Communications of the Korean Statistical Society
2012, Vol. 19, No. 1, 135–145

DOI: http://dx.doi.org/10.5351/CKSS.2012.19.1.135

An EM Algorithm for a Doubly Smoothed MLE in Normal Mixture Models

Byungtae Seo[1,a]

[a]Department of Statistics, Sungkyunkwan University

Abstract

It is well known that the maximum likelihood estimator(MLE) in normal mixture models with unequal variances does not fall in the interior of the parameter space. Recently, a doubly smoothed maximum likelihood estimator(DS-MLE) (Seo and Lindsay, 2010) was proposed as a general alternative to the ordinary maximum likelihood estimator. Although this method gives a natural modification to the ordinary MLE, its computation is cumbersome due to intractable integrations. In this paper, we derive an EM algorithm for the DS-MLE under normal mixture models and propose a fast computational tool using a local quadratic approximation. The accuracy and speed of the proposed method is then presented via some numerical studies.

Keywords: EM algorithm, normal mixture, doubly-smoothed MLE, quadratic approximation.

1. Introduction

Although normal mixture models play a central role in the mixture literature, it is well known that the maximum likelihood approach fails to produce a consistent estimator due to an unbounded likelihood. This type of failure is also common when we use a mixture of location-scale family of distributions. To resolve this problem, a constrained maximum likelihood estimator(MLE) (Hathaway, 1985; Tanaka and Takemura, 2006) uses a constraint on the scale parameters to compactify the parameter space. A penalized MLE proposed by Ciuperca *et al.* (2003) and Chen *et al.* (2008) adds some penalty functions to the ordinary likelihood so that the likelihood does not explode when one of the scale parameters goes to zero. The penalized MLE can also be considered as a Bayesian estimator (Fraley and Raftery, 2007) with an inverse Gamma or Wishart prior for scale parameters. The penalized MLE and constrained MLE can be obtained using slight modification of the EM algorithm for normal mixture models (Hathaway, 1986; Ingrassia and Rocci, 2007; Ciuperca *et al.*, 2003; Chen *et al.*, 2008).

As an alternative, Seo and Lindsay (2010) suggested the doubly-smoothed MLE(DS-MLE) which uses smoothing techniques for both the model and data with a fixed kernel and bandwidth. In some sense, the DS-MLE is considered a smoothed MLE because the DS-MLE tends to the ordinary MLE as the bandwidth goes to zero. However, unlike other estimators based on smoothing techniques, the DS-MLE is generally consistent even with a fixed bandwidth. Hence the choice of a bandwidth is less sensitive to the quality of estimators than other smoothing based estimators. If we use the DS-MLE to mixture models, the degeneracy problem is naturally removed, because the smoothed mixture likelihood is bounded for any fixed positive bandwidth.

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (2011-0014607).

[1] Assistant Professor, Department of Statistics, Sungkyunkwan University, 53 Myeogryun-dong 3-ga, Chongno-gu, Seoul 110-745, Korea. E-mail: seobt@skku.edu

Although the DS-MLE has many good theoretical properties, its computation requires several numerical integrations. Seo and Lindsay (2010) proposed a simple computational strategy using the Monte Carlo method and applied to the EM algorithm for normal mixtures. However, the EM algorithm under their framework has not been studied thoroughly and their method could be inaccurate in addition, it requires a large amount of computing effort.

In this paper, we derive an EM algorithm to obtain the DS-MLE in normal mixture models and give a fast computing method using a quadratic approximation. Although we focus on the EM algorithm of the DS-MLE for normal mixture models, our work can be easily extended to the DS-MLE for other types of mixture models. This paper is organized as follows: In Section 2, we give a brief review for the unbounded mixture likelihood and the DS-MLE. Then we derive an EM algorithm for the DS-MLE in normal mixtures in Section 3, and propose a fast computing method in Section 4. Some numerical examples to show the accuracy and the speed of the proposed method is given in Section 5. We then give brief concluding remarks in Section 6.

2. Unbounded Likelihood and DS-MLE

In this section, we briefly review the unboundedness of the mixture likelihood and the DS-MLE for normal mixture models. To explain the unboundedness of the mixture likelihood, let us consider the J-component univariate normal mixture model

$$f(x;\theta) = \sum_{j=1}^{J} p_j N\left(x;\mu_j,\sigma_j^2\right), \tag{2.1}$$

where $\theta = (\mu_1,\ldots,\mu_J,\sigma_1^2,\ldots,\sigma_J^2,p_1,\ldots,p_J) \in \mathbb{R}^J \times \mathbb{R}^{+J} \times \mathbb{R}(0,1)^J$ with the constraint $\sum_{j=1}^{J} p_j = 1$, and $N(x;\mu,\sigma^2)$ stands for the normal density with mean μ and variance σ^2. Suppose $X_i,\ldots,X_n$ is a random sample from (2.1), then it is easy to show that the likelihood of θ is unbounded (Kiefer and Wolfowitz, 1956). For a simple example, let us consider the likelihood function for $J = 2$

$$L(\theta) = \prod_{i=1}^{n}\left[\frac{p_1}{\sqrt{2\pi\sigma_1^2}}\exp\left(-\frac{(x_i-\mu_1)^2}{2\sigma_1^2}\right) + \frac{p_2}{\sqrt{2\pi\sigma_2^2}}\exp\left(-\frac{(x_i-\mu_2)^2}{2\sigma_2^2}\right)\right]. \tag{2.2}$$

In $L(\theta)$, if we fix $\mu_1 = x_1$ (or any observation) and let σ_1^2 go to zero, then (2.2) diverges regardless of a given sample. Indeed, we can observe many infinite spikes at $\sigma_j^2 = 0$, $j = 1,2$. Consequently, the ordinary MLE always occurs on the boundary of the parameter space and this degenerate MLE is neither meaningful nor consistent.

As a tool to regularize this likelihood, Seo and Lindsay (2010) proposed the doubly-smoothed(DS) log likelihood that smooths both the model and data. They constructed two smoothed densities based on a given model and data. The smoothed model density is given by

$$f_h^*(t;\theta) = \int f(x;\theta)K_h(x,t)dx, \tag{2.3}$$

where $K_h(x,t)$ is a kernel density with a bandwidth h. The smoothed empirical density is then constructed as

$$\hat{f}_h^*(t) = \frac{1}{n}\sum_{i=1}^{n} K_h(x_i,t), \tag{2.4}$$

which is known as a kernel density estimator based on a given sample $X_1, \ldots, X_n$. If we define a new random variable $T = X + \epsilon$, then T has a density $f_h^*(t; \theta)$, where the densities of X and ϵ are $f(x; \theta)$ and $K_h(0, t)$, respectively. The smoothed empirical density $\hat{f}_h^*(t)$ is then considered as a data dependent smoothed density. Note that we use the same kernel and bandwidth in (2.3) and (2.4). This implies that we view the model and data after we add a measurement error.

Now, the DS-MLE of θ is defined as the minimizer of the Kullback-Leibler divergence between two smoothed densities $f_h^*(t; \theta)$ and $\hat{f}_h^*(t)$

$$\hat{\theta}^* = \arg\min_{\theta} \mathrm{KL}\left(\hat{f}_h^*(t), f_h^*(t; \theta)\right) = \int \log\left(\frac{\hat{f}_h^*(t)}{f_h^*(t; \theta)}\right) \hat{f}_h^*(t) dt.$$

Equivalently, $\hat{\theta}^*$ is the maximizer of

$$l^*(\theta) = \int \log\left(f_h^*(t; \theta)\right) \hat{f}_h^*(t) dt = \sum_{i=1}^{n} \int \log\left(f_h^*(t; \theta)\right) K_h(x_i, t) dt. \tag{2.5}$$

We call $l^*(\theta)$ the DS log likelihood, because $l^*(\theta)$ approaches to the ordinary log likelihood $l(\theta)$ as $h \to 0$. The DS-MLE is consistent (Seo and Lindsay, 2011) for any kernel with any fixed bandwidth h under very mild conditions. Some guidelines for the choice of the kernel and h are discussed in Seo and Lindsay (2010).

For the normal mixture density $f(x; \theta)$ in (2.1), the smoothed model density $f_h^*(t; \theta)$ can be explicitly calculated as $\sum_{j=1}^{J} p_j N(t; \mu_j, \sigma_j^2 + h)$ if $K_h(x, t)$ is the normal density with mean t and variance h. The DS log-likelihood is then

$$l^*(\theta) = \sum_{i=1}^{n} \int \log\left(\sum_{j=1}^{J} p_j N\left(t; \mu_j, \sigma_j^2 + h\right)\right) K_h(x_i, t) dt. \tag{2.6}$$

Since there is no closed form for $l^*(\theta)$, maximizing $l^*(\theta)$ requires a numerical integration. For this computational problem, Seo and Lindsay (2010) approximated $l^*(\theta)$ by

$$l_{MC}^*(\theta) = \frac{1}{nS} \sum_{i=1}^{n} \sum_{s=1}^{S} \log\left(\sum_{j=1}^{J} p_j N\left(t_{ij}; \mu_j, \sigma_j^2 + h\right)\right), \tag{2.7}$$

where $\{t_{is}, s = 1, \ldots, S\}$ is a Monte-Carlo sample generated from $K_h(x_i, t)$ for each observed x_i. With this simple strategy, maximizing $l_{MC}^*(\theta)$ is equivalent to finding the usual MLE with the smoothed model $f^*(t; \theta)$ and the augmented data $\{t_{ij} : i = 1, \ldots, n, s = 1, \ldots, S\}$. Hence, if $f_h^*(t; \theta)$ has an explicit form, one can easily find a Monte-Carlo version of the DS-MLE using standard optimization techniques such as Newton methods and the EM algorithm.

One drawback of this computation is that it may require a large amount of computing time because the program will run as if we have $n \times S$ data points instead of n. Thus, we may need significant computing time when one needs a high accuracy of $l_{MC}^*(\theta)$ or the original sample size is large. This becomes even worse if a slow optimization technique such as the EM algorithm is applied.

3. EM Algorithm for DS-MLE

A popular computational tool for mixture models is the EM algorithm because it is a stable and easy to program (though it is slow). In this section, we derive an EM algorithm to maximize $l^*(\theta)$ for finite normal mixture models. To derive the EM algorithm, we will consider the mixture model as a component membership missing problem. Let us define a multinomial component membership indicator vector $y_i = (y_{i1}, \ldots, y_{iJ})$ as

$$y_{ij} = \begin{cases} 1, & \text{if } x_i \text{ comes from } N\left(\mu_j, \sigma_j^2\right), \\ 0, & \text{Otherwise.} \end{cases}$$

Then the joint density of (x_i, y_i) is given by

$$f(x_i, y_i; \theta) = \prod_{j=1}^{J} \left[p_j N\left(x_i; \mu_j, \sigma_j^2\right)\right]^{y_{ij}}$$

and the corresponding smoothed joint density is

$$f_h^*(t, y_i; \theta) = \int f(x_i, y_i; \theta) K_h(t, x_i) dx_i.$$

If we choose the normal kernel for K_h, $f_h^*(t, y_i; \theta)$ can be further simplified as

$$f_h^*(t, y_i; \theta) = \prod_{j=1}^{J} \left[p_j N\left(t; \mu_j, \sigma_j^2 + h\right)\right]^{y_{ij}}$$

and the corresponding DS log-likelihood function is

$$l^*(\theta) = \sum_{i=1}^{n} \int \log\left(f_h^*\left(t, y_i; \theta\right)\right) K_h(t, x_i) dt.$$

In the standard EM algorithm with the ordinary complete log likelihood, the objective function $Q(\theta|\theta^{(m)})$ is obtained by taking conditional expectation to $l(\theta)$ given $(x_1, \ldots, x_n)$ and a current estimator $\theta^{(m)}$. In this case, the E-step can be explicitly obtained as we have to only consider the conditional expectation of y_{ij}. However, for $l^*(\theta)$, this is not true anymore, because the complete DS log likelihood $l^*(\theta)$ involves an intractable integral operator. To overcome this difficulty, we propose to take conditional expectation only to $\log(f_h^*(t, y_i; \theta))$ given t and a current parameter estimator $\theta^{(m)}$ so that its conditional expectation remains analytic. Note that the conditional distribution of $(y_{i1}, \ldots, y_{iJ})$ given t is simply the following multinomial distribution:

$$f_h^*(y_{i1}, \ldots, y_{iJ} \,|\, t, \theta) = \prod_{j=1}^{J} \left[\frac{p_j N\left(t; \mu_j, \sigma_j^2 + h\right)}{f_h^*(t; \theta)}\right]^{y_{ij}},$$

where $f_h^*(t; \theta) = \sum_{j=1}^{J} p_j N(t; \mu_j, \sigma_j^2 + h)$. Now we can derive the following E-step and M-step using calculus similar to the standard EM for mixtures.

1. E-step: For a current estimator $\theta^{(m)}$, we construct Q^* as

$$\begin{aligned} Q^*\left(\theta|\theta^{(m)}\right) &= \sum_{i=1}^{n} \int E\left[\log f_h^*(t, y_i; \theta)|t; \theta^{(m)}\right] K_h(t - x_i) dt \\ &= \sum_{i=1}^{n} \int \sum_{j=1}^{J} \left[E\left(y_{ij}|t, \theta^{(m)}\right) \log\left(p_j N\left(t; \mu_j, \sigma_j^2 + h\right)\right)\right] K_h(t - x_i) dt \\ &= \sum_{i=1}^{n} \sum_{j=1}^{J} \int \hat{I}_{ij}(t) \log\left(p_j N\left(t; \mu_j, \sigma_j^2 + h\right)\right) K_h(t - x_i) dt, \end{aligned} \tag{3.1}$$

where

$$\hat{I}_{ij}(t) = E\left[y_{ij}|t, \theta^{(m)}\right] = \frac{p_j^{(m)} N\left(t; \mu_j^{(m)}, \sigma_j^{2(m)} + h\right)}{f_h^*(t; \theta^{(m)})}.$$

2. M-step: The maximizer of the last displayed expression in (3.1) can be then calculated as

$$p_j^{(m+1)} = \frac{1}{n}\sum_{i=1}^{n}\int \hat{I}_{ij}(t)K_h(t-x_i)dt,$$

$$\mu_j^{(m+1)} = \frac{\sum_{i=1}^{n}\int t\hat{I}_{ij}(t)K_h(t-x_i)dt}{\sum_{i=1}^{n}\int \hat{I}_{ij}(t)K_h(t-x_i)dt},$$

$$\sigma_j^{2(m+1)} = \frac{\sum_{i=1}^{n}\int \left(t-\mu_j^{(m+1)}\right)^2 \hat{I}_{ij}(t)K_h(t-x_i)dt}{\sum_{i=1}^{n}\int \hat{I}_{ij}(t)K_h(t-x_i)dt} - h.$$

This can be obtained from the first derivative with respect to each parameter and these first derivatives are given in Appendix.

Since E-step and M-step do not have closed forms due to integral operators, a numerical integration is still required to compute the updated parameters in M-step. A simple way is to apply Monte-Carlo integration as used in $l^*_{MC}(\theta)$. If we use such a Monte-Carlo integration in E-step, the updated parameters in M-step can be approximated by

$$p_j^{(m+1)} \approx \frac{1}{nS}\sum_{i=1}^{n}\sum_{s=1}^{S}\hat{I}_{ij}(t_{is}), \tag{3.2}$$

$$\mu_j^{(m+1)} \approx \frac{1}{np_j^{(m+1)}}\sum_{i=1}^{n}\sum_{s=1}^{S} t_{is}\hat{I}_{ij}(t_{is}), \tag{3.3}$$

$$\sigma_j^{2(m+1)} \approx \frac{1}{np_j^{(m+1)}}\sum_{i=1}^{n}\sum_{s=1}^{S}\left(t_{is}-\mu_j^{(m+1)}\right)^2 \hat{I}_{ij}(t_{is}) - h, \tag{3.4}$$

where $t_{i1}, \ldots, t_{iS}$ is a random sample from $K_h(x_i, t)$ for each x_i.

This gives the same answer as that from (2.7); therefore, the estimator is still simulation-dependent and requires a large amount of computing time especially when one needs high accuracy of numerical integrations. In the next section, we propose a method to significantly reduce the computing effort with a minimal sacrifice of accuracy.

4. Local Quadratic Approximation

To avoid such time consuming simulation-based integration in M-step, we use the second-order Taylor expansion of $\hat{I}_{ij}(t)$ at $t = x_i$ for each i:

$$\hat{I}_{ij}(t) \approx \hat{I}_{ij}(x_i) + \hat{I}'_{ij}(x_i)(x_i - t) + \frac{1}{2}\hat{I}''_{ij}(x_i)(x_i - t)^2, \tag{4.1}$$

where $\hat{I}'_{ij}(t)$ and $\hat{I}''_{ij}(t)$ are the first and second derivatives of $\hat{I}_{ij}(t)$ with respect to t. The explicit formulae for $\hat{I}'_{ij}(t)$ and $\hat{I}''_{ij}(t)$ are derived in Appendix. Now, if we replace $\hat{I}_{ij}(t)$ in (3.2)–(3.4) with

(4.1), one can verify the following approximated M-step using the approximations of $\int I(t)K(t, x_i)dt$, $\int tI(t)K(t, x_i)dt$, and $\int tI(t)K(t, x_i)dt$ given in Appendix:

$$\begin{aligned}
p_j^{(m+1)} &\approx \frac{1}{n}\sum_{i=1}^{n}\int\left(\hat{I}_{ij}(x_i) + \hat{I}'_{ij}(x_i)(x_i - t) + \frac{1}{2}\hat{I}''_k(x_i)(x_i - t)^2\right)K_h(x_i, t)dt \\
&= \frac{1}{n}\sum_{i=1}^{n}\hat{I}_{ij}(x_i) + \frac{h}{2n}\sum_{i=1}^{n}\hat{I}''_{ij}(x_i), \\
\mu_j^{(m+1)} &\approx \frac{1}{np_j^{(m+1)}}\sum_{i=1}^{n}\int t\left(\hat{I}_{ij}(x_i) + \hat{I}'_{ij}(x_i)(x_i - t) + \frac{1}{2}\hat{I}''_k(x_i)(x_i - t)^2\right)K_h(x_i, t)dt \\
&= \frac{1}{np_j^{(m+1)}}\sum_{i=1}^{n}\left(x_i\hat{I}_{ij}(x_i) + h\hat{I}'_{ij}(x_i) + \frac{hx_i\hat{I}'_{ij}(x_i)}{2}\right), \\
\sigma_j^{2(m+1)} &\approx \frac{1}{np_j^{(m+1)}}\sum_{i=1}^{n}\int\left(t - \mu_j^{(m+1)}\right)^2\left(\hat{I}_{ij}(x_i) + \hat{I}'_{ij}(x_i)(x_i - t) + \frac{1}{2}\hat{I}''_k(x_i)(x_i - t)^2\right)K_h(x_i, t)dt \\
&= \frac{1}{np_j^{(m+1)}}\sum_{i=1}^{n}\left(\hat{I}_{ij}(x_i)\left(h + x_i^2\right) + 2\hat{I}'_{ij}(x_i)x_ih + \frac{\hat{I}''_{ij}(x_i)}{2}h\left(3h + x_i^2\right)\right) - \left(\mu_j^{(m+1)}\right)^2.
\end{aligned}$$

The accuracy of this approximation depends on the moment of $(T - X_i)^3$ or $(T - X_i)^4$, so the magnitude of h is an important factor to control the accuracy of approximation. Clearly, an approximation error becomes large as h increases. In this case, we may need a higher-order Taylor expansion in (4.1) to gain more accuracy. However, a typical choice of h for the DS-MLE would be very small in the normal mixture case although any choice of h guarantees the consistency of DS-MLE. This is because a very small h is enough to remove degeneracy as shown in Seo and Linday (2010). Of course, a large h can also remove degeneracy but with some information loss.

There is a practical guideline for the choice of h in the DS-MLE. Seo and Linday (2010) suggested to use *spectral degrees of freedom*(sDOF) as a tool to determine a reasonable range of bandwidth that utilizes the spectral decomposition of quadratic distances for probability distributions. The empirical sDOF for a given kernel $K_h(\cdot\,,\cdot)$ with a bandwidth h can be calculated as

$$\widehat{\text{sDOF}} = \frac{\left(\frac{1}{n}\sum_i \tilde{K}_{2h}(x_i, x_j)\right)^2}{\frac{2}{n(n-1)}\sum_{i<j}\left(\tilde{K}_{2h}(x_i, x_j)\right)^2},$$

where

$$\tilde{K}_{2h}(x_i, x_j) = K_{2h}(x_i, x_j) - \frac{1}{n}\sum_i K_{2h}(x_i, x_j) - \frac{1}{n}\sum_j K_{2h}(x_i, x_j) + \frac{1}{n^2}\sum_i\sum_j K_{2h}(x_i, x_j).$$

For more detail, see Lindsay *et al.* (2008) and Ray and Lindsay (2008).

Since the sDOF is analogous to the usual degrees of freedom in the Chi-squared goodness of fit test, a rough rule of thumb is to choose h of which the sDOF is between 5 and $n/5$. A sDOF for a given kernel and bandwidth greater than $n/5$ implies that h is too small; however, the corresponding h is too large if the sDOF is less than 5. Although this guideline can be adopted to our case, for the proposed quadratic approximation, we recommend to use h whose sDOF is close to $n/5$ (the upper bound of sDOF in the guideline) to minimize the approximation error due to a large h.

Table 1: Computing times in seconds for the proposed method and Monte-Carlo EM.

h	MLE*	S				
		200	400	600	800	1000
0.001	0.59	11.13	23.85	47.70	75.38	71.33
0.01	0.62	33.36	56.06	77.86	104.02	120.84
0.1	0.59	66.95	120.59	165.46	215.84	251.56

5. Numerical Example

In this section, we present results of some numerical studies to see the performance of the proposed method in terms of the speed and accuracy compared to the Monte-Carlo method. Analysis of the parameter estimate with mixture models has some complex features due to label switching, spurious solutions, and multiple modes. It is very difficult to remove all of these undesirable feature with mixture models. For this reason, in this section, we only consider the speed and accuracy of the proposed method compared to the Monte-Carlo method proposed in Seo and Lindsay (2010).

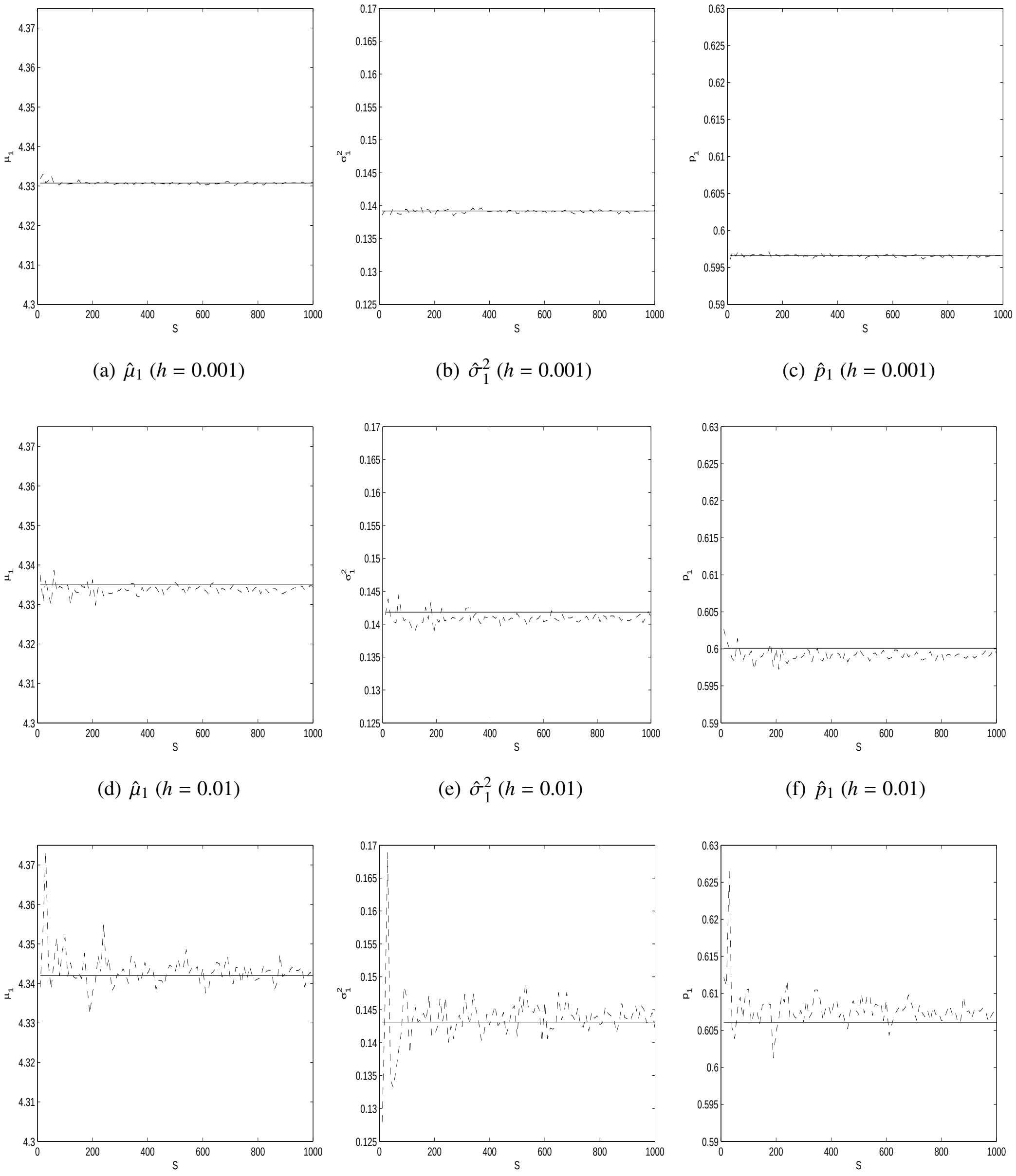
5.1. Real data example

To show the performance of the proposed method, we first use the Acidity data set that contains acidity indices from 155 lakes in the Northeastern United States (Crawford *et al.*, 1992). Many authors have used this data set to illustrate normal mixture models and found that the appropriate number of components would be two to five. To assess the performance of the proposed method, we use a two-component normal mixture model

$$\frac{p_1}{\sqrt{2\pi\sigma_1^2}}\exp\left(-\frac{(x_i-\mu_1)^2}{2\sigma_1^2}\right)+\frac{p_2}{\sqrt{2\pi\sigma_2^2}}\exp\left(-\frac{(x_i-\mu_2)^2}{2\sigma_2^2}\right).$$

Since we only aim to see the speed and accuracy of the proposed method, we use just one initial value for the EM algorithm. To choose the initial value, we first run the EM algorithm using a set of random multiple initial values. We found three local maxima and several degenerate parameter estimates that give infinite likelihood values as found in Seo and Lindsay (2010). For the initial value to be used in our example, we choose the one with the largest likelihood value among all non-degenerate local maxima. We expect that using this initial value is suitable to efficiently assess the performance of the proposed estimator because this frees us from the issues regarding degeneracy and multimodality.

To see the performance of the MCEM proposed by Seo and Lindsay (2010) and the new EM, we choose $h = 0.1, 0.01, 0.001$. Note that the empirical sDOF values corresponding to these h values range from 5 to 31 (=155/5). Figure 1 shows the estimated (μ_1, σ_1^2, p) over various S values. The x-axis represents S, the number of random numbers used in (2.7) for each datum. The dashed line represents the estimated parameter value obtained from the MCEM for each S. The solid line stands for the parameter estimate from the proposed method denoted as DSEM. Note that DSEM does not require Monte Carlo samples, and thus it remains the same over S. Due to the variability of the Monte-Carlo samples used in MCEM, the estimated parameter values using MCEM fluctuate and the magnitude of fluctuation taper off as S increases. For $h = 0.001$, it seems that $S = 50$ gives stable estimates while more than 1000 Monte Carlo random numbers are required for $h = 0.1$. However, DSEM gives quite accurate values similar to MCEM with $S = 1000$. This confirms that the proposed method provides a relatively high accuracy without simulation based computation. Table 1 shows the required computing times in seconds. Note that MCEM requires tremendous computing time compared to DSEM without a significant gain of accuracy. Together this with the observation from



(a) $\hat{\mu}_1$ ($h = 0.001$) (b) $\hat{\sigma}_1^2$ ($h = 0.001$) (c) $\hat{p}_1$ ($h = 0.001$)

(d) $\hat{\mu}_1$ ($h = 0.01$) (e) $\hat{\sigma}_1^2$ ($h = 0.01$) (f) $\hat{p}_1$ ($h = 0.01$)

(g) $\hat{\mu}_1$ ($h = 0.1$) (h) $\hat{\sigma}_1^2$ ($h = 0.1$) (i) $\hat{p}_1$ ($h = 0.1$)

Figure 1: *Parameter estimates from MCEM (dashed line) and DSEM (solid line)*

Figure 1, the proposed method is quite fast compared to MCEM while maintaining high accuracy. This improvement would become more evident when one uses mixtures with more than two components or the sample size is large.

Table 2: Bias, standard error, and average runtime of each algorithm

h	Method	bias × 100 (standard error × 100)					Average runtime
		μ_1	μ_2	σ_1^2	σ_1^2	p	
-	EM	−0.34(16.01)	0.64(14.92)	−3.44(23.52)	−3.60(22.78)	0.37(5.67)	0.11
0.01	DSEM	−0.32(15.98)	0.66(14.91)	−3.40(23.44)	−3.64(22.69)	0.37(5.67)	0.11
	MCEM50	−0.38(16.01)	0.64(14.90)	−3.40(23.44)	−3.66(22.68)	0.37(5.67)	2.18
	MCEM100	−0.35(15.99)	0.65(14.91)	−3.44(23.48)	−3.60(22.66)	0.37(5.67)	4.08
	MCEM300	−0.32(16.00)	0.67(14.92)	−3.38(23.48)	−3.65(22.71)	0.37(5.67)	10.97
	MCEM500	−0.33(15.98)	0.66(14.90)	−3.44(23.40)	−3.66(22.69)	0.37(5.67)	17.26
0.3	DSEM	0.23(16.16)	1.07(15.90)	−1.63(24.97)	−4.18(22.99)	0.46(5.84)	0.17
	MCEM50	−0.23(16.17)	1.04(15.31)	−2.53(23.70)	−4.79(22.42)	0.44(5.74)	3.04
	MCEM100	−0.10(16.01)	1.05(15.25)	−2.80(23.83)	−4.52(22.22)	0.44(5.77)	5.63
	MCEM300	0.02(16.06)	1.10(15.27)	−2.56(23.55)	−4.51(22.26)	0.44(5.74)	14.91
	MCEM500	−0.02(15.97)	1.09(15.20)	−2.77(23.62)	−4.63(22.29)	0.44(5.76)	23.47

5.2. Simulation study

As reviewers requested, we also conduct a simple simulation experiment to see the performance of the DS-MLE with each algorithm. Again, we mainly focus on the accuracy and speed of each algorithm in this simulation study. For this, we generate $n = 100$ random sample from $pN(\mu_1, \sigma_1^2) + (1-p)N(\mu_2, \sigma_2^2)$ to find the MLE and the DS-MLE with MCEM and the proposed algorithm. The true parameter used in this simulation is $(\mu_1, \mu_2, \sigma_1^2, \sigma_2^2, p) = (0, 5, 1, 1, 0.5)$. This true parameter value represents a well-separated normal mixture density. The initial set of parameter values is set to be true parameter values. We choose this well separated mixture model and the true initial value to remove undesirable features such as labeling switching and spurious solutions when we summarize the results; subsequently, the ordinary EM algorithm hardly converges to degenerate solutions. Hence this enables us to only see the accuracy and speed of the MCEM and DSEM. For all algorithms we considered here, each algorithm is stopped if the difference of the consecutive log likelihood values was less than 10^{-7}. We also test two choices of bandwidths $h = 0.01$ and 0.3 that represent the maximum and minimum bandwidths based on the guidelines explained in Section 4.

Table 2 shows the empirical bias, standard error, and average runtime for each algorithm based on 200 replications. For each sample, the ordinary EM algorithm is applied and then DSEM and MCEM are used for $h = 0.01$ and 0.3. In Table 2, MCEM50, MCEM100, MCEM300, and MCEM500 represent the MCEM algorithms with $S = 50$, 100, 300, 500, respectively. For both $h = 0.01$ and 0.3, all methods show similar performance but MCEM's require a large amount of computing time. Although $h = 0.3$ is quite a large bandwidth, the DS-MLE obtained by either DSEM or MCEM still produces reasonable estimators. This shows that the DS-MLE is not so sensitive to the choice of h. The average runtime for MCEM is very large compared to EM while the runtime of DSEM is slightly larger than EM. From this limited simulation, we can see that the proposed algorithm provides sufficient accuracy while it dramatically reduces the computing time compared to MCEM.

6. Conclusion

In this paper, we derive an EM procedure for the DS-MLE in normal mixture models. EM algorithms for other types of mixture models can be obtained similarly. In addition, we show that a quadratic approximation can dramatically reduce computing effort while it maintains high precision. However, when one needs to use a large bandwidth, the proposed method may not provide an accurate estimator. Hence, if the DS-MLE is used to remove degenerate MLEs, we recommend using the smallest bandwidth within the range of the bandwidth determined by the sDOF. It would be an important future task to develop a fast accurate method for the DS-MLE with a large bandwidth.

Appendix:

- The first derivatives of Q^* to obtain M-step in Section 3

$$\frac{dQ^*}{dp_j} = \sum_{j=1}^{J} \frac{1}{p_j} \left(\sum_{i=1}^{n} \int \hat{I}_{ij}(t) K_h(t - x_i) dt \right),$$

$$\frac{dQ^*}{d\mu_j} = \sum_{i=1}^{n} \int \left(\frac{t - \mu_j}{\sigma_1^2 + h} \right) \hat{I}_{ij}(t) K_h(t - x_i) dt,$$

$$\frac{dQ^*}{d\sigma_j^2} = \sum_{i=1}^{n} \int \left(\frac{\left(t - \mu_j^2\right)}{\left(\sigma_j^2 + h\right)^2} - \frac{1}{\left(\sigma_j^2 + h\right)} \right) \hat{I}_{ij}(t) K_h(t - x_i) dt.$$

- The explicit forms of $\hat{I}'_{ij}(t)$ and $\hat{I}''_{ij}(t)$

  Let $a_j(t) = p_j N(t; \mu_j, \sigma_j^2 + h)$ and $A(t) = \sum_j a_j(t)$, where $N(t; \mu, \sigma^2)$ is the normal density with mean μ and variance σ^2. One can then easily verify $a'_j(t) = -\{(t - \mu_j)/(\sigma_j^2 + h)\} a_j(t)$, $a''_j(t) = [\{(t - \mu_j)/(\sigma_j^2 + h)\} - 1/(\sigma_j^2 + h)] a_j(t)$, $A'(t) = \sum_j a'_j(t)$, and $A''(t) = \sum_j a''_j(t)$. Under these notations, we can verify

$$\hat{I}'_{ij}(t) = \frac{a'_j(t) A(t) - a_j(t) A'(t)}{(A(t))^2}$$

  and

$$\hat{I}''_{ij}(t) = \frac{a''_j(t) A(t) - a_j(t) A''(t)}{(A(t))^2} - \frac{2A'(t) \hat{I}'_{ij}(t)}{A(t)}.$$

- Approximations of $\int I(t) K_h(t, x_i) dt$, $\int t I(t) K_h(t, x_i) dt$, and $\int t^2 I(t) K_h(t, x_i) dt$ for $p_j^{(m+1)}$, $\mu_j^{(m+1)}$, and $\sigma_j^{2(m+1)}$ in Section 4.

  For the normal kernel $K_h(x_i, t)$ with variance h, we have $\int K_h(x_i, t) dt = 1$, $\int t K_h(x_i, t) dt = x_i$, $\int t^2 K_h(x_i, t) dt = x_i^2 + h$, $\int t^3 K_h(x_i, t) dt = x_i^3 + 3x_i h$, and $\int t^4 K_h(x_i, t) dt = x_i^4 + 6x_i^2 h + 3h^2$. Using these and tedious calculus, one can verify followings:

$$\begin{aligned}
\int I(t) K_h(t, x_i) dt &\approx \int \left(I(x_i) + I'(x_i)(t - x_i) + \frac{1}{2} I''(x_i)(t - x_i)^2 \right) K_h(t, x_i) dt \\
&= I(x_i) \int K_h(t, x_i) dt + I'(x_i) \int (t - x_i) K_h(t, x_i) dt + \frac{1}{2} I''(x_i) \int (t - x_i)^2 K_h(t, x_i) dt \\
&= I(x_i) + \frac{h}{2} I''(x_i), \\
\int t I(t) K_h(t, x_i) dt &\approx \int t \left(I(x_i) + I'(x_i)(t - x_i) + \frac{1}{2} I''(x_i)(t - x_i)^2 \right) K_h(t, x_i) dt \\
&= I(x_i) \int t K_h(t, x_i) dt + I'(x_i) \int t(t - x_i) K_h(t, x_i) dt + \frac{1}{2} I''(x_i) \int t(t - x_i)^2 K_h(t, x_i) dt
\end{aligned}$$

$$
\begin{aligned}
&= x_i I(x_i) + hI'(x_i) + \frac{hx_i I''(x_i)}{2}, \\
\int t^2 I(t) K_h(t, x_i) dt &\approx \int t^2 \left(I(x_i) + I'(x_i)(t - x_i) + \frac{1}{2} I''(x_i)(t - x_i)^2 \right) K_h(t, x_i) dt \\
&= I(x_i) \int t^2 K_h(t, x_i) dt + I'(x_i) \int t^2 (t - x_i) K_h(t, x_i) dt \\
&\quad + \frac{1}{2} I''(x_i) \int t^2 (t - x_i)^2 K_h(t, x_i) dt \\
&= I(x_i) \left(h + x_i^2 \right) + 2I'(x_i) x_i h + \frac{1}{2} I''(x_i) h \left(3h + x_i^2 \right).
\end{aligned}
$$

References

Chen, J., Tan, X. and Zhang, R. (2008). Consistency of penalized mle for normal mixtures in mean and variance, *Statistica Sinica*, **18**, 443–465.

Ciuperca, G. A., Ridolfi, A. and Idier, J. (2003). Penalized maximum likelihood estimator for normal mixtures, *Scandinavian Journal of Statistics*, **30**, 45–59.

Crawford, S. K., Degroot, M. H., Kadane, J. B. and Small, M. J. (1992). Modeling lake chemistry distributions: Approximate Bayesian methods for estimating a finite mixture model, *Technometrics*, **34**, 441–453.

Fraley, C. and Raftery, A. E. (2007). Bayesian regularization for normal mixture estimation and model-based clustering, *Journal of Classification*, **24**, 155–181.

Hathaway, R. J. (1985). A constrained formulation of maximum likelihood estimation for normal mixture distributions, *Annals of Statistics*, **13**, 795–800.

Hathaway, R. J. (1986). A constrained EM-algorithm for univariate normal mixtures, *Computational Statistics & Data Analysis*, **23**, 211–230.

Ingrassia, S. and Rocci, R. (2007). Constrained monotone EM algorithms for finite mixture of multivariate Gaussians, *Computational Statistics & Data Analysis*, **51**, 5339–5351.

Kiefer, J. and Wolfowitz, J. (1956). Consistency of the maximum likelihood estimator in the presence of infinitely many incidental parameters, *Annals of Statistics*, **27**, 886–906.

Lindsay, B. G., Markatou, M., Ray, S., Yang, K. and Chen, S. (2008). Quadratic distances on probabilities: A unified foundation, *Annals of Statistics*, **36**, 983–1006.

Ray, S. and Lindsay, B. G. (2008). Model selection in high-dimensions: A quadratic-risk based approach, *Journal of the Royal Statistical Society, Series B*, **70**, 95–118.

Seo, B. and Lindsay, B. G. (2010). A computational strategy for doubly smoothed MLE exemplified in the normal mixture model, *Computational Statistics and Data Analysis*, **54**, 1930–1941.

Seo, B. and Lindsay, B. G. (2011). A universally consistent modification of maximum likelihood, *Submitted.*

Tanaka, K. and Takemura, A. (2006). Strong consistency of the maximum likelihood estimator for finite mixtures of location-scale distributions when the scale parameters are exponentially small, *Bernoulli*, **12**, 1003–1017.

Received September 2011; Revised November 2011; Accepted November 2011