all variables but k. Therefore, to obtain the value $\theta_i^{(k)}$ during the ith iteration, we draw from the distribution

$$\theta_i^{(k)} \sim p(\theta^{(k)}|\theta^{(1)} = \theta_i^{(1)}, \ldots, \theta^{(k-1)} = \theta_i^{(k-1)}, \theta^{(k+1)} = \theta_{i-1}^{(k+1)}, \ldots, \theta^{(n)} = \theta_{i-1}^{(n)}.$$

For example, if there are three variables $(x; y; z)$ involved, the sampler becomes

$$x_i \sim p(x|y = y_{i-1}, z = z_{i-1})$$

$$y_i \sim p(y|x = x_i, z = z_{i-1})$$

$$z_i \sim p(z|x = x_i, y = y_i).$$

Not necessarily the most appropriate MCMC method in any given example, the Gibbs sampler and its various adaptations remain the most popular one in applied Bayesian statistics. This is because for many Bayesian models, the implementation of the Gibbs sampler is particularly convenient due to two properties: conditional conjugacy and conditional independence. The conditional conjugacy ensure that the posterior conditional distributions required by the Gibbs sampler are from the same family as the prior conditional distributions. If this family is easy to sample from, then the Gibbs sampler will be straightforward to implement.

Conditional independence arises in hierarchical models. Suppose that the likelihood for data x is $f(x|\boldsymbol{\theta})$, the prior for $\boldsymbol{\theta}$ is $f(\boldsymbol{\theta}|\boldsymbol{\phi})$ and the hyperprior for $\boldsymbol{\phi}$ is $f(\boldsymbol{\phi})$. Then $\boldsymbol{\phi}$ is conditionally independent of x given $\boldsymbol{\theta}$, and the posterior conditional densities are given by $f(\boldsymbol{\theta}|\boldsymbol{\phi}) \propto f(x|\boldsymbol{\theta})f(\boldsymbol{\theta}|\boldsymbol{\phi})$ and $f(\boldsymbol{\phi}|\boldsymbol{\theta}) \propto f(\boldsymbol{\theta}|\boldsymbol{\phi})f(\boldsymbol{\phi})$. Therefore, if each

stage of the hierarchical model permits convenient sampling, the Gibbs sampler will be again straightforward to implement.

3.1.2 General Gibbs Sampling for Mixture Models

Definition 3.1.1. *Given a distribution $f(x)$, a density g that satisfies $\int_z g(x,z)dz = f(x)$ is called a completion of f. When the vector of auxiliary parameters corresponds to latent data that are not directed observed, this is also called data augmentation.*

The data augmentation method proposed by Tanner and Wong (1987) was to notice that it may be simpler and more efficient to sample from a distribution $f(\boldsymbol{\theta}, \boldsymbol{\phi}|x)$ than from $f(\boldsymbol{\theta}|x)$. The augmentation parameter, also called the auxiliary variable $\boldsymbol{\phi}$, can be anything. If we can sample from $f(\boldsymbol{\theta}, \boldsymbol{\phi}|x)$, then the required $f(\boldsymbol{\theta}|x)$ is simply a marginal distribution of the augmented distribution, and a sample from $f(\boldsymbol{\theta}|x)$ consists of ignoring the $\boldsymbol{\phi}$ components of the $(\boldsymbol{\theta}, \boldsymbol{\phi})$ sample.

The Gibbs sampler has been the most commonly used approach in Bayesian mixture estimation (Diebolt and Robert, 1994; Lavine and West, 1992; Verdinelli and Wasserman, 1991; Chib, 1995; Escobar and West, 1995). Its basic feature in the mixture setup is the augmentation of the parameters, by associating each of the allocation variables z_i's with one of the observation x_i's. That is, we introduce a missing multinomial variable $z_j \sim M_k(1|w_1, \ldots, w_k)$ such that $x_i|z_j = i \sim f(x|\theta_i)$. This allows simulation of the parameters of each component conditionally on the allocations,

taking into account only the observations allocated to this component.

Therefore, in a heterogeneous population made of several homogeneous subgroups, it makes sense to interpret z_j, which is missing in the observation, as the index of the population which x_j comes from. In the alternative non-parametric perspective, it is often meaningless to analyze the individual mixture components. However, since the goal of an MCMC sampler is to provide a Markov chain that converges to the posterior distribution, the difference between natural and artificial completion is lost, and completion is merely a tool to generate such a chain.

Let $\pi(\boldsymbol{w}|\boldsymbol{z},\boldsymbol{x})$ denote the density of the distribution of $\boldsymbol{w}$ given $\boldsymbol{z} = (z_1,\ldots,z_n)$ and $\boldsymbol{x}$. Let $\pi(\boldsymbol{\theta}|\boldsymbol{z},\boldsymbol{x})$ be the density of the distribution of $\boldsymbol{\theta}$ given $(\boldsymbol{z},\boldsymbol{x})$. Note that $\pi(\boldsymbol{w}|\boldsymbol{z},\boldsymbol{x})$ is independent of $\boldsymbol{x}$, so $\pi(\boldsymbol{w}|\boldsymbol{z},\boldsymbol{x}) = \pi(\boldsymbol{w}|\boldsymbol{z})$.

We now describe the Gibbs sampler for mixture models given in Diebolt and Robert (1994), based on successive simulation of $\boldsymbol{z}$, $\boldsymbol{w}$, $\boldsymbol{\theta}$ conditional on one another and on the data:

0. Initialization: choose $\boldsymbol{w}^{(0)}$ and $\boldsymbol{\theta}^{(0)}$ arbitrarily.

1. Step t. For $t = 1,\ldots$

1.1 Generate $z_i^{(t)}$ $(i = 1,\ldots,n)$ from $(j = 1,\ldots,k)$

$$\mathbb{P}\left(z_i^{(t)} = j|w_j^{(t-1)},\theta_j^{(t-1)},x_i\right) \propto w_j^{(t-1)} f\left(x_i|\theta_j^{(t-1)}\right).$$

1.2 Generate $\boldsymbol{w}^{(t)}$ from $\pi(\boldsymbol{w}|\boldsymbol{z}^{(t)})$.

1.3 Generate $\boldsymbol{\theta}^{(t)}$ from $\pi(\boldsymbol{\theta}|\boldsymbol{z}^{(t)}, \boldsymbol{x})$.

3.2 Finite Mixture of Normal Distributions

3.2.1 Introduction

The density of a finite mixture of k univariate Normal distributions is given by

$$p(y|\theta) = w_1 N(y|\mu_1, \sigma_1^2) + \ldots + w_k N(y|\mu_k, \sigma_k^2), \tag{3.1}$$

with $N(y|\mu_k, \sigma_k^2)$ being the density of a univariate Normal distribution. Here we are interested in the estimation of the weight distribution $\boldsymbol{w} = (w_1, \ldots, w_k)$, the component means $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_k)$ and the component variances $\boldsymbol{\sigma}^2 = (\sigma_1^2, \ldots, \sigma_k^2)$, based on the data $\boldsymbol{y} = (y_1, \ldots, y_n)$.

Pioneering work on the estimation of Normal mixtures was based on the method of moments (Pearson, 1894). Maximum likelihood estimation was used for a mixture of two univariate Normal distributions with $\sigma_1^2 = \sigma_2^2$ as early as Rao (1948), and Hasselblad (1966). Later, numerical optimization procedures such as the EM algorithm (Dempster *et al.*, 1977) became available too.

A difficulty with the maximum likelihood estimation, first noted by Kiefer and Wolfowitz (1956) for univariate mixtures of Normals, is that the mixture likelihood

function

$$p(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\sigma}^2, \boldsymbol{w}) = \prod_{i=1}^{n} [\sum_{j=1}^{k} w_j \phi(y_i|\mu_j, \sigma_j^2)] \tag{3.2}$$

is unbounded and has many local spurious modes, therefore the resultant likelihood estimator is only a local maximum. We illustrate this using the Kiefer-Wolfowitz example in the next subsection.

3.2.2 The Kiefer-Wolfowitz Example

Consider the following mixture of two Normal distributions

$$Y \sim wN(\mu, 1) + (1 - w)N(\mu, \sigma^2), \tag{3.3}$$

where w is fixed, μ and σ^2 are unknown. This example was used by Kiefer and Wolfowitz (1956) to show that each observation in an arbitrary data set $\boldsymbol{y} = (y_1, \ldots, y_n)$, of arbitrary size n, generates a singularity in the mixture likelihood function (3.2). Whenever $\mu = y_i$, as $\sigma^2 \to 0$, the mixture likelihood $p(\boldsymbol{y}|\mu = y_i, \sigma^2)$ is dominated by a term proportional to a constant times $1/\sigma^2$. Therefore,

$$\lim_{\sigma^2 \to 0} p(\boldsymbol{y}|\mu = y_i, \sigma^2) = \infty.$$

To illustrate this, we simulate $n = 20$ observations from the model (3.3), with $w = 0.5$, $\mu = 0$, and $\sigma^2 = 4$. The sorted observations are:

```
-3.1954 -3.1063 -1.7727 -1.5011 -0.9633 -0.6443 -0.3316 -0.1623
-0.1162  0.2987  0.3706  0.5202  1.1207  1.2242  1.2408  1.5592
 1.6338  1.8173  2.9116  3.6102
```

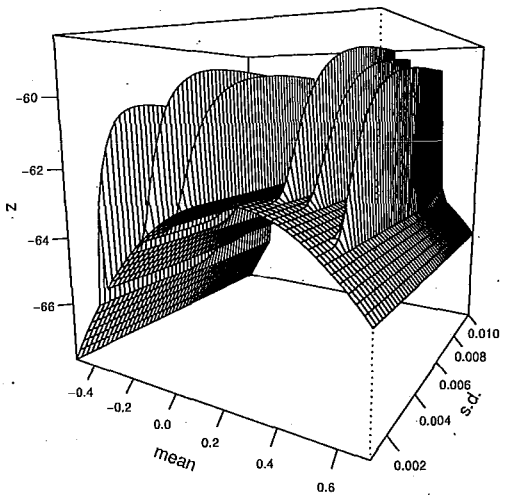

Figure 3.1: The Kiefer-Wolfowitz example - zoom of surface plot for very small values of standard deviation σ.

There is a mode (not shown in graph) around the true value $(\mu, \sigma) = (0, 2)$, but the mixture likelihood is unbounded over a region corresponding to very small values of σ. Figure 3.1 zooms on this part of the parameter space, where we find more than one spurious local mode in the surface plot of $\log p(\boldsymbol{y}|\mu = y_i, \sigma^2)$.

This pathological part of the parameter corresponds to mixtures that fit one component to a small group of similar observations, whereas all other observations are assumed to belong to the second component. Thus, the EM algorithm or other numerical methods for maximizing this likelihood will have a high risk of being trapped at the spurious local modes.

Now assume that the mixture likelihood $p(\boldsymbol{y}|\mu, \sigma^2)$ is combined with the prior $p(\mu, \sigma^2) \propto p(\sigma^2)$, where $\sigma^2 \sim IG(1, 2)$. Figure 3.2 shows the the MCMC draws of μ and σ^2 from the posterior density $p(\mu, \sigma^2|\boldsymbol{y})$ under the prior $\sigma^2 \sim IG(1, 2)$ for the simulated data with $n = 20$ and $n = 200$. The Inverse Gamma prior introduces a constraint and keeps the variance sufficiently away from 0, and all singularities and local modes are cut out. In addition, for the larger data set with $n = 200$, it is less likely to run an MCMC sampler being "trapped"; we will discuss this issue in the next subsection.

3.2.3 Gibbs Sampling for Normal Mean Mixtures

Consider a two-component Normal mixture

$$wN(\mu_1, 1) + (1 - w)N(\mu_2, 1), \tag{3.4}$$

where the weight w is known and not equal to 0.5. In this case, the parameters are identifiable, because μ_1 cannot be confused with μ_2 when w is known and different from 0.5. The log-likelihood surface exhibits two modes: one is close to the true values of the parameters and the other one is a spurious mode but always present, see Figure 3.3. However, if we plot the likelihood, only one mode is visible because of the difference in their magnitudes.

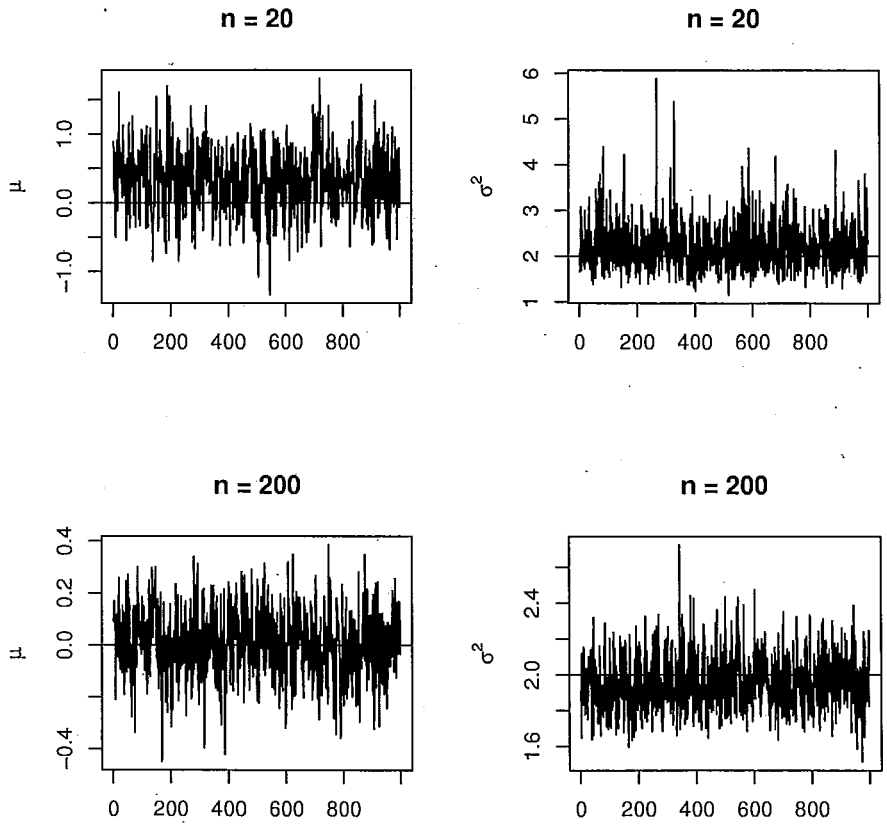

Figure 3.2: The Kiefer-Wolfowitz example - MCMC draws of μ and σ^2 from the posterior $p(\mu, \sigma^2|\boldsymbol{y})$ under the prior $\sigma^2 \sim IG(1, 2)$ based on different sample sizes $n = 20$ (top) and $n = 200$ (bottom); the horizontal line indicates the true values.

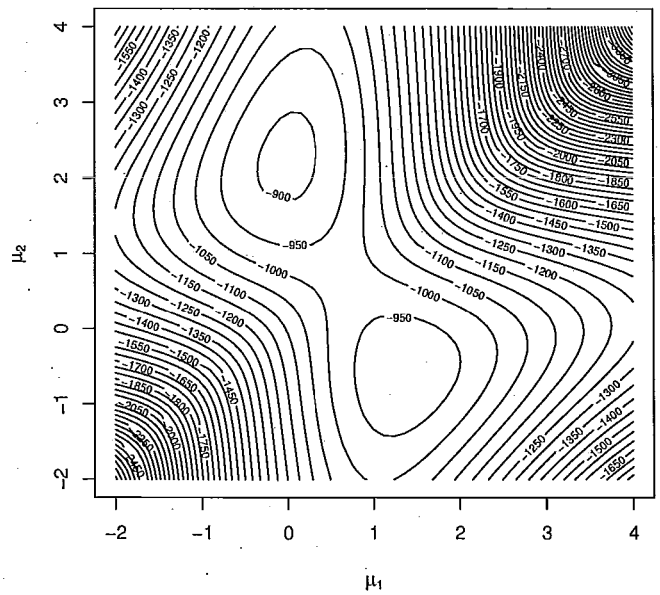

Figure 3.3: Log-posterior surface for the model $0.7N(0,1) + 0.3N(2.5,1)$.

Under a Normal prior $N(\delta, 1/\lambda)$ on both μ_1 and μ_2, with $(s_j^x) = \sum_{i=1}^n \mathbb{I}_{z_i=j} x_i$, it is easy to see that μ_1 and μ_2 are independent, given $(\boldsymbol{z}, \boldsymbol{x})$, with conditional distributions

$$N\Big(\frac{\lambda\delta + (s_j^x)}{\lambda + n_j}, \frac{1}{\lambda + n_j}\Big)$$

for j =1, 2. Similarly, the conditional posterior distribution of $\boldsymbol{z}$ given (μ_1, μ_2) is easily seen to be a product of Bernoulli random variables on {1, 2}, with $(i = 1, \ldots, n)$

$$\mathbb{P}\big(z_i = 1 | \mu_1, x_i\big) \propto w \exp\big(-0.5(x_i - \mu_1)^2\big).$$

The Gibbs sampling for a two-component Normal mean mixture (3.4) is

0. Initialization: choose $\mu_1^{(0)}$ and $\mu_2^{(0)}$.

1. Step t. For $t = 1, \ldots$

1.1 Generate $z_i^{(t)}$ $(i = 1, \ldots, n)$ from

$$\mathbb{P}\big(z_i^{(t)} = 1\big) = 1 - \mathbb{P}\big(z_i^{(t)} = 2\big) \propto w \exp\big(-0.5(x_i - \mu_1^{(t-1)})^2\big),$$

1.2 Compute $n_j^{(t)} = \sum_{i=1}^n \mathbb{I}_{z_i^{(t)}=j}$ and $(s_j^x)^{(t)} = \sum_{i=1}^n \mathbb{I}_{z_i^{(t)}=j} x_i$,

1.3 Generate $\mu_j^{(t)}$ $(j = 1, 2)$ from

$$N\Big(\frac{\lambda\delta + (s_j^x)^{(t)}}{\lambda + n_j^{(t)}}, \frac{1}{\lambda + n_j^{(t)}}\Big).$$

Although this scheme seems straightforward, MCMC algorithms that use Gibbs sampling

through completion do not necessarily enjoy good convergence properties, as shown in Diebolt and Robert (1990b). One of the main defects of the Gibbs sampler is the strong attraction of the local modes, which are usually called trapping (or absorbing) states. While the Gibbs sampler chain $(z^{(t)}, \theta^{(t)})$ is formally irreducible, escaping from these trapping states usually requires an enormous number of iterations.

When only a small number of observations are allocated to a given component, the probability of allocating new observations to this component is very small, and so is the probability of reallocating observations of this component to another component. Components with very small variances can also become so concentrated that there is little probability of moving observations in or out of them.

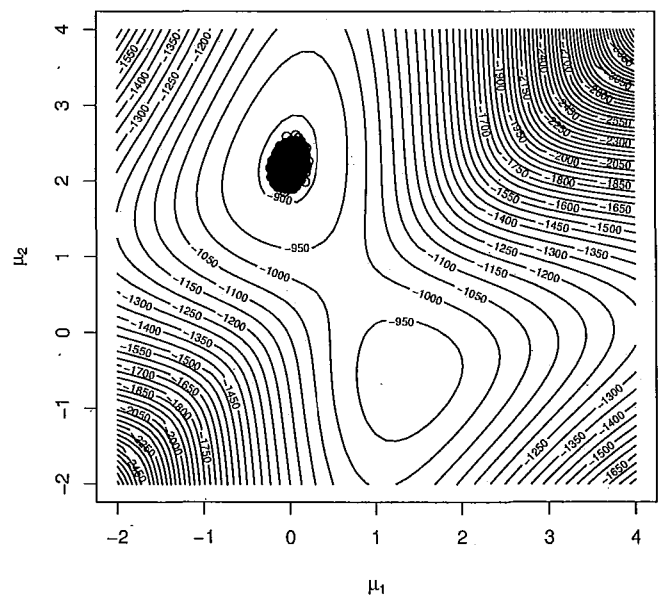

Figure 3.4: Log-postcrior surface and the corresponding Gibbs sample for the model $0.7N(0,1) + 0.3N(2.5,1)$, based on 5000 draws.

Figure 3.4 and 3.5 illustrate the false security of the performance of the Gibbs sampler for a simulated data set of $n = 500$ observations from $0.7N(0,1) + 0.3N(2.5,1)$. Since the Gibbs sampler uses conditional distributions, its moves are restricted in their width. Conditioning on z, the proposals for the means are quite concentrated and do not allow for big jumps in the allocations at the next steps. Even after many iterations, a Gibbs sampler initialized close to the spurious second mode is unable to leave it easily. Unfortunately, there is no way to judge whether the neighborhood of a specific mode has been sufficiently explored, even thought the path of the Markov chain can be exploited to provide an approximation of the marginal posterior distribution of the component parameters.

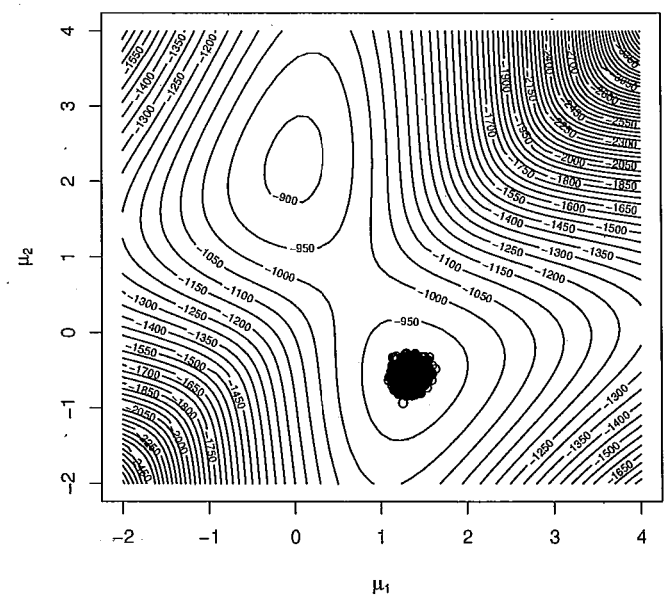

Figure 3.5: Same graph, when initialized close to the second and lower mode, based on 5000 draws.

3.2.4 Gibbs Sampling for Normal Mixtures

Consider a 3-component Normal mixture of the form $\sum_{j=1}^{3} w_j N(\mu_j, \sigma_j^2)$. As in Casella *et al.* (2001) we use conjugate priors

$$\sigma_j^2 \sim IG(\alpha_j, \beta_j),\ \mu_j|\sigma_j^2 \sim N(\lambda_j, \sigma_j^2/\tau_j),\ (w_1, w_2, w_3) \sim D(\gamma_1, \gamma_2, \gamma_3),$$

where IG denotes the Inverse Gamma distribution and λ_j,τ_j,α_j,β_j,γ_j are known hyper-parameters. If we denote

$$(s_j^x)^{(\nu)} = \sum_{i=1}^{n} \mathbb{I}_{z_i=j}(x_i - \mu_j)^2,$$

then

$$[\mu_j|\sigma_j^2, \boldsymbol{x}, \boldsymbol{z}] \sim N\Big(\frac{\lambda_j\delta_j + (s_j^x)^{(x)}}{\tau_j + n_j}, \frac{\sigma_j^2}{\tau_j + n_j}\Big),$$

$$\sigma_j^2|\mu_j, \boldsymbol{x}, \boldsymbol{z} \sim IG\big(\alpha_j + 0.5(n_j + 1), \beta_j + 0.5\tau_j(\mu_j - \lambda_j)^2 + 0.5s_j^{\nu}\big).$$

Raftery (1996b) used the following data-dependent hyper-parameters: $\lambda = \bar{y}$, $\tau = 2.6/(y_{max} - y_{min})^2$, $\alpha = 1.28$, $\beta = 0.36s_y^2$, whereas Bensmail *et al.* (1997) use $\lambda = \bar{y}$, $\tau = 1$, $\alpha = 2.5$, and $\beta = 0.5s_y^2$.

The Gibbs sampler for a 3-component Normal mixture described by Marin *et al.* (2005) is as follows:

0. Initialization: choose $\boldsymbol{w}^{(0)}$ and $\boldsymbol{\theta}^{(0)}$.

1. Step t. For $t = 1, \ldots$

1.1 Generate $z_i^{(t)}$ $(i = 1, \ldots, n)$ from $(j = 1, 2, 3)$

$$\mathbb{P}\big(z_i^{(t)} = j\big) \propto \frac{w_j^{(t-1)}}{\sigma_j^{(t-1)}} \exp\big(-(x_i - \mu_j^{(t-1)})^2/2(\sigma_j^2)_j^{(t-1)}\big),$$

Compute $n_j^{(t)} = \sum_{l=1}^{n} \mathbb{I}_{z_l^{(t)}=j'}(s_j^x)^{(t)} = \sum_{l=1}^{n} \mathbb{I}_{z_l^{(t)}=j} x_l$.

1.2 Generate $\boldsymbol{w}^{(t)}$ from $D(\gamma_1 + n_1, \gamma_2 + n_2, \gamma_3 + n_3)$.

1.3 Generate $\mu_j^{(t)}$ from

$$N\Big(\frac{\lambda_j \delta_j + (s_j^x)^{(t)}}{\tau_j + n_j^{(t)}}, \frac{(\sigma_j^2)^{(t-1)}}{\tau_j + n_j^{(t)}}\Big).$$

Compute $(s_j^\nu)^{(t)} = \sum_{l=1}^{n} \mathbb{I}_{z_l^{(t)}=j}\big(x_l - \mu_j^{(t)}\big)^2$.

1.4 Generate $(\sigma_j^2)^{(t)}(j = 1, 2, 3)$from

$$IG\big(\alpha_j + 0.5(n_j + 1), \beta_j + 0.5\tau_j(\mu_j^{(t)} - \lambda_j)^2 + 0.5(s_j^\nu)^{(t)}\big).$$

To incorporate with any finite number of components, I programmed this in R to estimate the means, variances and weights in a Normal mixture.

```
gibbsnorm <- function (dat, k, niter, alpha = 1.28, beta = 0.36 * var(dat),
    lam = mean(dat), tau = 2.6/(diff(range(dat)))^2, g = 1)
{
    rigamma <- function(n, a, b) {
        return(1/rgamma(n, shape = a, rate = b))
    }
    rdirichlet <- function(n, par) {
        k = length(par)
        z = array(0, dim = c(n, k))
        s = array(0, dim = c(n, 1))
        for (i in 1:k) {
            z[, i] = rgamma(n, shape = par[i])
            s = s + z[, i]
```

```
    }
    for (i in 1:k) {
        z[, i] = z[, i]/s
    }
    return(z)
}
n <- length(dat)
mu <- rnorm(k, mean = mean(dat), sd = sd(dat))
sig <- sd(dat)/k
p <- rep(1/k, k)
mixparam <- list(p = p, mu = mu, sig = sig)
z <- rep(0, k)
nj <- z
sj <- z
sj2 <- z
gibbsmu <- matrix(0, nrow = niter, ncol = k)
gibbssig <- gibbsmu
gibbsp <- gibbsmu
for (i in 1:niter) {
    for (t in 1:n) {
        prob <- mixparam$p * dnorm(dat[t], mean = mixparam$mu,
            sd = mixparam$sig)
        z[t] <- sample(x = 1:k, size = 1, prob = prob)
    }
    for (j in 1:k) {
        nj[j] <- sum(z == j)
        sj[j] <- sum(as.numeric(z == j) * dat)
    }
    repeat {
        gibbsmu[i, ] <- rnorm(k, mean = (lam * tau + sj)/(nj +
            tau), sd = sqrt(mixparam$sig^2/(tau + nj)))
        if (max(gibbsmu[i, ]) < max(dat) & min(gibbsmu[i,
            ]) > min(dat))
            break
    }
    mixparam$mu <- gibbsmu[i, ]
    for (j in 1:k) {
        sj2[j] = sum(as.numeric(z == j) * (dat - mixparam$mu[j])^2)
    }
    gibbssig[i, ] <- sqrt(rigamma(k, alpha + 0.5 * (nj +
        1), beta + 0.5 * tau * (mixparam$mu - lam)^2 + 0.5 *
        sj2))
    mixparam$sig <- gibbssig[i, ]
    gibbsp[i, ] <- rdirichlet(1, par = nj + g)
```

```
        mixparam$p <- gibbsp[i, ]
    }
    data.frame(p = gibbsp, mu = gibbsmu, sigma = gibbssig)
}
```

3.3 Practical Example: The Fishery Data

A histogram of the fishery data is shown in Figure 3.6. We fit a Normal mixture model with $k = 3$ and $k = 4$ respectively, assuming a Dirichlet$(4, \ldots, 4)$ prior for $\boldsymbol{w}$ and using the hierarchical independence priors introduced by Richardson and Green (1997).

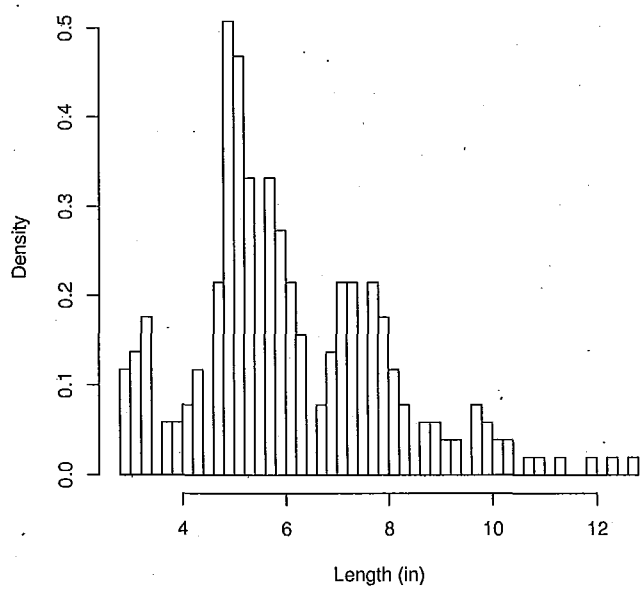

Figure 3.6: Histogram for the fishery data.

We used the Gibbs sampler described in the previous subsection and stored 2000 MCMC

Table 3.1: The fishery data, Normal mixtures with $k = 3$

Parameter	Posterior Mean	Posterior SD	Lower 2.5%	Upper 2.5%
p_1	0.107	0.022	0.067	0.153
p_2	0.372	0.054	0.264	0.480
p_3	0.520	0.058	0.413	0.641
μ_1	3.285	0.099	3.097	3.499
μ_2	5.171	0.072	5.030	5.308
μ_3	7.294	0.269	6.764	7.822
σ_1	0.373	0.070	0.261	0.536
σ_2	0.482	0.066	0.363	0.618
σ_3	1.749	0.140	1.481	2.036

draws after a burn-in period of 3000 draws. The parameter estimation for $k = 3$ is given in Table 3.1. Figure 3.7 shows a two-dimensional scatter plot of the MCMC draws of $(\mu^{(m)}, \sigma^{(m)})$. We can see three well-separated clusters, and the means are different between the groups. The variances are nearly identical for group 1 and group 2 with smaller fish, but are quite large for the group with the largest fish. In this case, while σ is not a suitable variable for ordering the component estimates, it is sensible to order with respect to μ.

Figure 3.8 shows that the predictive density based on the 3-component Normal mixture is inadequate for the observed data, calling for more components. We thus fitted a Normal mixture with $k = 4$, using the same Gibbs sampler. A two-dimensional scatter plot of the draws of $(\mu^{(m)}, \sigma^{(m)})$ is given in Figure 3.9. Groups 1 and 2 remain roughly where they are as in the three-component mixture, whereas the third group seems to be split into two separate groups. The variance of the third group is now much smaller and similar to that of groups 1 and 2; the variance of the fourth group is still considerably large.

The MCMC draws from the marginal bivariate density $p(\mu_k, \mu_l|y)$, with $k \neq l$, are shown

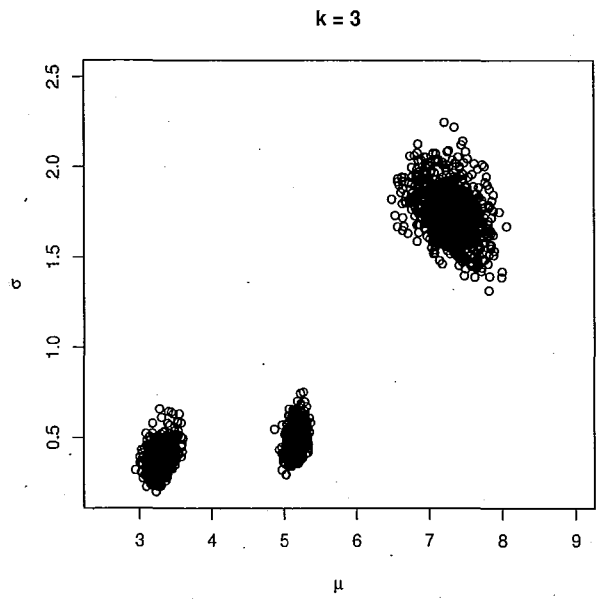

Figure 3.7: MCMC draws from a 3-component Normal mixture of the fishery data for μ_k against σ_k.

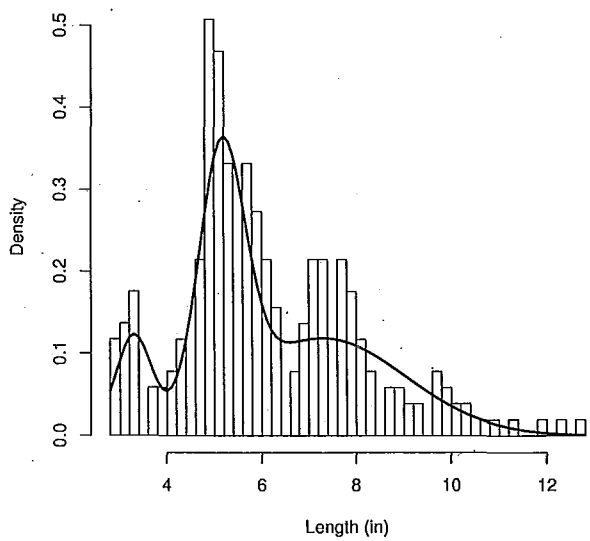

Figure 3.8: Predictive density based on the 3-component Normal mixture.

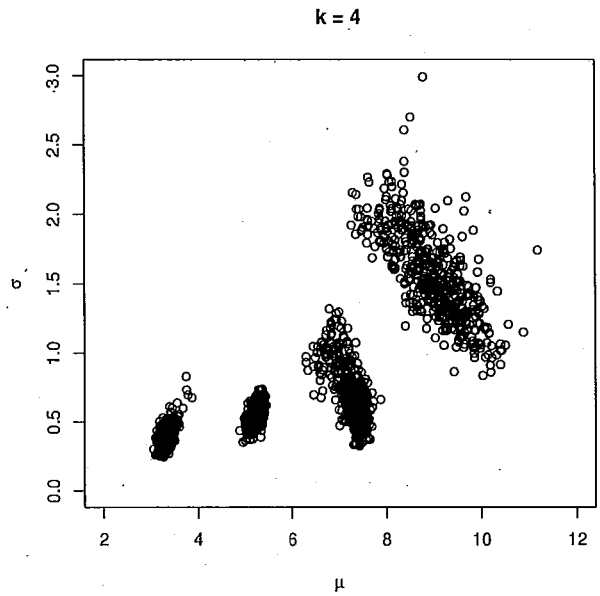

Figure 3.9: MCMC draws from a 4-component Normal mixture of the fishery data for μ_k against σ_k.

in Figure 3.10. For $k = 3$, all of the $k(k-1) = 6$ modes are far away from the diagonal line $\mu_k = \mu_l$ which corresponds to a model where two components have equal means. Therefore the constraint $\mu_1 \leq \mu_2 \leq \mu_3$ induces a unique labeling. However, for $k = 4$, the MCMC draws from the marginal bivariate density $p(\mu_k, \mu_l|y)$ indicate that not all of the $k(k-1) = 12$ possible modes are bounded away from the line $\mu_k = \mu_l$. Thus, the constraint $\mu_1 \leq \cdots \leq \mu_4$ alone does not induce a unique labeling.

In this case, we find that the label switching effect is very likely to occur if we use non-informative data-dependent priors on σ^2, such as $IG(1.28, 0.36s^2)$ suggested by Raftery (1996b) and $IG(2.5, 0.5s^2)$ by Bensmail *et al.* (1997), where s^2 denotes the sample variance. However, this problem can be solved if we arrange the mixture components in order of non-

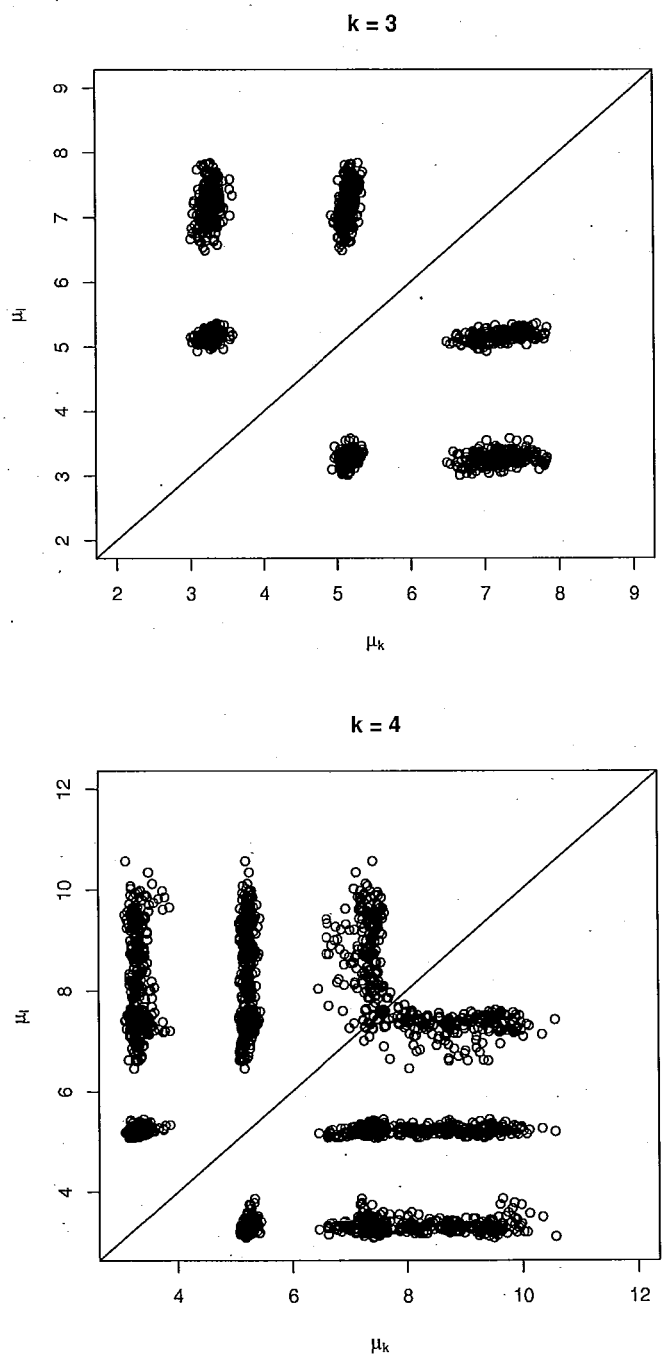

Figure 3.10: MCMC draws from a Normal mixture of the fishery data for μ_k against μ_l for $k = 3$ (top) and $k = 4$ (bottom).

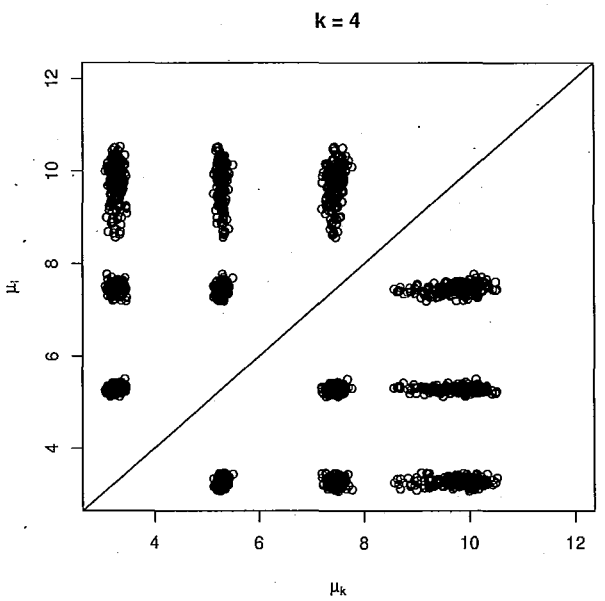

Figure 3.11: MCMC draws from a Normal mixture of the fishery data for μ_k against μ_l for $k = 4$, using $IG(10, 1)$ on σ^2.

decreasing means whilst choosing priors that are more informative. Here we used $IG(10, 0.5)$ instead to reduce the simulated values of σ^2. Figure 3.11 illustrates the success of our approach. It shows that we have achieved a unique labeling and can proceed with parameter estimation. The estimates of the parameters for $k = 4$ are given in Table 3.2.

The predictive density based on the 4-component Normal mixture indicates a reasonable fit to the observed data, as in Figure 3.12. Based on these results, the mean lengths (in) of the 256 snappers are 3.262 ± 0.076, 5.289 ± 0.065, 7.461 ± 0.105, and 9.794 ± 0.330, respectively; the length standard deviations are 0.316 ± 0.039, 0.584 ± 0.055, 0.484 ± 0.090, and 0.938 ± 0.145, respectively; the proportions for the 4 age groups are 0.104 ± 0.021, 0.543 ± 0.038, 0.236 ± 0.040, and 0.117 ± 0.030, respectively.

Table 3.2: The fishery data, Normal mixtures with $k = 4$

Parameter	Posterior Mean	Posterior SD	Lower 2.5%	Upper 2.5%
p_1	0.104	0.021	0.067	0.149
p_2	0.543	0.038	0.463	0.613
p_3	0.236	0.040	0.162	0.320
p_4	0.117	0.030	0.067	0.187
μ_1	3.262	0.076	3.115	3.420
μ_2	5.289	0.065	5.159	5.417
μ_3	7.461	0.105	7.234	7.647
μ_4	9.794	0.330	9.058	10.381
σ_1	0.316	0.039	0.250	0.406
σ_2	0.584	0.055	0.482	0.697
σ_3	0.484	0.090	0.342	0.692
σ_4	0.938	0.145	0.699	1.264

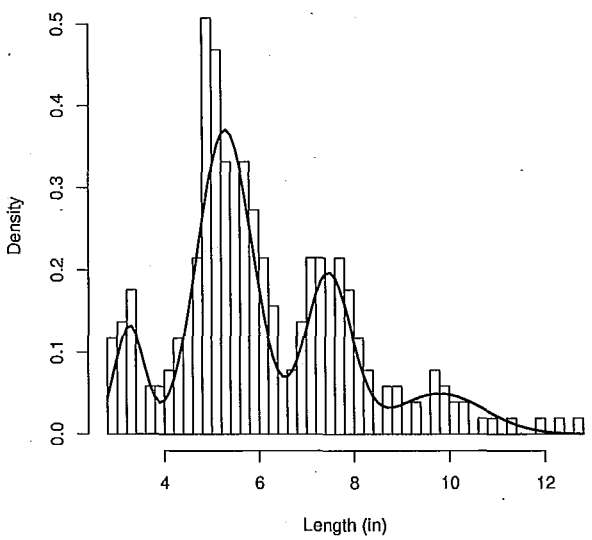

Figure 3.12: Predictive density based on the 4-component Normal mixture.

Chapter 4

Bayesian Modeling and Inference for Mixture Distributions with Unknown Number of Components

> One of the things we do not know is the number of things we do not know.
>
> — P. Green (1996)

A number of approaches have been proposed for estimating k, the number of components in a finite mixture model. In particular, Richardson and Green (1997) showed how to use the reversible jump Markov chain Monte Carlo (RJMCMC) algorithm to model the number of components and the mixture parameters simultaneously. Alternatively, selection of k can be based on a comparison of models fitted with different numbers of components by the means of

some joint measure of model fit and model complexity. Especially in Bayesian modeling, the deviance information criterion (DIC, Spiegelhalter *et al.*, 2002) and the penalized expected deviance (PED, Plummer, 2008) are popular.

We introduce a recent `R` package `mixAK` (Komárek, 2009) which implements the reversible jump, birth death and other trans-dimensional MCMC algorithms for mixture problems. It works either with a pre-specified number of components or with the number of components estimated jointly with the remaining model parameters.

This chapter is organized as follows. Section 1 displays an introduction to the Reversible Jump MCMC algorithm for mixtures. Section 2 reviews the DIC and PED methods for Bayesian model comparison. A fully Bayesian mixture analysis of the acidity data using `mixAK` is illustrated in Section 3. We present a discussion in Section 4.

4.1 Reversible Jump MCMC

4.1.1 Hierarchical Model and Priors

In the Bayesian framework, the number of components k, the mixing weights $\boldsymbol{w}$ and component parameters $\boldsymbol{\theta}$ are regarded as drawn from appropriate prior distributions. The joint distribution of all variables can be written as

$$p(k, \boldsymbol{w}, \boldsymbol{z}, \boldsymbol{\theta}, \boldsymbol{y}) = p(k)p(\boldsymbol{w}|k)p(\boldsymbol{z}|\boldsymbol{w}, k)p(\boldsymbol{\theta}|\boldsymbol{z}, \boldsymbol{w}, k)p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{z}, \boldsymbol{w}, k),$$

by allowing the priors for k, $\boldsymbol{w}$ and $\boldsymbol{\theta}$ to depend on hyper-parameters $\boldsymbol{\lambda}$, $\boldsymbol{\delta}$ and $\boldsymbol{\eta}$ respectively. The joint distribution of all variables can be expressed by the factorization

$$p(\boldsymbol{\lambda}, \boldsymbol{\delta}, \boldsymbol{\eta}, k, \boldsymbol{w}, \boldsymbol{z}, \boldsymbol{\theta}, \boldsymbol{y}) = p(\boldsymbol{\lambda})p(\boldsymbol{\delta})p(\boldsymbol{\eta})p(k|\boldsymbol{\lambda})p(\boldsymbol{w}|k, \boldsymbol{\delta})p(\boldsymbol{z}|\boldsymbol{w}, k)p(\boldsymbol{\theta}|k, \boldsymbol{\eta})p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{z}).$$

In the univariate Normal mixture setting, the parameter $\boldsymbol{\theta}$ is a vector of pairs $(\mu_j,\ \sigma_j^2)$, $j = 1, 2, \ldots, k$. Richardson and Green (1997) only considered Bayesian Normal mixtures estimation in the setups where strong prior information on the mixture parameters are unavailable. Dirichlet prior for $\boldsymbol{w}$, Normal and Gamma priors for μ_j and σ_j^2 are used in the following form:

$$\begin{aligned} \boldsymbol{w} &\sim D(\delta, \delta, \ldots, \delta), \\ \mu_j &\sim N(\xi, \kappa^{-1}), \\ \sigma_j^2 &\sim \Gamma(\alpha, \beta). \end{aligned}$$

The $N(\xi, \kappa^{-1})$ is taken to be flat over an interval of variation of the data, by setting ξ equal to the midpoint of this interval and setting κ equal to a small multiple of $1/R^2$, where R is the length of the interval.

Concerning $\sigma_j^2 \sim \Gamma(\alpha, \beta)$, Richardson and Green (1997) introduced an additional hierarchical level by allowing β to follow a $\Gamma(g, h)$ distribution. They set $\alpha > 1 > g$ to express the belief that the σ_j^2 are similar. The scale parameter h is a small multiple of $1/R^2$. For k, a uniform distribution between 1 and a pre-specified integer k_{max} is used. Finally, λ and δ are

held fixed.

The completed hierarchical model is displayed as a directed acyclic graph (DAG) in Figure 4.1, in which the square boxes represent fixed or observed quantities and circles the unknowns.

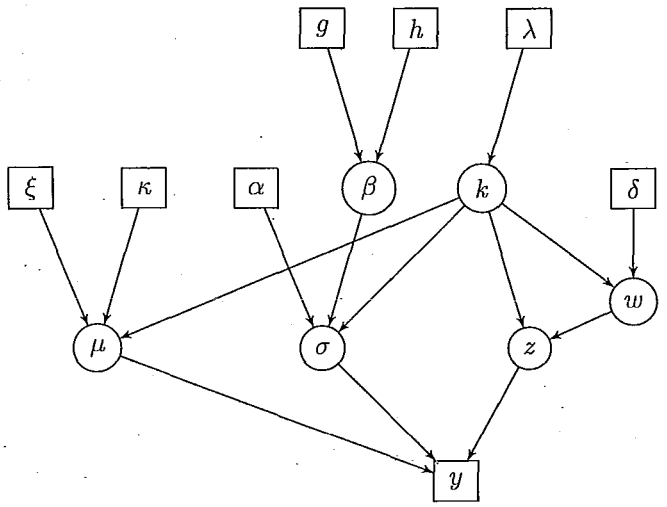

Figure 4.1: DAG of a Normal mixture model.

With the specification of the proper conjugate priors above, the intent is to avoid using strong prior information on the mixture parameters. It is not possible to have fully non-informative priors and obtain proper posterior distributions, because there is always the possibility that no observations will be allocated to one or more components. Therefore, independent improper non-informative priors cannot be used (Diebolt and Robert, 1994; Roeder and Wasserman, 1997).

4.1.2 Reversible Jump Moves for Normal Mixtures

The key idea in RJMCMC is to allow moves between parameter subspaces of different dimensionality by permitting a series of different "move types". For a Normal mixture model, six move types are involved:

(a) updating the weights $\boldsymbol{w}$;

(b) updating the parameters (μ, σ);

(c) updating the allocation $\boldsymbol{z}$;

(d) updating the hyper-parameter β;

(e) splitting one mixture component into two, or combining two into one;

(f) the birth or death of an empty component.

Move types (a), (b), (c) and (d) are conventional, following Diebolt and Robert (1994); they do not alter the dimension of the complete parameter vector $(\beta, \mu, \sigma, k, \boldsymbol{w}, \boldsymbol{z})$. The only randomness is the random choice between splitting and combining in move (e), or birth and death in move (f).

The split-combine move consists of either splitting an existing component into two new components or combining two existing components into a new one. First, a random choice is made whether to perform the split or combine move. Given k, the probability of attempting the split move is π_k^{split} and the probability of attempting the combine move is $\pi_k^{combine} = 1 - \pi_k^{split}$.

When a combine move is attempted, a pair of neighboring components is chosen at random. Let k_1 and k_2 be such that $\mu_{k_1} < \mu_{k_2}$ and there is no other mixture mean in the

interval $[\mu_{k_1}, \mu_{k_2}]$. Then we transform the current vector of pairs $w_{k_1}, w_{k_2}, \mu_{k_1}, \mu_{k_2}, \sigma^2_{k_1}, \sigma^2_{k_2}$ to the new vector $(w_{k^*}, \mu_{k^*}, \sigma^2_{k^*})$, given by

$$\begin{aligned} w_{k^*} &= w_{k_1} + w_{k_2}, \\ \mu_{k^*} &= \frac{w_{k_1}\mu_{k_1} + w_{k_2}\mu_{k_2}}{w_{k^*}}, \\ \sigma^2_{k^*} &= \frac{w_{k_1}(\mu^2_{k_1} + \sigma^2_{k_1}) + w_{k_2}(\mu^2_{k_2} + \sigma^2_{k_2})}{w_{k^*}} - \mu^2_{k^*}. \end{aligned}$$

With components k_1 and k_2 replaced by k^* and the number of mixture components k decreased by 1, this move is then either accepted or rejected with a certain Metropolis-Hastings probability (Richardson and Green, 1997). In the case of acceptance, observations allocated originally in k_1 and k_2 are re-allocated in the new component k^*. Note that once the choice of the pair has been made, the combine move is deterministic.

In the split move, a component k^* is chosen at random and two new components k_1 and k_2 are proposed such that the split move is a reverse to the combine move. The weights w_{k_1}, w_{k_2}, the means μ_{k_1}, μ_{k_2}, and the variances $\sigma^2_{k_1}$, $\sigma^2_{k_2}$ of the two new components are given by

$$\begin{aligned} w_{k_1} &= w_{k^*}u_1, \quad w_{k_2} = w_{k^*}(1-u_1), \\ \mu_{k_1} &= \mu_{k^*} - u_2\sigma_{k^*}\sqrt{\frac{w_{k_2}}{w_{k_1}}}, \quad \mu_{k_2} = \mu_{k^*} - u_2\sigma_{k^*}\sqrt{\frac{w_{k_1}}{w_{k_2}}}, \\ \sigma^2_{k_1} &= u_3(1-u_2^2)\sigma^2_{k^*}\frac{w_{k^*}}{w_{k_1}}, \quad \sigma^2_{k_1} = u_3(1-u_2^2)\sigma^2_{k^*}\frac{w_{k^*}}{w_{k_2}}, \end{aligned}$$

where $\boldsymbol{u} = (u_1, u_2, u_3)'$ is an auxiliary random vector whose components are generated randomly and independently, and u_l are from Beta(a_l, b_l), $l = 1, 2, 3$. Again, once the choice of

the pair has been made, the split proposal is deterministic. The birth-death move (f) consists of either proposing a new component (birth) or deleting one of the empty components (death). Similarly to the split-combine move, a random choice is made whether to perform the birth or the death move. Given k, the probability of attempting the birth move is π_k^{birth} and the probability of attempting the death move is $\pi_k^{death} = 1 - \pi_k^{birth}$.

When a birth move is attempted, a new component weight w_{k^*}, mean μ_{k^*} and variance $\sigma_{k^*}^2$ are sampled from the prior distributions, the weights of existing components are re-scaled to satisfy the sum-to-one constraint and the whole proposal is accepted with a certain Metropolis-Hastings probability (Richardson and Green, 1997).

When a death move is attempted, one of the "empty" components that have no observations is chosen at random. The proposal consists of deleting this component and re-scaling the remaining weights to sum to one. Similarly to the previous case, the whole proposal is accepted with a certain Metropolis-Hastings probability (Richardson and Green, 1997).

4.2 Bayesian Model Comparison

4.2.1 Introduction

Model selection and model comparison are a fundamental step of data analysis. Within Bayesian modeling, a number of approaches have been proposed for these problems. The Bayes factor (Kass and Raftery, 1995), which quantifies the weight of evidence in favor of one model over another, is widely recognized as a formal solution. However, Bayes factors

have some practical limitations: they are undefined when the model parameters are given improper prior distributions, and are numerically unstable when proper but diffuse reference priors are used. To overcome these limitations, modifications to the Bayes factor have been proposed (O'Hagan, 1995; Berger and Pericchi, 1996) by sacrificing a small fraction of the data to estimate model parameters and using the remainder to calculate the Bayes factor.

Another approach is cross-validation (Geisser and Eddy, 1979), in which the notion of splitting the data between parameter estimation and assessment of model adequacy is used as well. In this case, a model is considered useful if, given a set of data, it makes accurate out-of-sample predictions. A third approach to Bayesian model choice is based on hypothetical replicates from the same process that generated the data. In this posterior predictive approach, replicate data are simulated from the posterior distribution, and the adequacy of the model is assessed by the closeness of these replicates to the original data. The approach is recommended as a general framework for model criticism by Gelman *et al.* (2002).

A more recent addition to the collection of Bayesian model-choice methods is the deviance information criterion (DIC) (Spiegelhalter *et al.*, 2002), a Bayesian analogue of classical model choice criteria, such as the Akaike information criterion (AIC). DIC combines a measure of model fit with a measure of model complexity. It is simple to calculate using Markov chain Monte Carlo simulation and has been implemented in the `WinBUGS` software package (Spiegelhalter *et al.*, 2004). Plummer (2008) provided a formal justification for DIC by demonstrating the link between DIC and cross-validation. He showed that DIC is an approximation to a penalized loss function based on the deviance, with a penalty derived from a cross-validation argument.

4.2.2 The Deviance Information Criterion

A general approach to compare complex models based on the samples from the posterior distribution was suggested by Spiegelhalter *et al.* (2002), who defined the deviance information criterion (DIC) as:

$$DIC = \bar{D} + p_D,$$

where the expected deviance $\bar{D} = E(D(\boldsymbol{\theta})|\mathbf{Y})$ is considered to be a measure of model fit, and $p_D = \bar{D} - D(\bar{\boldsymbol{\theta}})$, called the "effective number of parameters", is a measure of model complexity, where $\bar{\boldsymbol{\theta}} = E(\boldsymbol{\theta}|\boldsymbol{Y})$.

Note that DIC can also be written as $D(\bar{\boldsymbol{\theta}}) + 2p_D$, since

$$\begin{aligned} DIC = \bar{D} + p_D &= \bar{D} + (\bar{D} - D(\bar{\boldsymbol{\theta}})) \\ &= 2\bar{D} + (D(\bar{\boldsymbol{\theta}}) - 2D(\bar{\boldsymbol{\theta}})) \\ &= D(\bar{\boldsymbol{\theta}}) + 2(\bar{D} - D(\bar{\boldsymbol{\theta}})) \\ &= D(\bar{\boldsymbol{\theta}}) + 2p_D. \end{aligned}$$

In this form it resembles the classical Akaike Information Criterion (AIC, Akaike, 1974)

$$D(\hat{\boldsymbol{\theta}}) + 2p,$$

where $\hat{\boldsymbol{\theta}}$ is the maximum likelihood estimate of $\boldsymbol{\theta}$ and p is the number of parameters. For

non-hierarchical models with a non-informative prior on $\boldsymbol{\theta}$, DIC = AIC.

4.2.3 Loss Functions and Penalized Losses

In an ideal situation, two independent data sets are available: a set of training data $\boldsymbol{Y}_{tr}$ and a set of test data $\boldsymbol{Y} = Y_1, \ldots, Y_n$. Suppose we assess the model adequacy by a loss function $L(\boldsymbol{Y}, \boldsymbol{Y}_{tr})$, which measures the ability of the model to make accurate predictions of $\boldsymbol{Y}$ from $\boldsymbol{Y}_{tr}$. Given a set of candidate models for $\boldsymbol{Y}$ and $\boldsymbol{Y}_{tr}$, we would choose the one with the smallest loss, or more realistically, we would choose a subset of models with losses close to the minimum for further consideration.

Based on the deviance, consider two loss functions: the "plug-in deviance":

$$L^p(\boldsymbol{Y}, \boldsymbol{Y}_{tr}) = -2\log[p\{\boldsymbol{Y}|\ \bar{\boldsymbol{\theta}}(\boldsymbol{Y}_{tr})\}],$$

where $\bar{\boldsymbol{Y}}_{tr} = E(\theta|\boldsymbol{Y}_{tr})$, and the "expected deviance":

$$L^e(\boldsymbol{Y}, \boldsymbol{Y}_{tr}) = -2\int d\boldsymbol{\theta} p(\boldsymbol{\theta}|\boldsymbol{Y}_{tr}) \log\{p(\boldsymbol{Y}|\boldsymbol{\theta})\},$$

where the expectation is taken over the posterior distribution of $\boldsymbol{\theta}$ given $\boldsymbol{Y}_{tr}$, and the test data $\boldsymbol{Y}$ are fixed. Since the deviance is defined only up to an additive function of the data, both these loss functions are relative losses. That is, only the difference in loss between two candidate models is meaningful.

Although both L^p and L^e are derived from the deviance function, there are some dif-

ferences between them. The plug-in deviance is sensitive to re-parametrization. Changing the coordinates of $\boldsymbol{\theta}$ changes the definition of the posterior expectation, and hence the loss function L^p. It gives equal loss to all models that yield the same posterior expectation of $\boldsymbol{\theta}$, regardless of the precision of this estimate. On the other hand, the expected deviance is coordinate free. Furthermore, it is a function of the full posterior of $\boldsymbol{\theta}$ given $\boldsymbol{Y}_{tr}$, and therefore takes precision of the estimates into account.

When there are no training data $\boldsymbol{Y}_{tr}$, the test data $\boldsymbol{Y}$ must be used both to estimate $\boldsymbol{\theta}$ and to assess the adequacy of the model. A natural loss function would be $L(\boldsymbol{Y}, \boldsymbol{Y})$, which was referred to as the "exact replicate" form of the loss function by Plummer (2008). In general, $L(\boldsymbol{Y}, \boldsymbol{Y})$ gives an optimistic assessment of model adequacy, since the same data are used twice, for calculating the posterior distribution of the model parameters and in place of new test data. The degree of optimism can be estimated for loss functions that can be decomposed into the sum of individual contributions:

$$L(\boldsymbol{Y}, \boldsymbol{Y}_{tr}) = \sum_{i=1}^{n} L(Y_i, \boldsymbol{Y}_{tr}).$$

For such loss functions, the extent to which $L(Y_i, \boldsymbol{Y})$ overstates the model adequacy can be assessed by comparing it with the cross-validation loss $L(Y_i, \boldsymbol{Y}_{-i})$, where $\boldsymbol{Y}_{-i}$ denotes the set of observations $Y_1, \ldots, Y_n$ with Y_i removed. The expected decrease in loss from using $L(Y_i, \boldsymbol{Y})$ instead of $L(Y_i, \boldsymbol{Y}_{-i})$ is

$$p_{opt_i} = E\{L(Y_i, \boldsymbol{Y}_{-i}) - L(Y_i, \boldsymbol{Y}) | \ \boldsymbol{Y}_{-i}\}.$$

Following the terminology of Efron (1983), p_{opt_i} is called the "optimism" of $L(Y_i, \boldsymbol{Y})$. The penalized loss function $L(Y_i, \boldsymbol{Y}) + p_{opt_i}$ has the same expectation given $\boldsymbol{Y}_{-i}$ as the cross-validation loss $L(Y_i, \boldsymbol{Y}_{-i})$. The two loss functions are therefore equivalent to an observer who has not seen Y_i. Applying the same argument to each observation Y_i in turn, Plummer (2008) proposed to use the sum of the penalized loss functions $L(Y_i, \boldsymbol{Y}) + p_{opt}$ to assess model adequacy, where the total optimism $p_{opt} = \sum_i p_{opt_i}$ is a rational cost that must be paid for using the observed data $\boldsymbol{Y}$ twice. When there are no influential observations $p_{D_i} << 1$ such that

$$\sum_i p_{D_i}/(1 - p_{D_i}) \approx \sum_i pD_i = p_D,$$

DIC is an approximation to the penalized plug-in deviance.

4.2.4 Challenges

Despite the practical advantages of DIC, its theoretical foundations remain controversial. DIC inherits some of the limitations of AIC, including that it is restricted to nested models; it is not consistent (given a set of nested models, DIC will tend to choose a model that is too large as n goes to ∞); outside of exponential family models, p_D is not easy to calculate; it is not coordinate free. Although various ad hoc extensions or modifications to DIC have been proposed (Gelman *et al.*, 2002; Plummer, 2002; Celeux *et al.*, 2006a), none of them is more convincing than DIC itself.

In the finite mixture model framework, it is challenging to use DIC as model-choice

criterion, as noted by several contributors to the discussion of Spiegelhalter *et al.* (2002). In such models, the posterior expectation is not a suitable plug-in estimate for the model parameters since it lies in between multiple modes of the posterior density, and alternative plug-in estimators are hard to define. Celeux *et al.* (2006a) applied eight variations of DIC to mixture distributions, but were unable to recommend any of them in the end, concluding that DIC was neither a well-defined criterion nor a solution to a well-defined optimization problem.

The difficulties in defining a plug-in estimate rule out the use of L^p as a loss function. Consequently, Plummer (2008) considered the penalized expected deviance L^e as a loss function. In the next section, we will use the `R` package `mixAK` to analyze the acidity data and illustrate how DIC and PED can be used to assess the number of components in a mixture model.

4.3 Practical Example: The Acidity Data

A histogram of the acidity data is given in Figure 4.2. The following prior distributions, their parameters and parameters of the proposal densities have been used: uniform prior on k with $k_{max} = 30$, $\delta = 1$, semiconjugate prior on $\boldsymbol{\mu}$ and $\boldsymbol{Q}$ with $\boldsymbol{\xi}_k = 4.73$, $\boldsymbol{D}_k = 1.94$ for all k, $\zeta = 4$, $g_1 = 0.2$, $h_1 = 0.29$, $a_1 = b_1 = 2$, $a_2 = b_2 = 2$, $a_3 = b_3 = 1$. The data were neither shifted nor scaled before running the RJMCMC, that is, $\boldsymbol{m} = 0$, $\boldsymbol{S} = 1$. We report results based on 100,000 iterations of RJMCMC obtained after a burn-in period of 100,000 iterations.

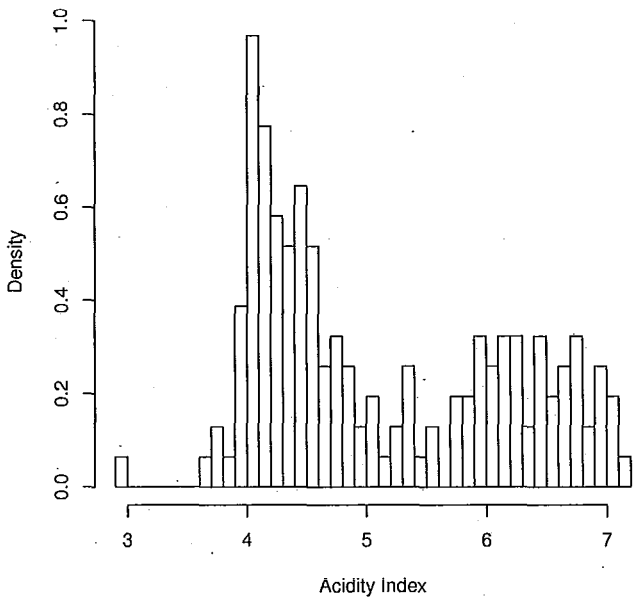

Figure 4.2: Histogram of the acidity data.

During the course of the RJMCMC, the chain visited models with the number of mixture components k ranging from 1 to 8 with the highest posterior probabilities of 0.71, 0.24, and 0.04 for $k = 2$, 3, and 4, respectively. For the remaining values of k, the posterior probability was lower than 0.01, see Figure 5. The split-combine move was accepted in around 4.3% of cases and the birth-death in around 2.4% of cases. We will discuss the sensitivity of posterior distributions, as well as the performance of MCMC sampler in §4.4.

4.3.1 Preparation of MCMC Simulation

```
# Load packages and data
> library(mixAK)
> library(coda)
> data(Acidity)
```

```
> mixdat <- Acidity

# Summary of the data
> summary(mixdat)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  2.929   4.219   4.727   5.105   6.075   7.105

# Length of the MCMC simulation:
# 100,000 burn-in iterations
# 100,000 additional iterations for inference
> nMCMC <- c(burn=100000, keep=100000, thin=1, info=2000)
```

4.3.2 Specification of the Prior Distributions

```
# The minimal specification of the prior distribution
# RJMCMC with default values for all prior parameters (Not run)
RJPrior1 <- list(priorK="uniform", Kmax=30)
RJModel1 <- NMixMCMC(y0=mixdat, prior=RJPrior1, nMCMC=nMCMC,
scale=list(shift=0, scale=1), PED=TRUE)
```

In the following analysis, we will use the same prior hyper-parameters and tuning parameters as in Richardson and Green (1997). Note that the prior for the mixture inverse variances in mixAK is parameterized in terms of the Wishart distribution while a Gamma distribution was used by Richardson and Green (1997).

```
# Use the same prior hyper-parameters as in Richardson and Green (1997)
# Here priors for the mixture inverse variances follow a Wishart distribution
# xi=2*2 corresponds to alpha=2 in [RG]
# h=5/R^2 corresponds to h=10/R^2 in [RG]
# D=(R/3)^2 instead of D=R^2 to allow higher k
> R <- diff(range(mixdat))
> RJPrior2 <- list(priorK="uniform", Kmax=30, priormuQ="independentC",
+ xi=median(mixdat), D=(R/3)^2, g=0.2, h=5/R^2, zeta=4)

# Use the same tuning parameters as in Richardson and Green (1997)
```

```
> parRJMCMC2 <- list(par.u1=c(2,2), par.u2=c(2,2), par.u3=c(1,1))

# Running the MCMC simulation
> set.seed(12345)
> RJModel2 <- NMixMCMC(y0=mixdat, prior=RJPrior2, RJMCMC=parRJMCMC2,
+ nMCMC=nMCMC, scale=list(shift=0, scale=1),PED=TRUE)

Chain number 1
==============
MCMC sampling started on Thu Jun 24 16:15:00 2010.
Burn-in iteration 100000
Iteration 200000
MCMC sampling finished on Thu Jun 24 16:15:59 2010.

Chain number 2
==============
MCMC sampling started on Thu Jun 24 16:16:00 2010.
Burn-in iteration 100000
Iteration 200000
MCMC sampling finished on Thu Jun 24 16:16:56 2010.

Computation of penalized expected deviance started on Thu Jun 24 16:16:57 2010.
Computation of penalized expected deviance finished on Thu Jun 24 16:18:42 2010.

# Acceptance rates of different move types
> print(RJModel2[[1]]$moves)
                  Performed Accepted Proportion accepted (%)
Gibbs with fixed K    100000   100000                  100.00
Split                  49885     1584                    3.18
Combine                50115     2567                    5.12
Birth                  50060     1565                    3.13
Death                  49940      582                    1.17
> print(RJModel2[[2]]$moves)
                  Performed Accepted Proportion accepted (%)
Gibbs with fixed K    100000   100000                  100.00
Split                  50230     1629                    3.24
Combine                49770     2663                    5.35
Birth                  49842     1707                    3.42
Death                  50158      672                    1.34
```

4.3.3 Posterior Inference

```
# Posterior summary of the fitted model
> print(RJModel2)

     Normal mixture with at most 30 components estimated using RJMCMC
     ================================================================
Posterior distribution of K:
----------------------------
            1       2       3       4       5       6       7     8
Chain 1 0.00019 0.70782 0.24090 0.04432 0.00632 0.00040 0.00005 0e+00
Chain 2 0.00005 0.66921 0.26782 0.05454 0.00695 0.00122 0.00017 4e-05

Posterior summary statistics for moments of mixture for original data:
-----------------------------------------------------------------------
Mean:
           Mean   Std.Dev.     Min.     2.5%  1st Qu.   Median
Chain 1 5.105700 0.09008539 2.591163 4.945561 5.049160 5.104579
Chain 2 5.105358 0.08982873 2.644795 4.944281 5.048685 5.104114
         3rd Qu.    97.5%     Max.
Chain 1 5.161148 5.271408 8.758423
Chain 2 5.160636 5.271328 9.957364

Standard deviation:
           Mean   Std.Dev.       Min.      2.5%  1st Qu.   Median
Chain 1 1.038854 0.04881007 0.08640945 0.9608796 1.011393 1.038057
Chain 2 1.039019 0.04834467 0.72040780 0.9606500 1.011429 1.038102
         3rd Qu.    97.5%    Max.
Chain 1 1.064918 1.119777 4.80476
Chain 2 1.065335 1.119950 6.11653

# Grid of values for evaluating and plotting predictive densities
> ygrid <- seq(2, 8, length=200)

# Computation of the predictive density
> PDensRJ2<-list()
> PDensRJ2[[1]]<-NMixPredDensMarg(RJModel2[[1]], grid=ygrid)
> PDensRJ2[[2]]<-NMixPredDensMarg(RJModel2[[2]], grid=ygrid)

# Plot the predictive density, see Figure 2
> plot(PDensRJ2[[1]], xlab="Acidity Index", col=1)

# Plots of conditional predictive densities given K
```

```
# along with the overall predictive density, see Figure 3
> plot(PDensRJ2[[1]], K=c(2:5), lty=c(1,2,3,4), xlab="Acidity Index",
+ ylim=c(0,.7), col=1)

# Plots of predictive densities with histograms, see Figure 4
> par(mfrow=c(2,1))
## Chain1
> hist(mixdat,prob=TRUE, breaks=40,
+ xlab="Acidity Index", ylab="Density",main="Chain 1")
> lines(PDensRJ2[[1]]$x$x1,PDensRJ2[[1]]$dens[[1]],lwd=2)
## Chain2
> hist(mixdat,prob=TRUE, breaks=40,
+ xlab="Acidity Index", ylab="Density",main="Chain 1")
> lines(PDensRJ2[[2]]$x$x1,PDensRJ2[[2]]$dens[[1]],lwd=2)

# Traceplot of K of last 10000 iterations drawn, see Figure 5
> chKpart <- mcmc(RJModel2[[CH]]$K, start=start, end=end)
> traceplot(chKpart, smooth=FALSE, main="K")
```

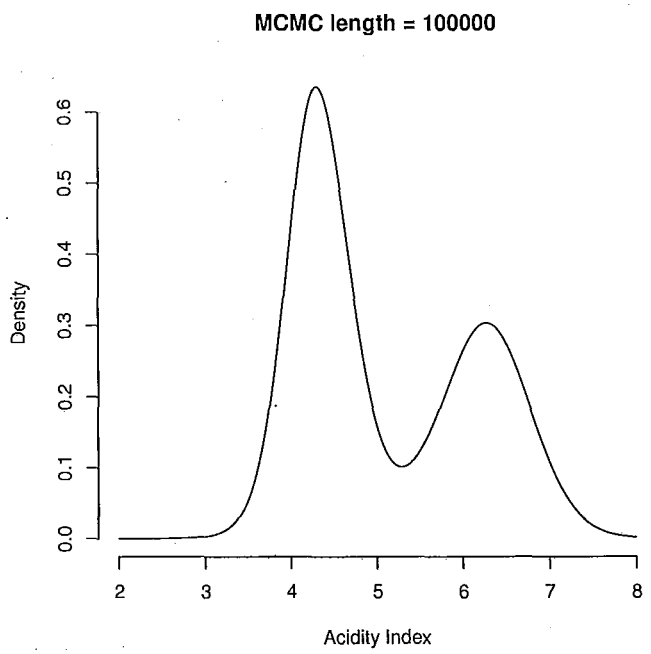

Figure 4.3: Predictive density based on the model with a random number of mixture components from Chain 1.

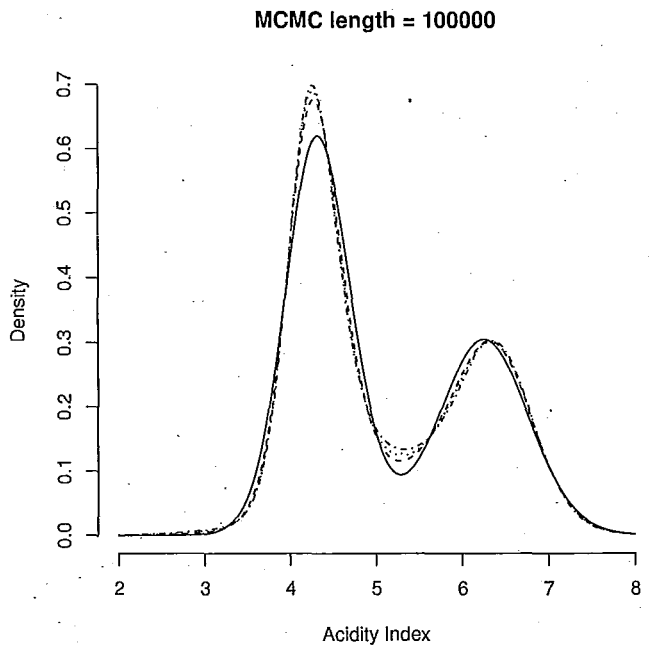

Figure 4.4: Overall predictive density and conditional predictive densities for $K = 2$, 3, 4, 5.

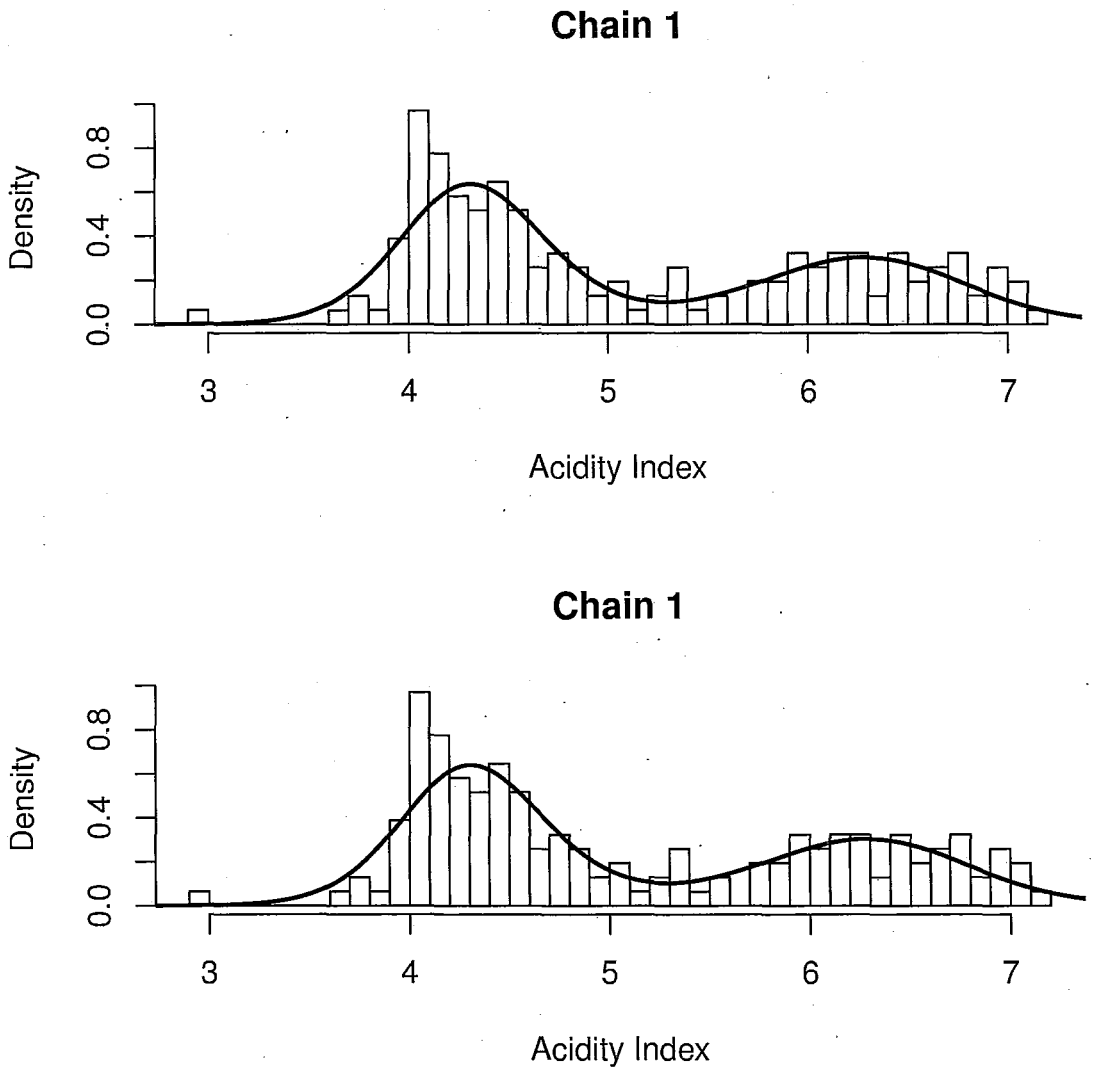

Figure 4.5: Predictive density based on the model with a random number of mixture components.

4.3.4 Convergence Diagnostics

```
# Converting the chains into mcmc objects
# Single chain convergence diagnostics using Chain 1
> CH<-1

# Converting the chains into mcmc objects
> start<-RJModel2[[CH]]$nMCMC["burn"]+1
> end<-RJModel2[[CH]]$nMCMC["burn"]+RJModel2[[CH]]$nMCMC["keep"]
> chK<-mcmc(RJModel2[[CH]]$K,start=start,end=end)
> chgammaInv<-mcmc(RJModel2[[CH]]$gammaInv,start=start,end=end)
> chmixture<-mcmc(RJModel2[[CH]]$mixture,start=start,end=end)
> chdeviance<-mcmc(RJModel2[[CH]]$deviance,start=start,end=end)

# Traceplot of K of last 10000 iterations drawn, see Figure 5
> chKpart <- mcmc(RJModel2[[CH]]$K, start=start, end=end)
> traceplot(chKpart, smooth=FALSE, main="K")

# Posterior density estimates for selected parameters, see Figure 6
> par(mfrow=c(2,2), bty="n")
> densplot(chK, show.obs=FALSE, main="K")
> densplot(chgammaInv[,"gammaInv1"], show.obs=FALSE,
+ main="gamma^{-1}", xlim=c(0,4))
> densplot(chmixture[,"y.Mean.1"], show.obs=FALSE,
+ main="EY", xlim=c(4.5,6))
> densplot(chmixture[,"y.SD.1"], show.obs=FALSE,
+ main="sd(Y)", xlim=c(.8,1.4))

# Autocorrelation plots for selected parameters, see Figure 7
> par(mfrow=c(2,2), bty="n")
> autocorr.plot(chK, auto.layout=FALSE, ask=FALSE, lwd=2, main="K")
> autocorr.plot(chgammaInv[,"gammaInv1"], auto.layout=FALSE, ask=FALSE,
+ lwd=2,main="gamma^{-1}")
> autocorr.plot(chmixture[,"y.Mean.1"], auto.layout=FALSE, ask=FALSE,
+ lwd=2,main="EY")
> autocorr.plot(chmixture[,"y.SD.1"], auto.layout=FALSE, ask=FALSE,
+ lwd=2,main="sd(Y)")
```

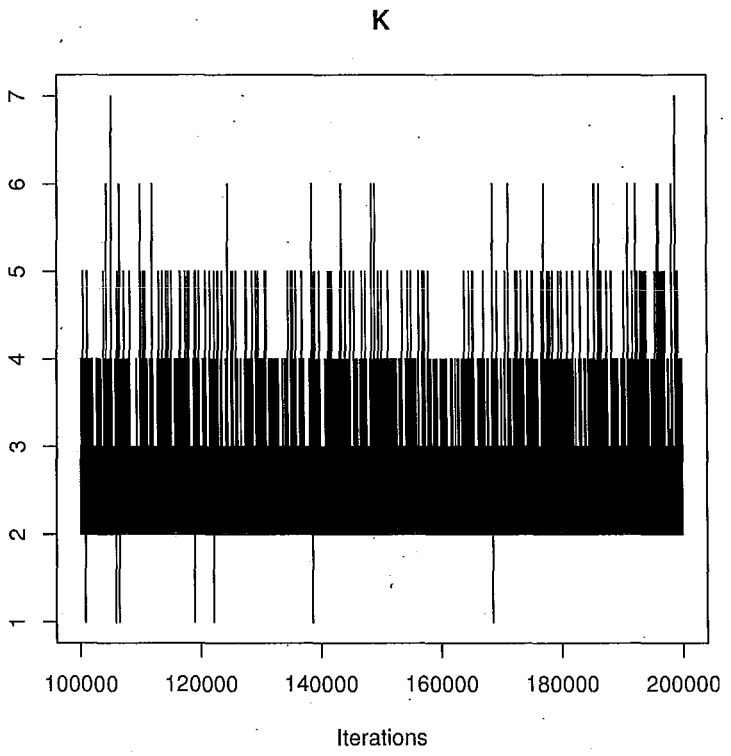

Figure 4.6: Model with a random number of mixture components. Trace plots for the number of mixture components k using the last 10,000 iterations.

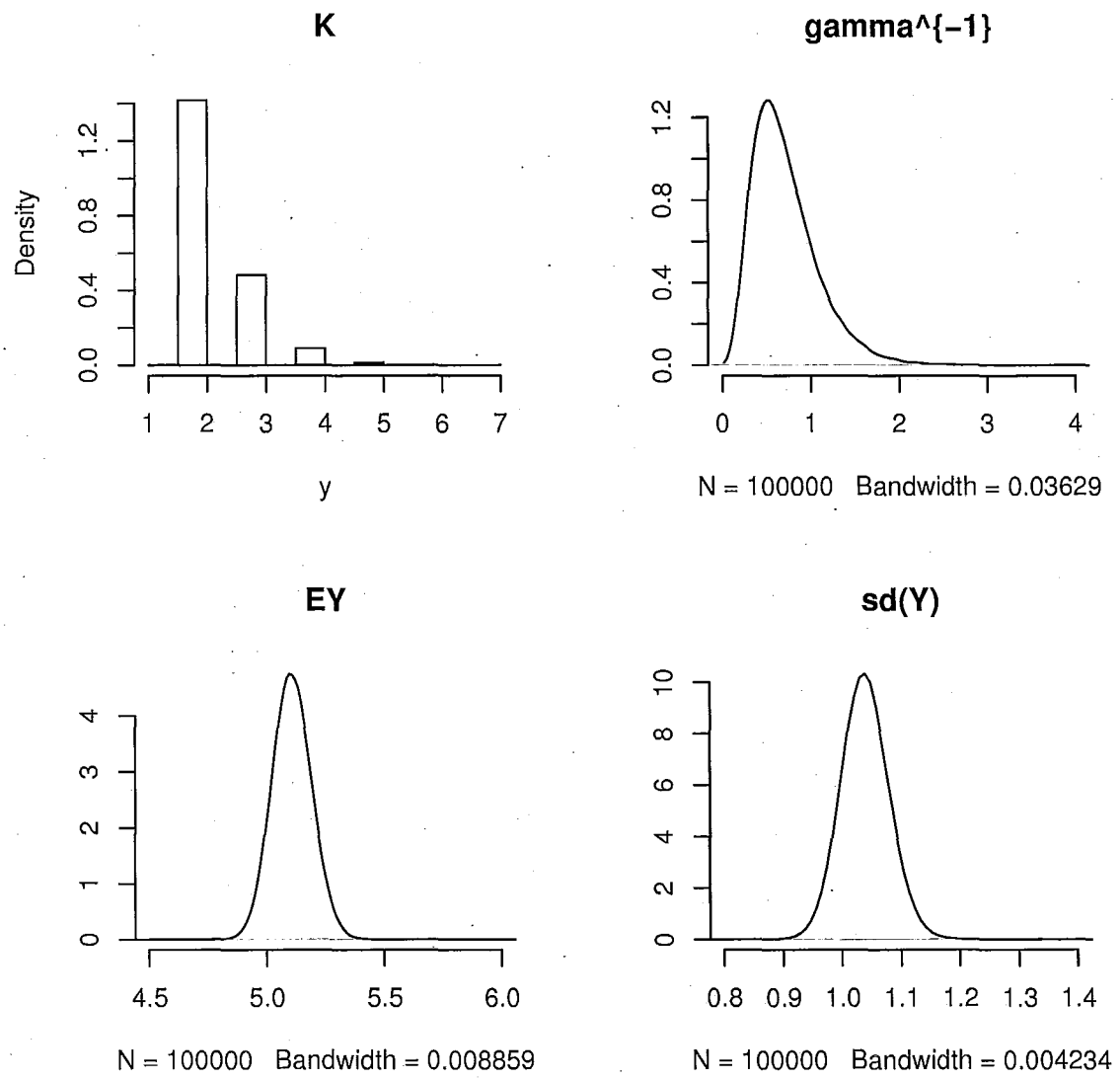

Figure 4.7: Model with a random number of mixture components. Posterior density estimates for selected parameters.

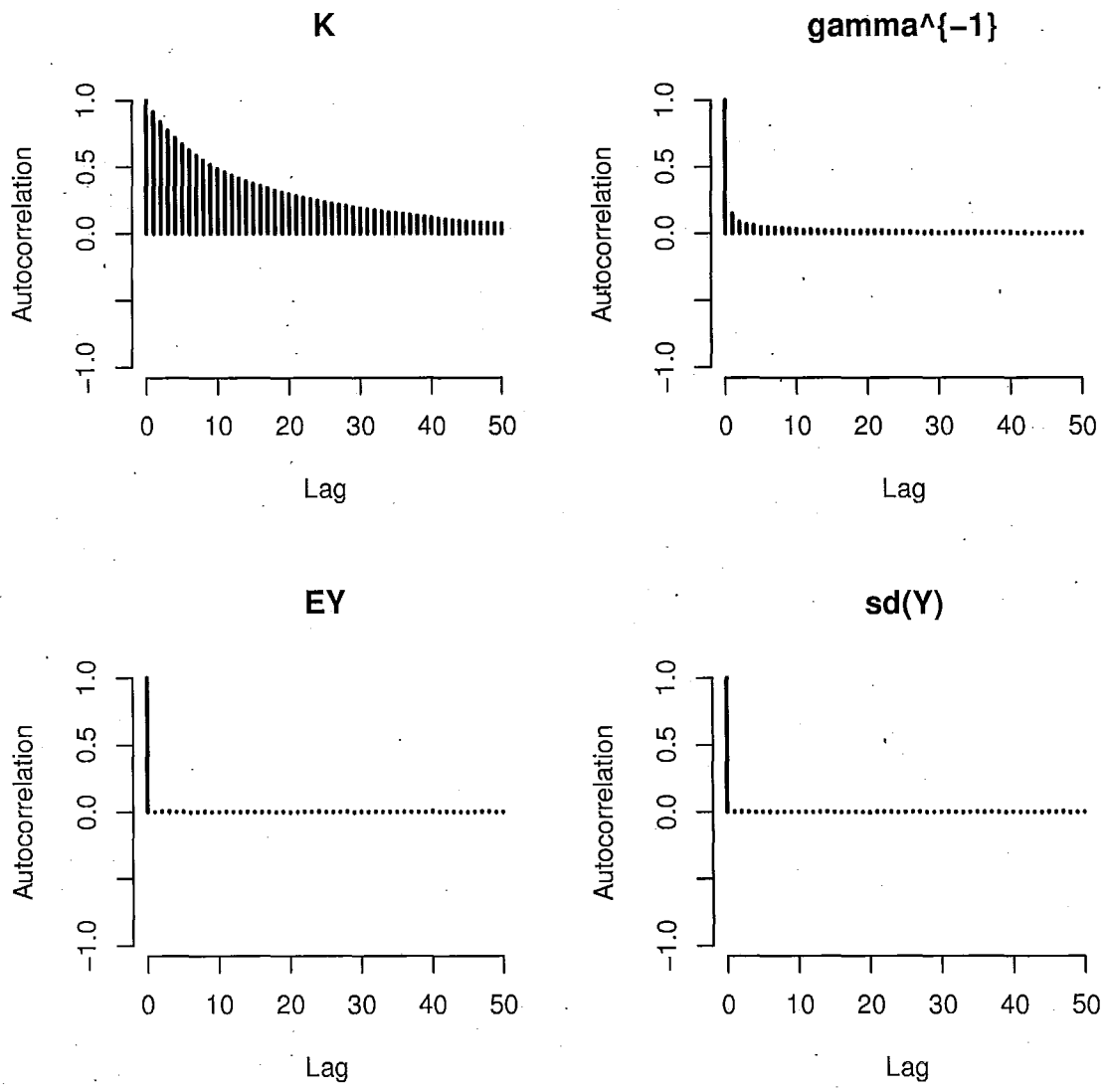

Figure 4.8: Model with a random number of mixture components. Autocorrelation plots for selected parameters.

4.3.5 Model with a Fixed Number of Components

We will fit a mixture model for $k = 1, ..., 6$, compare the deviance based quantities and predictive densities. For the prior hyper-parameters, we will use the same ones as in the model `RJModel2`.

```
# Specification of the prior hyper-parameters:
> FixPrior2 <- list(priorK="fixed", xi=median(mixdat),
+ D=(R/3)^2, g=0.2, h=5/R^2, zeta=4)

# Length of the MCMC simulation:
# 40,000 burn-in iterations
# 40,000 additional iterations for inference
> nMCMC2 <- c(burn=40000, keep=40000, thin=1, info=1000)

# Running MCMC simulation for k = 1, ..., 6, compute the predictive densities
# Then remove all chains from resulting objects to save some memory
> Keep<-c("iter","nMCMC2","dim","prior","init","RJMCMC",
+ "scale","state","freqK","propK","DIC","moves",
+ "pm.y","pm.z","pm.indDev","pred.dens","summ.y.Mean",
+ "summ.y.SDCorr","summ.z.Mean","summ.z.SDCorr")
> set.seed(12345)
> FixModel2<-list()
> PDensFix2<-list()
> for(k in 1:6){
+ cat(paste("K=",k,"\n------------------------------\n",sep=""))
+ PriorNow<-FixPrior2
+ PriorNow$Kmax<-k
+ FixModel2[[k]]<-NMixMCMC(y0=mixdat,prior=PriorNow, nMCMC=nMCMC2,
+ scale=list(shift=0,scale=1),PED=TRUE)
+
+ cat(paste("\nComputationofpred.densitiesstartedon",date(),
+ "\n",sep=""))
+ PDensFix2[[k]]<-list()
+ PDensFix2[[k]][[1]]<-NMixPredDensMarg(FixModel2[[k]][[1]],grid=ygrid)
+ PDensFix2[[k]][[2]]<-NMixPredDensMarg(FixModel2[[k]][[2]],grid=ygrid)
+ cat(paste("Computationofpred.densitiesfinishedon", date(),
+ "\n\n\n",sep=""))
+
+ FixModel2[[k]][[1]]<-FixModel2[[k]][[1]][Keep]
```

```
+ FixModel2[[k]][[2]]<-FixModel2[[k]][[2]][Keep]
+ class(FixModel2[[k]][[1]])<-class(FixModel2[[k]][[2]])<-"NMixMCMC"
+ }

# Summary of PED and DIC for the fitted models
> PED<-RJModel2$PED
> DIC<-list(Chain1=RJModel2[[1]]$DIC,Chain2=RJModel2[[2]]$DIC)
> for(k in 1:length(FixModel2)){
+ PED<-rbind(PED,FixModel2[[k]]$PED)
+ DIC[[1]]<-rbind(DIC[[1]],FixModel2[[k]][[1]]$DIC)
+ DIC[[2]]<-rbind(DIC[[2]],FixModel2[[k]][[2]]$DIC)
+ }
> rownames(PED)<-rownames(DIC[[1]])<-rownames(DIC[[2]])<-c("RJMCMC",
+ paste("K=",1:length(FixModel2),sep=""))

> print(PED)
        D.expect    p(opt)      PED   wp(opt)     wPED
RJMCMC 372.9857 14.968286 387.9540 18.002013 390.9877
K=1     453.5623  4.056124 457.6184  4.077197 457.6395
K=2     374.3781 11.779583 386.1577 11.879171 386.2573
K=3     367.7564 15.974221 383.7306 16.856490 384.6129
K=4     367.4465 19.867253 387.3138 22.315775 389.7623
K=5     366.4559 22.711205 389.1671 26.261377 392.7173
K=6     365.8764 25.254817 391.1312 28.297167 394.1735
> print(DIC)
$Chain1
             DIC       pD    D.bar D.in.bar
RJMCMC 380.9548  7.701561 373.2533 365.5517
K=1     454.9422  1.381621 453.5605 452.1789
K=2     380.2232  5.854943 374.3683 368.5133
K=3     375.8324  7.829397 368.0030 360.1736
K=4     376.9114  9.431267 367.4801 358.0489
K=5     376.7353 10.317638 366.4177 356.1001
K=6     377.0222 11.131525 365.8907 354.7592

$Chain2
             DIC       pD    D.bar D.in.bar
RJMCMC 380.7260  7.802109 372.9239 365.1218
K=1     454.9528  1.388840 453.5640 452.1751
K=2     380.2637  5.875719 374.3880 368.5123
K=3     375.0512  7.541466 367.5098 359.9683
K=4     376.6985  9.285583 367.4130 358.1274
K=5     376.8632 10.369001 366.4942 356.1252
K=6     377.0582 11.196126 365.8620 354.6659
```

```
# Plot the predictive densities for different values of k (Chain 1)
# See Figure 8 and 9
> hist(mixdat,prob=TRUE, breaks=30, xlab="Acidity Index",
+ ylab="Density", main="")
> for(k in 1:6){
+ lines(PDensFix2[[k]][[1]]$x$x1,PDensFix2[[k]][[1]]$dens[[1]])
+ }
> par(mar=c(3,2,2,1)+0.1)
> par(mfrow=c(3,2), bty="n")
> for(k in 1:6){
+ hist(mixdat, prob=TRUE, breaks=30, xlab="",ylab="",
+ main=paste("K=", k, sep=""))
+ lines(PDensFix2[[k]][[1]]$x$x1,PDensFix2[[k]][[1]]$dens[[1]],lwd=2)
+ }
```

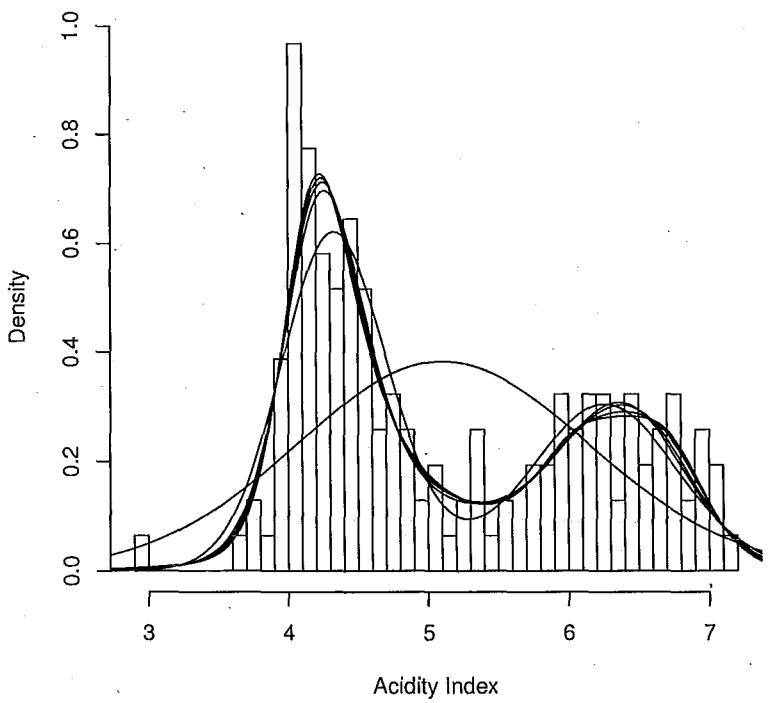

Figure 4.9: Predictive densities based on the models with a fixed number of mixture components from Chain 1.

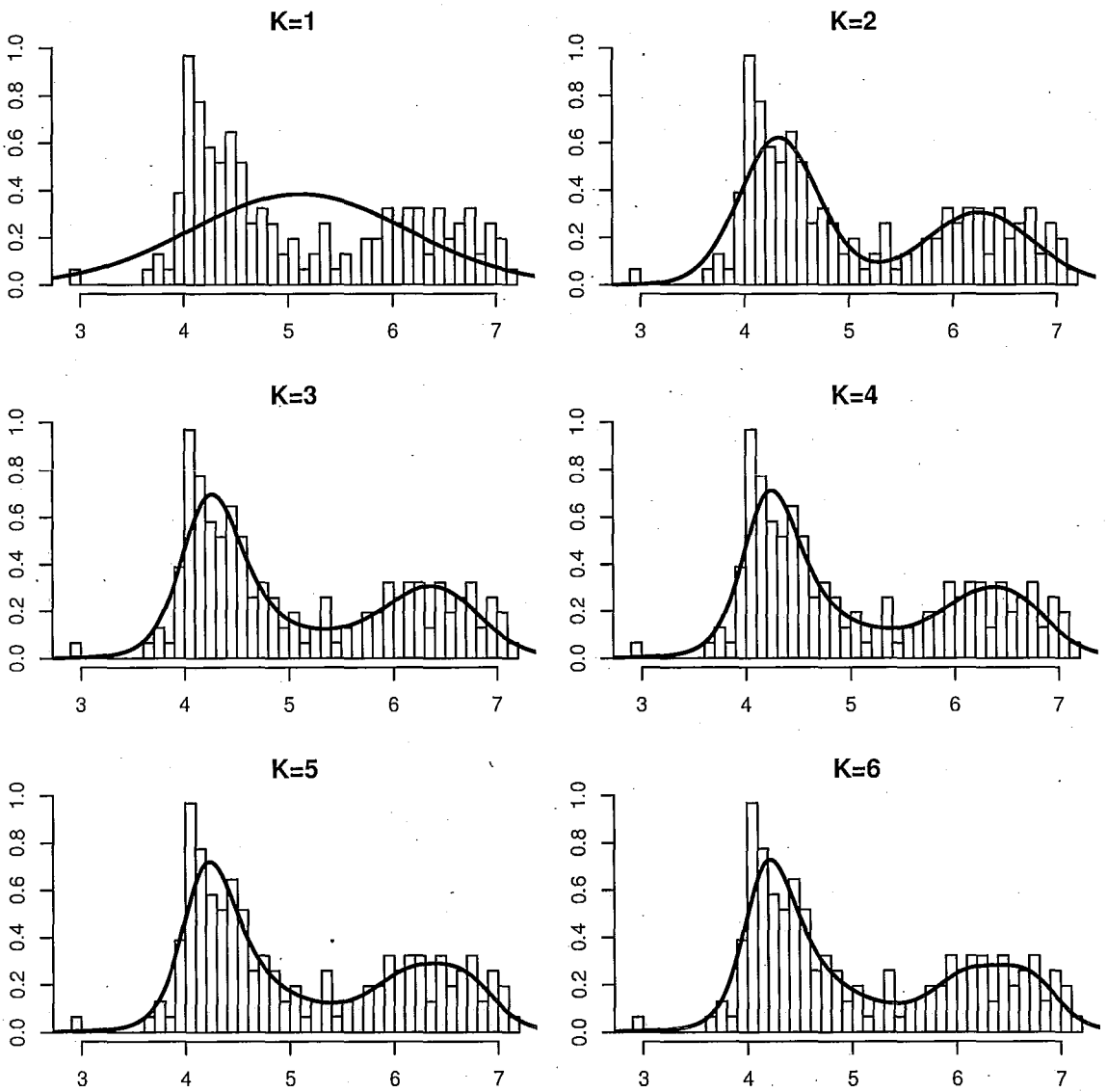

Figure 4.10: Predictive densities based on the models with a fixed number of mixture components from Chain 1.

Table 4.1: Penalized Expected Deviance and Related Quantities

K	D.expect	p(opt)	PED	wp(opt)	wPED
K=1	453.56	4.06	457.62	4.08	457.64
K=2	374.38	11.78	386.16	11.88	386.26
K=3	367.76	15.97	383.73	16.86	384.61
K=4	367.45	19.87	387.31	22.32	389.76
K=5	366.46	22.71	389.17	26.26	392.72
K=6	365.88	25.25	391.13	28.30	394.17

Table 4.2: DIC for Chain 1

K	DIC	p_D	D.bar	D.in.bar
K=1	454.94	1.38	453.56	452.18
K=2	380.22	5.85	374.37	368.51
K=3	375.83	7.83	368.00	360.17
K=4	376.91	9.43	367.48	358.05
K=5	376.74	10.32	366.42	356.10
K=6	377.02	11.13	365.89	354.76

Table 4.3: DIC for Chain 2

K	DIC	p_D	D.bar	D.in.bar
K=1	454.95	1.39	453.56	452.18
K=2	380.26	5.88	374.39	368.51
K=3	375.05	7.54	367.51	359.97
K=4	376.70	9.29	367.41	358.13
K=5	376.86	10.37	366.49	356.13
K=6	377.06	11.20	365.86	354.67

Table 4.1 shows the expected deviance $\bar{D}$, the optimism p_{opt}, and the penalized expected deviance $\bar{D} + p_{opt}$ for models with 1 - 6 components. The expected deviance was identical (within Markov chain error) for models with 3 - 6 components. If any additional component beyond 3 was added to the model, it would improve the model adequacy only by accounting for a small number of outliers, probably those with low acidity index. Such outliers are likely to be an artifact of the log transformation.

Table 4.2 and 4.3 show a similar pattern. Since we would choose the model with the smallest DIC, a 3-component mixture was preferred. Note that the predictive plots for $k = 2$ are very similar to those for $k = 3$. Therefore, a Normal mixture of either 2 or 3 components are acceptable.

These results indicate the existence of either two or three distinct groups of lakes with different distributions of acidity index. Group membership thus provides a crude indicator of risk for loss of biological resources. To obtain a better prediction on which lakes will be at risk, characteristics such as lake type (seepage or drainage), lake area, and geochemistry should be considered.

4.4 Discussion

4.4.1 Sensitivity to Prior Distribution of Means

For the prior on the means μ_j, which are drawn independently from the Normal distribution $N(\xi, \kappa^{-1})$, Richardson and Green (1997) suggested that ξ was set equal to the mid-range and

precision κ such that $\kappa^{-1/2}$ was equal to the range R, claiming that these sensible weakly informative priors place no constraint on the location of the μ_j.

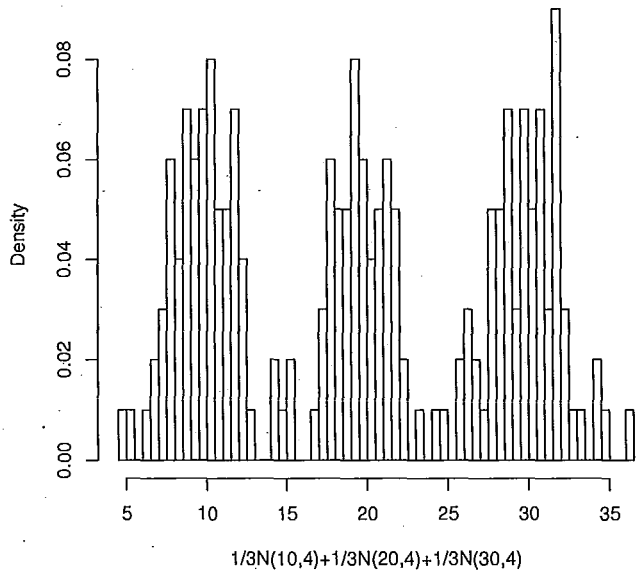

Figure 4.11: Histogram of 200 points simulated from the model $1/3N(10,4) + 1/3N(20,4) + 1/3N(30,4)$.

We present a simulation study to explore the relationship between the prior information on the location of the means and the number of components k. A histogram is shown in 4.11 for the simulated data set of size $n = 200$ from a 3-component Normal mixture $1/3N(10,4) + 1/3N(20,4) + 1/3N(30,4)$. Using the same hierarchical structures as in §4.1.1 but varying κ^{-1}, we find that at first reducing κ^{-1} tends to favor a higher number of components. This can be interpreted as the result of defining a prior which is increasingly more permissive of components with close means. However, as κ^{-1} is further reduced, the number of components start to decrease; this is called a shrinkage effect. It prohibits components with means located

towards the extremes of the range.

We illustrate this in Table 4.4. As the values of $\kappa^{-1/2}$ decrease from R to $R/50$, the number of components with the highest posterior probability first increases to reach a peak value of $k = 13$ for $\kappa^{-1/2}$ between $R/3$ and $R/4$, and then decreases.

Table 4.4: Influence of prior $N(\xi, \kappa^{-1})$ for μ on the posterior of k

$\kappa^{-1/2}$	Range of k with $p(k\|y) \geq 0.05$	Range of k with $p(k\|y) \geq 0.001$	k with highest $p(k)$
R	5-12	2-17	8
R/2	7-15	3-22	11
R/3	9-17	4-24	13
R/4	9-17	4-25	13
R/5	8-16	4-23	11
R/6	8-16	4-24	11
R/8	6-13	4-20	10
R/10	5-11	2-17	8
R/20	4-10	3-15	7
R/50	2-7	1-12	3

4.4.2 Performance of MCMC Sampler

An essential element of the performance of the reversible jump MCMC sampler is its ability to move between different values of k. Therefore, the acceptance rate of a chain acts as a barometer of the quality of the sampling process. If the acceptance rate is too low, then the chain will not be able to move freely and it will take a long time to sample the probability density functions fully. However, a high acceptance rate may also be a problem, since the chain will explore only a small portion of the parameter space. From the convergence point of view, the too high acceptance rate is a worse situation. A low acceptance rate is undesirable

only from the computing-time point of view.

In practice, one should try to tune the chain until the acceptance rate is acceptable. This is unfortunately not easily to achieve using the package `mixAK`, especially for a complex algorithm like the RJMCMC. A plot of the changes in k against the number of iterations was presented in Figure 4.6. It shows that the MCMC algorithm mixes reasonably well over k, excursions into high values being short lived. The proportions of accepted split-combine moves and birth-death moves are both less than 10% in our case, but they are still adequate to obtain good statistics.

Working with other data sets, we find that the acceptance rates for reversible jump moves are highest for those with small sample sizes and multiple modes. Thus it is not surprising that in the example of the acidity data, we have low acceptance rates for the reversible jump moves.

Chapter 5

Conclusion and Suggestion for Future Work

In this thesis, we reviewed some of the recent published work in Bayesian inference for finite mixture models. For a mixture with known number of components, we illustrated the implementation of Gibbs sampler and applied it to Normal mixture models. When the number of components is unknown, we used the reversible jump MCMC algorithm to perform a fully Bayesian analysis. Application to the acidity data was presented and the choice of k was based on criterion such as the DIC and PED. Issues in Bayesian framework including trapping states, choices of priors, label switching, and convergence diagnostics were discussed and illustrated with examples.

Adopting a Bayesian approach for finite mixture models has many advantages. First, including a proper prior may introduce a smoothing effect on the mixture likelihood function

and reduce the risk of obtaining spurious modes. This was shown to be useful in particular for mixtures of Normal distributions, as in the Kiefer-Wolfowitz example. Secondly, Bayesian estimation does not rely on asymptotic normality. It provides valid inference in cases where regularity conditions are violated, such as when the sample size is small and the component weights are small. Finally, with the fast development and straightforward implementation of many MCMC methods, Bayesian estimation of finite mixture models has become possible.

While being a fairly natural algorithm, the Gibbs sampler can easily fall victim to label switching. We simply do not know how to estimate the parameters when this happens. To avoid label switching, we suggest that the mixture components should be arranged in order of non-decreasing means $\mu_1 \leq \mu_2 \leq \cdots \leq \mu_k$ and increasing variances $\sigma_1 < \sigma_2 < \cdots < \sigma_k$ when the means are equal. If this fails, we should choose more informative priors instead of data-dependent ones on the parameters. As shown in the fishery example, choosing hyper-parameters that reduce the simulated variances in a Normal mixture can be effective in removing label switching. Note however, our goal is to make "minimal" assumptions on the data, and modifications of hyper-parameters can be rather influential and delicate. It is usually difficult to make sensible choices of them, unless something else is known about the model.

The techniques we have mentioned for convergence diagnostics of an MCMC sampler in this thesis are mostly empirical. We suggest that such diagnostics should always be used, however, they can only find obvious problems. It is almost impossible for us to control the flow of a Markov chain or assess its convergence behavior based on a few thousand or million realizations of this chain.

It is important to start the MCMC algorithms with "good" initial values. Despite the formal irreducibility of the Gibbs sampler, escaping from trapping states usually requires an enormous number of iterations. In general, if no good initial values are available, we should try to improve the sampler or simply use a better one. The longer we run the chain, the better.

Since we only presented the implementation of the Gibbs sampler for mixtures of Normal distributions in this thesis, future work should include Gibbs sampling for other mixtures that are also commonly used, such as the Poisson, Gamma, Weibull, and Lognormal.

Several recent computer packages for Bayesian mixture models deserve exploring, including `bayesm` by Rossi and McCulloch (2008), an `R` package that provides the implementation of multivariate Normal mixtures; `AdMit` by Ardia *et al.* (2009), an `R` package for fitting adaptive mixtures of Student-t distributions to a target density through its kernel function; a `MATLAB` package `bayesf` by Frühwirth-Schnatter (2008) and its `R` version `bayesmix` by Grüen (2010), both designed to fit finite mixture models using a Bayesian approach based on MCMC methods.

Bibliography

[1] Akaike, H. (1974). A new look at statistical model identification. *IEEE Transactions on Automatic Control* **19**: 716-723.

[2] Allenby, G. M., Arora, N. and Ginter, J. L. (1998). On the heterogeneity of demand. *Journal of Marketing Research* **35**: 384-389.

[3] Ardia, D., Hoogerheide, L. F. and Van Dijk H. K. (2009). The 'AdMit' Package: Adaptive Mixture of Student-t Distributions. *R package version 1-01.03.* `http://www.jstatsoft.org/v29/i03/`

[4] Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics* **41**: 164-171.

[5] Bensmail, H., Celeux, G., Raftery, A. E. and Robert, C. P. (1997). Inference in model-based cluster analysis. *Statistics and Computing* **7**: 1-10.

[6] Berger, J. and Pericchi, L. (1996). The intrinsic Bayes factor for model selection and prediction. *Journal of the American Statistical Association* **91**: 109-122.

[7] Besag, J. and Clifford, P. (1989). Generalized Monte Carlo significance tests. *Biometrika* **76**: 633-642.

[8] Böhning, D. (2000). *Computer Assisted Analysis of Mixtures and Applications.* London: Chapman & Hall.

[9] Böhning, D. and Seidel, W. (2003). Editorial: recent developments in mixture models, *Computational Statistics & Data Analysis* **41**: 349-357.

[10] Böhning, D., Seidel, W., Alfó, M., Garel, B., Patilea, V., and Walther, G. (2007). Editorial: advances in mixture models, *Computational Statistics & Data Analysis* **51**: 5205-5210.

[11] Broët, P., Richardson, S., and Radvanyi, F. (2002) Bayesian hierarchical model for identifying changes in gene expression from microarray experiments. *Journal of Computational Biology* **9(4)**: 671-683.

[12] Brooks, S. and Roberts, G. (1998). Assessing convergence of Markov chain Monte Carlo algorithms. *Statistics and Computing* **8**: 319-335.

[13] Cassie, R. M. (1954). Some uses of probability paper in the analysis of size frequency distributions. *Australian Journal of Marine & Freshwater Research* **5**: 513-522.

[14] Carlin, B. P. and Chib, S. (1995). Bayesian model choice via Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society, Series B* **57**: 473-484.

[15] Casella, G. and Berger, R. (2001). *Statistical Inference, 2nd edition.* Belmont, CA: Wadsworth.

[16] Celeux, G. (1998). Bayesian inference for mixture: The label switching problem. In: Green, P. J. and Rayne, R. (eds.) *COMPSTAT 1998*, pp. 227-232. Heidelberg: Physica.

[17] Celeux, G., Forbes, F., Robert, C. and Titterington, D. (2006a). Deviance information criteria for missing data models. *Bayesian Analysis* 1: 651-706.

[18] Celeux, G., Forbes, F., Robert, C. and Titterington, D. (2006b). Rejoinder to discussion of deviance information criteria for missing data models. *Bayesian Analysis* **1**: 701-706.

[19] Celeux, G., Hurn, M. and Robert, C. P. (2000). Computational and inferential difficulties with mixture posterior distributions. *Journal of the American Statistical Association* **95**: 957-970.

[20] Chen, H., Chen, J. and Kalbfleisch, J. D. (2002). Testing for a finite mixture model with two components. *Statistics and Actuarial Science Technical Report* **2001-02**, University of Waterloo.

[21] Chen, J., Tan, X. and Zhang, R. (2008). Inference for normal mixtures in mean and variance. *Statistica Sincia* **18**: 443-465.

[22] Chib, S. (1995). Marginal likelihood from the Gibbs output. *Journal of the American Statistical Association* **90**: 1313-1321.

[23] Cowles, M. and Carlin, B. (1996). Markov chain Monte Carlo convergence diagnostics: a comparative study. *Journal of the American Statistical Association* **91**: 883-904.

[24] Craigmile, P. F. and Titterington, D. M. (1998). Parameter estimation for finite mixtures of uniform distributions. *Communications in Statistics - Theory and Methods* **26**: 1981-1995.

[25] Crawford, S. L. (1994). An application of the Laplace method to finite mixture distributions. *Journal of the American Statistical Association* **89**: 259-267.

[26] Crawford, S. L., DeGroot, M. H., Kadane, J. B., and Small, M. J. (1994). Modeling lake chemistry distributions: Approximate Bayesian methods for estimating a finite mixture model. *Technometrics* **34**: 441-453.

[27] Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977). Maximum likelihood from incomplete data via EM algorithm (with discussion). *Journal of the Royal Statistical Society, Series B* **39**: 1-38.

[28] Diebolt, J. and Robert, C. P. (1990) Bayesian estimation of finite mixture distributions: part II, Sampling implementation. *Technical Report 111.* Laboratoire de Statistique Théorique et Appliquée, Université Paris VI, Paris.

[29] Diebolt, J. and Robert, C. P. (1994). Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society, Series B* **56**: 363-375.

[30] Efron, B. (1983). Estimating the error rate of a prediction rule: improvements on cross-validation. *Journal of the American Statistical Association* **78**: 316-331.

[31] Everitt, B. S. and Hand, D. J. (1981). *Finite Mixture Distributions.* London: Chapman & Hall.

[32] Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* **97**: 611-631.

[33] Frühwirth-Schnatter, S. (2006). *Finite Mixture and Markov Switching Models.* New York/Berlin/Heidelberg: Springer.

[34] Frühwirth-Schnatter, S. (2008). *Finite Mixture and Markov Switching Models: Implementation in MATLAB using the package bayesf Version 2.0.* Berlin/Heidelberg/New York/Hong Kong/London/Milan/Paris/Tokyo: Springer.

[35] Furman, W. D. and Lindsay, B. G. (1994a). Testing for the number of components in a mixture of normal distributions using moment estimators. *Computational Statistics and Data Analysis* **17**: 473-492.

[36] Furman, W. D. and Lindsay, B. G. (1994b). Measuring the effectiveness of moment estimators as starting values in maximizing mixture likelihoods. *Computational Statistics and Data Analysis* **17**: 493-507.

[37] Geisser, S. and Eddy, W. (1979). A predictive approach to model selection. *Journal of the American Statistical Association* **74**: 153-160.

[38] Gelfand, A. E. and Smith A. F. M. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association* **85**: 398-409.

[39] Gelman, A., Carlin, J., Stern, H. and Rubin, D. (2002). *Bayesian Data Analysis, 2nd edition.* Boca Raton, FL: Chapman & Hall/CRC.

[40] Gelman, A. and Rubin, D. B. (1992). Inferences from iterative simulation using multiple sequences (with discussion). *Statistical Science* **7**: 457-511.

[41] Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transaction on Pattern Analysis and Machine Intelligence* **6**: 721-741.

[42] Geweke, J. (1992). Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. In: Bernardo, J. M., Berger, J. O., Dawid, A. P. and Smith, A. F. M. (eds.) *Bayesian Statistics 4*. Oxford University Press.

[43] Geyer, C. J. (1992). Practical Markov chain Monte Carlo (with discussion). *Statistical Science* **7**: 473-511.

[44] Green, P. J. and Richardson, S. (2002). Hidden Markov models and disease mapping. *Journal of the American Statistical Association* **97**: 1-16.

[45] Grüen, B. (2010). bayesmix: Bayesian Mixture Models with JAGS. *R package version 0.7-0.* `http://CRAN.R-project.org/package=bayesmix`

[46] Hamilton, J. D. (1989). A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica* **57**: 357-384.

[47] Hartigan, J. A. (1985). A failure of likelihood asymptotics for normal mixtures. In: LeCam, L. and Olshen, R. A. (eds.) *Proceedings of the Berk Conference in Honor of J. Neyman and J. Kiefer* **2**: 807-810.

[48] Hasselblad, V. (1966). Estimation of parameters for a mixture of normal distributions. *Technometrics* **8**: 431-444.

[49] Hastings, W. (1970). Monte Carlo sampling methods using Markov chains and their application. *Biometrika* **57**: 97-109.

[50] Hathaway, R. J. (1985). A constrained formulation of maximun-likelihood estimation for normal mixture distributions. *The Annals of Statistics* **13**: 795-800.

[51] Heidelberger, P. and Welch, P. (1983). A spectral method for confidence interval generation and run length control in simulations. *Communications of the Association for Computing Machinery* **24**: 233-245.

[52] Jasra, A., Holmes, C. C. and Stephens, D. A. (2005). Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling. *Statistical Science* **20(1)**: 50-67.

[53] Jedidi, K., Jagpal, H. S. and DeSarbo, W. S. (1997) Finite-mixture structural equation models for response-based segmentation and unobserved heterogeneity. *Marketing Science* **16(1)**: 39-59.

[54] Kass, R. E. and Raftery, A. E. (1995). Bayes factors. *Journal of the American Statistical Association* **90**: 773-795.

[55] Kaufman, S., Frühwirth-Schnatter, S. (2002). Bayesian analysis of switching ARCH models. *Journal of Time Series Analysis* **23**: 425-458.

[56] Kiefer, J. and Wolfowitz, J. (1956). Consistency of the maximum likelihood estimates in the presence of infinitely many incidental paramters. *Annals of Mathematical Statistics* **27**: 887-906.

[57] Komárek, A. (2009). A new R package for Bayesian estimation of multivariate normal mixtures allowing for selection of number of components and interval-censored data. *Computational Statistics and Data Analysis* **53**: 3932-3947.

[58] Kullback, S. and Leibler, R. A. (1951). On the information and sufficiency. *Annals of Mathematical Statistics* **22**: 79-86.

[59] Lamoureux, C. G., Lastsrapes, W. D. (1994). Endogenous trading volume and momentum in stock return volatility. *Journal of Business & Economic Statistics* **12**: 253-260.

[60] Lavine, M. and M. West (1992). A Bayesian method for classification and discrimination. *The Canadian Journal of Statistics* **20**: 451-461.

[61] LeSage, J. P. (1992), A comparison of time-varying parameter and multiprocess mixture models in the case of money-supply announcements. *Journal of Business & Economic Statistics* **10**: 201-211.

[62] Lindsay, B. G. (1995). *Mixture Models: Theory, Geometry, and Applications.* Hayward: Institute of Mathematical Statistics.

[63] Lindsay, B. G. and Basak, P. (1993). Multivariate normal mixtures: a fast, consistent method of moments. *Journal of the American Statistical Association* **86**: 96-107.

[64] Lunn, D., Thomas, A., Best, N., and Spiegelhalter, D. (2000). WinBUGS - a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and Computing* **10**: 325-337.

[65] Macdonald, P. D. M. and Du, J. (2010). mixdist: Finite Mixture Distribution Models. *R package version 0.5-3.* `http://CRAN.R-project.org/package=mixdist`

[66] Marin, J. and Mengersen, K. L. and Robert, C. (2005) Bayesian modelling and inference on mixtures of distributions. In: Dey, D. and Rao, C. R. (eds.) *Handbook of Statistics - 25.* Elsevier.

[67] McLachlan, G. J. and Basford, K. E. (1988). *Mixture Models: Inference and Applications to Clustering.* New York/Basel: Marcel Dekker.

[68] McLachlan, G. J. and Krishnan, T. (1997). *The EM algorithm and Extensions.* New York: Wiley.

[69] McLachlan, G. J. and Peel, D. (2000). *Finite Mixture Models.* New York: Wiley.

[70] Mengersen, K. and Robert, C. P. (1996) Testing for mixtures: a Bayesian entropy approach. In: Berger, J. O., Bernardo, J. M., Dawid, A. P., Lindley, D. V. and Smith A. F. M. (eds.) *Bayesian Statistics 5*, pp. 255-276. London: Oxford University Press.

[71] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics* **21(6)**: 1087-1092.

[72] Moreno, E. and Liseo, B. (2003). A default Bayesian test for the number of components in a mixture. *Journal of Statistical Planning and Inference* **111(1)**: 129-142.

[73] Newcomb, S. (1886). A generalized theory of the combination of observations so as to obtain the best result. *American Journal of Mathematics* **8**: 343-366.

[74] O'Hagan, A. (1995). Fractional Bayes factors for model comparison (with discussion). *Journal of the Royal Statistical Society, Series B* **57**: 99-138.

[75] Pearson, K. (1894). Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society Of London A* **185**: 71-110.

[76] Pérez, J. M. and Berger, J. (2002). Expected posterior prior distributions for model selection. *Biometrika* **89**: 491-511.

[77] Peskun, P. (1973). Optimum Monte Carlo sampling using Markov chains. *Biometrika* **60**: 607-612.

[78] Phillips, D. B. and Smith, A. F. M. (1996). Bayesian model comparison via jump diffusions. In Gilks, W., Richardson, S. and Spiegelhalter, D. J. (eds.) *Markov Chain Monte Carlo in Practice*, pp. 215-239. London: Chapman & Hall.

[79] Plummer, M. (2002). Discussion of the paper by Spiegelhalter *et al. Journal of the Royal Statistical Society, Series B* **64**: 620.

[80] Plummer, M. (2008). Penalized loss functions for Bayesian model comparison. *Biostatistics* **9(3)**: 523-539.

[81] Plummer, M., Best, N., Cowles, K., and Vines, K. (2006). CODA: convergence diagnosis and output analysis for MCMC. *R News*, **6(1)**: 7-11.

[82] Raftery, A. E. (1996b). Hypothesis testing and model selection. In Gilks, W. R., Richardson, S. and Spiegelhalter D. J. (eds.) *Markov Chain Monte Carlo in Practice*, pp. 163-188. London: Chapman & Hall.

[83] Raftery, A. E. and Lewis, S. (1992b). Comment: One long run with diagnostics: Implementation strategies for Markov Chain Monte Carlo. *Statistical Science* **7**: 493-497.

[84] Rao, C. (1948). The utilization of multiple measurements in problems of biological classification. *Journal of the Royal Statistical Society, Series B* **10**: 159-203.

[85] Richardson, S. and Green, P. J. (1997). On Bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society, Series B* **59**: 731-792.

[86] Robert, C. P., Ryden, T., Titterington, D. M. (2000). Bayesian inference in hidden Markov models through the reversible jump Markov chain Monte Carlo method. *Journal of the Royal Statistical Society: Series B* **62**: 57-75.

[87] Roeder, K. (1990). Density estimation with confidence sets exemplified by superclusters and voids in the galaxies. *Journal of the American Statistical Association* **85**: 617-624.

[88] Roeder, K. and Wasserman, L. (1997). Practical Bayesian density estimation using mixtures of normals. *Journal of the American Statistical Association* **92**: 894-902.

[89] Rossi, P. and McCulloch, R. (2008). bayesm: Bayesian Inference for Marketing/Microeconometrics. *R package version 2.2-2.* `http://faculty.chicagogsb.edu/peter.rossi/research/bsm.html`

[90] Schlattmann, P. (2009). *Medical Applications of Finite Mixture Models.* Verlag/Berlin/Heidelberg: Springer.

[91] Schlattmann, P. and Böhning, D. (1993). Mixture models and disease mapping. *Statistics in Medicine* **12**: 943-1950.

[92] Sperrin, M., Jaki, T. and Wit, E. (2010). Probabilistic relabelling strategies for the label switching problem in Bayesian mixture models. *Statistics and Computing* **20(3)**: 357-366.

[93] Stephens, M. (2000a). Bayesian analysis of mixture models with an unknown number of components - An alternative to reversible jump methods. *The Annals of Statistics* **28**: 40-74.

[94] Stephens, M. (2000b). Dealing with label switching in mixture models. *Journal of the Royal Statistical Society, Series B* **62**: 795-809.

[95] Tan X., Chen J. and Zhang R. (2006). Consistency of the constrained maximum likelihood estimator in finite normal mixture models. Submitted.

[96] Tanner, M. Y. and Wong, W. H. (1987) The calculation of poserior distribution by data augmentation. *Journal of the American Statistical Association* **67**: 702-708.

[97] Teicher, H. (1960). On the mixture of distributions. *Annals of Mathematical Statistics* **31**: 55-73.

[98] Teicher, H. (1963). Identifiability of finite mixtures. *Annals of Mathematical Statistics* **34**: 1265-1269.

[99] Tierney, L. (1994). Markov chains for exploring posterior distributions *Annals of Statistics* **22(4)**: 1701-1728.

[100] Titterington, D. M., Smith, A. F. M. and Markov, U. E. (1985). *Statistical Analysis of Finite Mixture Distributions*. New York: Wiley.

[101] Verdinelli, I. and L. Wasserman (1991). Bayesian analysis of outlier problems using the Gibbs sampler. *Statistics and Computing* **1**: 105-117.

[102] Wasserman, L. (2000). Asymptotic inference for mixture models using datadependent priors. *Journal of the Royal Statistical Society, Series B* **62**: 159-180.

[103] Withers, C. S. (1996). Moment estimates for mixtures of several distributions its different means of scales. *Communications in Statistics - Theory and Methods* **25**: 1799-1824.

[104] Wolfowitz, J. (1957). The Minimum Distance Method. *The Annals of Mathematical Statistics* **28(1)**: 75-88.

[105] Yao, W. and Lindsay, B. (2009). Bayesian mixture labelling by posterior density. *Journal of American Statistical Association* **104**: 758-767.

[106] Yakowitz, S. J. and Spragins, J. D. (1968). On the identifiability of finite mixtures. *Annals of Mathematical Statistics* **39**: 209-214.