Maximum Lq-Likelihood Estimation via the Expectation Maximization Algorithm: A Robust Estimation of Mixture Models

Yichen Qin and Carey E. Priebe*

Abstract

We introduce a maximum Lq-likelihood estimation (MLqE) of mixture models using our proposed expectation maximization (EM) algorithm, namely the EM algorithm with Lq-likelihood (EM-Lq). Properties of the MLqE obtained from the proposed EM-Lq are studied through simulated mixture model data. Compared with the maximum likelihood estimation (MLE) which is obtained from the EM algorithm, the MLqE provides a more robust estimation against outliers for small sample sizes. In particular, we study the performance of the MLqE in the context of the gross error model, where the true model of interest is a mixture of two normal distributions, and the contamination component is a third normal distribution with a large variance. A numerical comparison between the MLqE and the MLE for this gross error model is presented in terms of Kullback Leibler (KL) distance and relative efficiency.

Keywords: EM algorithm, mixture model, gross error model, robustness

*Yichen Qin is PhD student (E-mail: yqin2@jhu.edu) and Carey E. Priebe is Professor (E-mail: cep@jhu.edu), Department of Applied Mathematics and Statistics, Johns Hopkins University, 100 Whitehead Hall, 3400 North Charles Street, Baltimore, MD 21210.

1 INTRODUCTION

Maximum likelihood is among the most commonly used estimation procedures. For mixture models, the maximum likelihood estimation (MLE) via the expectation maximization (EM) algorithm introduced by Dempster et al. (1977) is a standard procedure. Recently, Ferrari and Yang (2010) introduced the concept of maximum Lq-likelihood estimation (MLqE), which can yield robust estimation by trading bias for variance, especially for small or moderate sample sizes. This article combines the MLqE with the EM algorithm to obtain the robust estimation for mixture models, and studies the performance of this robust estimator.

In this article, we propose a new EM algorithm — namely expectation maximization algorithm with Lq-likelihood (EM-Lq) which addresses MLqE within the EM framework. In the EM-Lq algorithm, we propose a new objective function at each M step which plays the role that the complete log likelihood plays in the traditional EM algorithm. By doing so, we inherit the robustness of the MLqE and make it available for mixture model estimation.

Our study focuses on the performance of the MLqE for estimation in a gross error model $f_0^*(x) = (1-\epsilon)f_0(x)+\epsilon f_{\text{err}}(x)$, where $f_0(x)$ is what we are interested in estimating and $f_{\text{err}}(x)$ is the measurement error component. For simplicity, we consider the object of interest $f_0(x)$ to be a mixture of two normal distributions. And $f_{\text{err}}(x)$ is a third normal distribution with a large variance. We will examine the properties of the MLqE, in comparison to that of the MLE, at different levels of the contamination ratio ϵ.

The measurement error problem is one of the most practical problems in Statistics. Let us consider that some measurements $X = (X_1, X_2, ..., X_n)$ are produced by a scientific experiment. X has a distribution f_θ with a interpretable parameter θ that we are interested in. However, we do not observe X directly. Instead, we observe $X^* = (X_1^*, X_2^*, ..., X_n^*)$ where most of the $X_i^* = X_i$, but there are a few outliers. In other words, X^* is X contaminated with gross errors which are mostly due to either human error or instrument malfunction.

But f_θ is still the target of our estimation (Bickel and Doksum (2007)). To overcome this problem and still be able to do statistical inference for f_θ in the mixture model case, we come up with this idea of the EM-Lq.

There has been an extensive amount of early work on robust estimation of mixture models and clustering. For example, Peel and McLachlan (2000) used t distributions instead of normal distributions to incorporate the phenomena of fat tails, and gave a corresponding expectation/conditional maximization (ECM) algorithm which was originally introduced by Meng and Rubin (1993). McLachlan et al. (2006) further formalized this idea and applied it to robust cluster analysis. Tadjudin and Landgrebe (2000) made a contribution on robust estimation of mixture model parameters by using both labeled and unlabeled data, and assigning different weights to different data points. Garcia-Escudero and Gordaliza (1999) studied the robustness properties of the generalized k-means algorithm from the influence function and the breakpoint perspectives. Finally, Cuesta-Albertos et al. (2008) applied the trimmed subsample of the data for fitting mixture models and iteratively adjusted the subsample after each estimation.

The remainder of this article is organized as follows. Section 2 gives an introduction to the MLqE along with its advantages compared to the MLE. Properties of the MLqE for mixture models are discussed in Section 3. In Section 4, we present our EM-Lq, and explain the rationale behind it. The application of the EM-Lq in mixture models is introduced and discussed in Section 5. The comparisons of the MLqE (obtained from the EM-Lq) and the MLE based on simulation as well as real data are presented in Section 6. We address the issue of tuning parameter q in Section 7. We conclude with a discussion and directions for future research in Section 8, and relegate the proofs to Section 9.

2 MAXIMUM Lq-LIKELIHOOD ESTIMATION

2.1 Definitions and Basic Properties

First, let us start with the traditional maximum likelihood estimation. Suppose data X follows a distribution with probability density function f_θ parameterized by $\theta \in \Theta \subset \mathbb{R}^d$. Given the observed data $\mathbf{x} = (x_1, ..., x_n)$, the maximum likelihood estimate is defined as $\hat{\theta}_{\text{MLE}} = \arg\max_{\theta\in\Theta}\{\sum_{i=1}^n \log f(x_i;\theta)\}$. Similarly, the maximum Lq-likelihood estimate (Ferrari and Yang 2010) is defined as

$$\hat{\theta}_{\text{ML}q\text{E}} = \arg\max_{\theta\in\Theta} \sum_{i=1}^n L_q(f(x_i;\theta)),$$

where $L_q(u) = (u^{1-q}-1)/(1-q)$ and $q > 0$. By L'Hopital's rule, when $q \to 1$, $L_q(u) \to \log(u)$. The tuning parameter q is called the distortion parameter, which governs how distorted L_q is away from the log function. Based on this property, we conclude that the MLqE is a generalization of the MLE.

Define $U(x;\theta) = \nabla_\theta \log f(x;\theta) = f'_\theta(x;\theta)/f(x;\theta)$ and $U^*(x;\theta,q) = \nabla_\theta L_q(f(x;\theta)) = U(x;\theta)f(x;\theta)^{1-q}$, we know that $\hat{\theta}_{\text{MLE}}$ is a solution of the likelihood equation $0 = \sum_{i=1}^n U(x_i;\theta)$. Similarly, $\hat{\theta}_{\text{ML}q\text{E}}$ is a solution of the Lq-likelihood equation

$$0 = \sum_{i=1}^n U^*(x_i;\theta,q) = \sum_{i=1}^n U(x_i;\theta)f(x_i;\theta)^{1-q}. \tag{1}$$

It is easy to see that $\hat{\theta}_{\text{ML}q\text{E}}$ is a solution to a weighted version of the likelihood equation that $\hat{\theta}_{\text{MLE}}$ solves. The weights are proportional to the power transformation of the probability density function, $f(x_i;\theta)^{1-q}$. When $q < 1$, the MLqE puts more weight on the data points with high likelihoods, and less weight on the data points with low likelihoods. The tuning parameter q adjusts how aggressively the MLqE distorts the weight allocation. The MLE

can be considered as a special case of the MLqE with equal weights.

In particular, when f is a normal distribution, our $\hat{\mu}_{\mathrm{ML}q\mathrm{E}}$ and $\hat{\sigma^2}_{\mathrm{ML}q\mathrm{E}}$ satisfy

$$\hat{\mu}_{\mathrm{ML}q\mathrm{E}} = \frac{1}{\sum_{i=1}^{n} w_i} \sum_{i=1}^{n} w_i x_i, \tag{2}$$

$$\hat{\sigma^2}_{\mathrm{ML}q\mathrm{E}} = \frac{1}{\sum_{i=1}^{n} w_i} \sum_{i=1}^{n} w_i (x_i - \hat{\mu}_{\mathrm{ML}q\mathrm{E}})^2, \tag{3}$$

where $w_i = \varphi(x_i; \hat{\mu}_{\mathrm{ML}q\mathrm{E}}, \hat{\sigma^2}_{\mathrm{ML}q\mathrm{E}})^{1-q}$ and φ is a normal probability density function.

From equations (2) and (3), we conclude that the MLqE of the mean and the variance of a normal distribution are just the weighted mean and weighted variance. When $q < 1$, the MLqE gives smaller weights for data points lying in the tail of the normal distribution, and puts more weights on data points near the center. By doing so, the MLqE becomes less sensitive to outliers than the MLE at the cost of introducing bias into the estimation. A simple and fast re-weighting algorithm is available for solving (2) and (3). Details of the algorithm are described in Section 9.

2.2 Consistency and Bias-Variance Trade Off

Before discussing the consistency of the MLqE, let us look at the MLE first. It is well studied that the MLE is quite generally a consistent estimator. Suppose the true distribution $f_0 \in \mathcal{F}$, where $\mathcal{F}$ is a family of distributions; we know that $f_0 = \arg\max_{g \in \mathcal{F}} E_{f_0} \log g(X)$, which shows the consistency of the MLE. However, when we replace the log function with the L_q function, we do not have the same property.

We first define $f^{(r)}$, a transformed distribution of f called the escort distribution, as

$$f^{(r)} = \frac{f(x;\theta)^r}{\int f(x;\theta)^r dx}. \tag{4}$$

We also define $\mathcal{F}$ to be a family of distributions that is closed under such a transformation

(i.e., $\forall f \in \mathcal{F}$, $f^{(r)} \in \mathcal{F}$). Equipped with these definitions, we have the following property:

$$f_0^{(1/q)} = \arg\max_{g \in \mathcal{F}} E_{f_0} L_q(g(X)).$$

Thus we see that the maximizer of the expectation of Lq-likelihood is the escort distribution ($r = 1/q$) of the true density f_0. In order to also achieve consistency for the MLqE, Ferrari and Yang (2010) let q tend to 1 as n approaches infinity.

For a parametric distribution family $\mathcal{G} = \{f(x;\theta) : \theta \in \Theta\}$, suppose it is closed under the escort transformation (i.e., $\forall \theta \in \Theta$, $\exists \theta' \in \Theta$, s.t. $f(x;\theta') = f(x;\theta)^{(1/q)}$). We have a similar property, $\tilde{\theta} = \arg\max_{\theta \in \Theta} E_{\theta_0} L_q(f(X;\theta))$, where $\tilde{\theta}$ satisfies $f(x;\tilde{\theta}) = f(x;\theta_0)^{(1/q)}$

We now understand that, when maximizing the Lq-likelihood, we are essentially finding the escort distribution of the true density, not the true density itself, so our MLqE is asymptotically biased. However, this bias can be compensated by variance reduction if the distortion parameter q is properly selected. Take the MLqE for the normal distribution for example. With an appropriate $q < 1$, the MLqE will partially ignore the data points on the tails while focusing more on fitting data points around the center. The MLqE obtained this way is possibly biased (especially for the scale parameter), but will be less volatile to a significant change of data on the tails, hence, a good example of bias-variance trade off. q can be considered as a tuning parameter that adjusts the magnitude of the bias-variance trade off.

2.3 Confidence Intervals

There are generally two ways to construct confidence intervals for the MLqE. One is parametric, the other is nonparametric. In this section, we discuss the univariate case. The multivariate case can be extended naturally.

For the parametric way, we know that the MLqE is an M-estimator, whose asymptotic

variance is available. In order to have the asymptotic variance be valid, we need the sample size to be reasonably large so that the Central Limit Theorem works. However, in our application, the MLqE deals with small or moderate sample sizes in most cases. So the parametric way is not ideal, but it does provide a guideline to evaluate the estimator.

The second way is the nonparametric bootstrap method. We create bootstrap samples from the original sample, calculate their MLqEs for all bootstrap samples. We further calculate the lower and upper quantiles of these MLqEs, and call these quantiles the lower and upper bounds of the confidence interval. This method is model agnostic, and works well with the MLqE.

3 MLqE OF MIXTURE MODELS

We now look at the problem of estimating mixture models. A mixture model is defined as $f(x) = \sum_{j=1}^{k} \pi_j f_j(x; \theta_j)$. Unlike the exponential family which is proved to be closed under the escort transformation (equation (4)), the mixture model family is not closed under such a transformation. For example, consider a mixture model with the complexity $k = 2$. The escort transformation with $1/q = 2$ of this distribution is $f(x)^{(1/q)} \propto (\pi_1\varphi_1(x) + \pi_2\varphi_2(x))^2 = \pi_1^2\varphi_1(x)^2 + \pi_2^2\varphi_2(x)^2 + 2\pi_1\pi_2\varphi_1(x)\varphi_2(x)$, which is a mixture model with three components.

More generally, suppose $f_0 \in \mathcal{F}$, where $\mathcal{F}$ is a mixture model family with complexity k. Since $f_0^{(1/q)} \notin \mathcal{F}$, we know that

$$f_0^{(1/q)} \neq \tilde{g} := \arg\max_{g \in \mathcal{F}} E_{f_0} L_q(g(X)),$$

where $\tilde{g}$ can be considered as the projection of $f_0^{(1/q)}$ onto $\mathcal{F}$. Again, the MLqE of mixture models brings more bias to the estimate. This time, the new bias is a model bias as opposed to the estimation bias which we have discussed in the previous section. When estimating

mixture models using the MLqE, we carry two types of bias: estimation bias and model bias. The distortion parameter q now adjusts both of them. This idea is illustrated in Figure 1a.

There is a simple way to partially correct the bias. Since we know that the MLqE is unbiased for the escort distribution of the true distribution. After we obtain the MLqE from data, $\hat{f}_{\mathrm{ML}q\mathrm{E}}$, we can blow it up by a power transformation $g = \hat{f}^q_{\mathrm{ML}q\mathrm{E}} / \int \hat{f}^q_{\mathrm{ML}q\mathrm{E}} dx$ to get a less biased estimate. However, this only partially corrects the bias since the projection from the escort distribution onto the mixture model family cannot be recovered by this transformation.

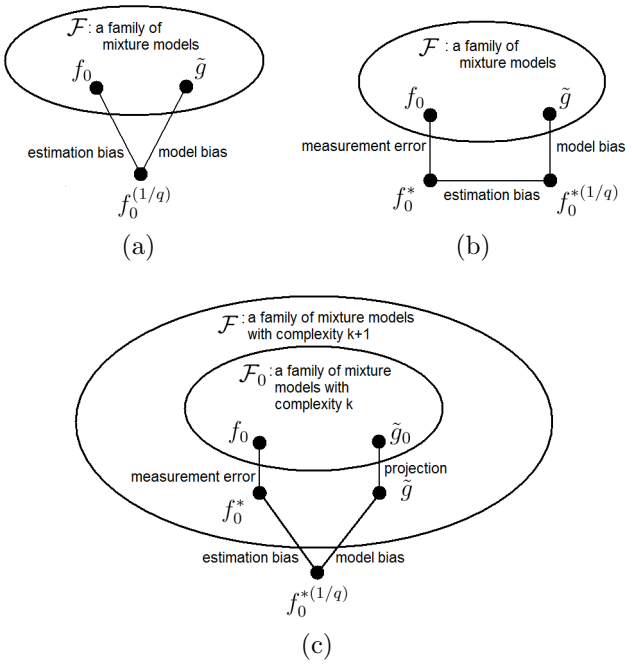


Figure 1: Illustration of the MLqE of mixture models: a) shows the usual case, which is the MLqE of mixture models with correctly specified models, b) shows the MLqE of non-measurement error components f_0 within the gross error model f_0^* using the misspecified model, c) shows the MLqE of non-measurement error components f_0 within the gross error model f_0^* using the correctly specified model.

Because the MLqE has the desirable property of being robust against outliers, we introduce the gross error model to evaluate the MLqE's performance. A gross error model is defined as $f_0^*(x) = (1-\epsilon) f_0(x) + \epsilon f_{\mathrm{err}}(x)$, where f_0 is a mixture model with complexity k,

f_{err} can be considered as a measurement error component, and ϵ is the contamination ratio. Hence, f_0^* is also a mixture model with complexity $k+1$. The gross error density f_0^* can be considered as a small deviation from the target density f_0. In order to build an estimator for f_0 that is robust against f_{err}, we apply the MLqE. Generally, there are two ways to apply the MLqE in this situation.

First, we can directly use a mixture model with complexity k to estimate f_0 based on data from f_0^*. We call this approach the direct approach. This time the model is more complex than before. The idea is illustrated in Figure 1b. Suppose $\mathcal{F}$ is a mixture model family with complexity k, and $f_0 \in \mathcal{F}$, $f_0^* \notin \mathcal{F}$, $f_0^{*(1/q)} \notin \mathcal{F}$. We obtain the ML$q$E of $f_0(x)$, $\tilde{g}$, by

$$f_0^{*(1/q)} \neq \tilde{g} := \arg\max_{g \in \mathcal{F}} E_{f_0^*} L_q(g(X)).$$

Here we use the estimation bias and the model bias to offset the measurement error effect on f_0. Please note that this approach is essentially an estimation under the misspecified model.

The second approach is that we use a mixture model with complexity $k+1$ to estimate f_0^* and project the estimate to the k component mixture model family by removing the largest variance component (i.e., the measurement error component) and normalizing the weights. We call this approach the indirect approach. The projected model is our estimate for f_0. In this case, we essentially treat the parameters of the measurement error component as nuisance parameters. This idea is illustrated in Figure 1c. In Figure 1c, $\tilde{g}$ is our estimate of f_0^*. And $\tilde{g}_0$, the projection of $\tilde{g}$ onto $\mathcal{F}_0$, is our estimate of f_0. This approach is an estimation conducted under the correctly specified model. Although the model is correctly specified, we may have higher estimation variance as we estimate more parameters.

In this article, we will study the MLqE using the above two approaches.

4 EM ALGORITHM WITH Lq-LIKELIHOOD

We now propose a variation of the EM algorithm — the expectation maximization algorithm with Lq-likelihood (EM-Lq), which gives the local maximum Lq-likelihood. Before introducing our EM-Lq, let us briefly review the rationale of the EM. Throughout this article, we use X, Z, $\mathbf{Z}$ for random variables and vectors, and x, z, $\mathbf{z}$ for realizations.

4.1 Why Does the EM Algorithm Work

The EM algorithm is an iterative method for finding a local maximum likelihood by making use of observed data X and missing data Z. The rationale behind the EM is that

$$\sum_{i=1}^{n} \log p(x_i;\Psi) = \underbrace{\sum_{i=1}^{n} E_{\Psi^{\text{old}}}[\log p(X,Z;\Psi)|X=x_i]}_{J(\Psi,\Psi^{\text{old}})} \quad \underbrace{-\sum_{i=1}^{n} E_{\Psi^{\text{old}}}[\log p(Z|X;\Psi)|X=x_i]}_{K(\Psi,\Psi^{\text{old}})},$$

where $J(\Psi,\Psi^{\text{old}})$ is the expected complete log likelihood, and $K(\Psi,\Psi^{\text{old}})$ takes its minimum at $\Psi=\Psi^{\text{old}}$ and $\frac{\partial}{\partial\Psi}K(\Psi,\Psi^{\text{old}})\big|_{\Psi=\Psi^{\text{old}}}=0$. Standing at the current estimate Ψ^{old}, to climb uphill on $\sum_{i=1}^{n}\log p(x_i;\Psi)$ only requires us to climb J, and K will automatically increase. Meanwhile, the incomplete log likelihood and the expected complete log likelihood share the same derivative at $\Psi=\Psi^{\text{old}}$, i.e.,

$$\frac{\partial}{\partial\Psi}\sum_{i=1}^{n}\log p(x_i;\Psi)\Big|_{\Psi=\Psi^{\text{old}}} = \frac{\partial}{\partial\Psi}J(\Psi,\Psi^{\text{old}})\Big|_{\Psi=\Psi^{\text{old}}}. \tag{5}$$

This is also known as the minorization-maximization algorithm (MM). A detailed explanation of the algorithm can be found in Lange et al. (2000). Our algorithm presented in the next section is essentially built on Lange et al. (2000) with variation made for the Lq-likelihood.

4.2 EM Algorithm with Lq-Likelihood

Having the idea of the traditional EM in mind, let us maximize the Lq-likelihood $\sum_{i=1}^{n} L_q(p(x_i;\Psi))$ in a similar fashion. For any two random variables X and Z, we have

$$L_q(p(X;\Psi)) = L_q\Big(\frac{p(X,Z;\Psi)}{p(Z|X;\Psi)}\Big) = \frac{L_q(p(X,Z;\Psi)) - L_q(p(Z|X;\Psi))}{p(Z|X;\Psi)^{1-q}},$$

where we have used $L_q(a/b) = [L_q(a) - L_q(b)]/b^{1-q}$ (Lemma 1, part (iii) in Section 9). Applying the above equation on data $x_1, ..., x_n$, and taking expectation (under Ψ^{old}) given observed data $x_1, ..., x_n$, we have

$$\sum_{i=1}^{n} E_{\Psi^{\text{old}}}\Big[L_q(p(X;\Psi))\Big|X = x_i\Big] = \sum_{i=1}^{n} E_{\Psi^{\text{old}}}\Big[\frac{L_q(p(X,Z;\Psi)) - L_q(p(Z|X;\Psi))}{p(Z|X;\Psi)^{1-q}}\Big|X = x_i\Big],$$

$$\sum_{i=1}^{n} L_q(p(x_i;\Psi)) = \sum_{i=1}^{n} E_{\Psi^{\text{old}}}\Big[\Big(\frac{p(Z|X;\Psi^{\text{old}})}{p(Z|X;\Psi)}\Big)^{1-q}\Big(\frac{L_q(p(X,Z;\Psi))}{p(Z|X;\Psi^{\text{old}})^{1-q}} - \frac{L_q(p(Z|X;\Psi))}{p(Z|X;\Psi^{\text{old}})^{1-q}}\Big)\Big|X = x_i\Big],$$

where we multiply and divide $P(Z|X,\Psi^{\text{old}})^{1-q}$ in the numerator and the denominator.

Define

$$\begin{aligned}
A(\Psi,\Psi^{\text{old}}) &= \sum_{i=1}^{n} E_{\Psi^{\text{old}}}\Big[\frac{L_q(p(X,Z;\Psi))}{p(Z|X;\Psi^{\text{old}})^{1-q}} - \frac{L_q(p(Z|X;\Psi))}{p(Z|X;\Psi^{\text{old}})^{1-q}}\Big|X = x_i\Big],\\
B(\Psi,\Psi^{\text{old}}) &= \sum_{i=1}^{n} E_{\Psi^{\text{old}}}\Big[\frac{L_q(p(X,Z;\Psi))}{p(Z|X;\Psi^{\text{old}})^{1-q}}\Big|X = x_i\Big],\\
C(\Psi,\Psi^{\text{old}}) &= -\sum_{i=1}^{n} E_{\Psi^{\text{old}}}\Big[\frac{L_q(p(Z|X;\Psi))}{p(Z|X;\Psi^{\text{old}})^{1-q}}\Big|X = x_i\Big],\\
\Rightarrow A(\Psi,\Psi^{\text{old}}) &= B(\Psi,\Psi^{\text{old}}) + C(\Psi,\Psi^{\text{old}}). \qquad (6)
\end{aligned}$$

Based on the definitions above, we have the following theorems.

Theorem 1. $C(\Psi,\Psi^{\text{old}})$ takes its minimum at $\Psi = \Psi^{\text{old}}$. i.e., $C(\Psi^{\text{old}},\Psi^{\text{old}}) = \min_{\Psi} C(\Psi,\Psi^{\text{old}})$.

Proof.

$$
\begin{aligned}
C(\Psi^{\text{old}}, \Psi^{\text{old}}) - C(\Psi, \Psi^{\text{old}}) &= \sum_{i=1}^{n} E_{\Psi^{\text{old}}}\Big[L_q\Big(\frac{p(Z|X;\Psi)}{p(Z|X;\Psi^{\text{old}})}\Big)\Big|X = x_i\Big] \\
&\leq \sum_{i=1}^{n} E_{\Psi^{\text{old}}}\Big[\frac{p(Z|X;\Psi)}{p(Z|X;\Psi^{\text{old}})} - 1\Big|X = x_i\Big] \\
&= \sum_{i=1}^{n}\sum_{z}\Big(\frac{p(z|x_i;\Psi)}{p(z|x_i;\Psi^{\text{old}})} - 1\Big)p(z|x_i;\Psi^{\text{old}}) = 0,
\end{aligned}
$$

where the inequality comes from the fact that $L_q(u) \leq u - 1$ (Lemma 1, part (iv) in Section 9). The above inequality becomes equality only when $\Psi = \Psi^{\text{old}}$. □

Theorem 2. When A, B and C are differentiable with respect to Ψ, we have

$$
\begin{aligned}
&\frac{\partial}{\partial\Psi}C(\Psi, \Psi^{\text{old}})\Big|_{\Psi=\Psi^{\text{old}}} = 0, \\
&\frac{\partial}{\partial\Psi}A(\Psi, \Psi^{\text{old}})\Big|_{\Psi=\Psi^{\text{old}}} = \frac{\partial}{\partial\Psi}B(\Psi, \Psi^{\text{old}})\Big|_{\Psi=\Psi^{\text{old}}}.
\end{aligned} \tag{7}
$$

Proof. The first part is a direct result from Theorem 1. By equation (6) and the first part of the theorem, we have the second part. □

Comparing equation (7) with equation (5), we can think of B as a proxy of the complete Lq-likelihood (i.e., J), A as a proxy of the incomplete Lq-likelihood, and C as a proxy of K.

We know that A is only an approximation of $\sum_{i=1}^{n} L_q(p(x_i;\Psi))$ due to the factor of $(p(Z|X;\Psi^{\text{old}})/p(Z|X;\Psi))^{1-q}$. However, at $\Psi = \Psi^{\text{old}}$, we do have

$$
A(\Psi, \Psi^{\text{old}})\big|_{\Psi=\Psi^{\text{old}}} = \sum_{i=1}^{n} L_q(p(x_i;\Psi))\big|_{\Psi=\Psi^{\text{old}}}. \tag{8}
$$

A will be a good approximation of $\sum_{i=1}^{n} L_q(p(x_i;\Psi))$ because that (1) within a small neighborhood $N_r(\Psi^{\text{old}}) = \{\Psi : d(\Psi, \Psi^{\text{old}}) < r\}$, $p(Z|X;\Psi^{\text{old}})/p(Z|X;\Psi)$ is approximately 1; (2) due to the transformation $y = x^{1-q}$, $(p(Z|X;\Psi^{\text{old}})/p(Z|X;\Psi))^{1-q}$ gets to be pushed

toward 1 even further when q is close to 1; and (3) even if $(p(Z|X;\Psi^{\text{old}})/p(Z|X;\Psi))^{1-q}$ is far from 1, because we sum over all the x_i's, we still average out these poorly approximated data points.

Given that C achieves minimum at Ψ^{old}, starting at Ψ^{old} and maximizing A requires only maximizing B. In order to take advantage of this property, we use A to approximate $\sum_{i=1}^{n} L_q(p(x_i;\Psi))$ at each iteration, and then maximize B to maximize A, and eventually to maximize $\sum_{i=1}^{n} L_q(p(x_i;\Psi))$. B is usually easy to maximize. Based on this idea, we build our EM-Lq as follows:

- 1. E step: Given Ψ^{old}, calculate B.
- 2. M step: Maximize B and obtain $\Psi^{\text{new}} = \arg\max_{\Psi} B(\Psi, \Psi^{\text{old}})$.
- 3. If Ψ^{new} converges, we terminate the algorithm. Otherwise, we set $\Psi^{\text{old}} = \Psi^{\text{new}}$, and return to step 1.

4.3 Monotonicity and Convergence

In this section, we will discuss the monotonicity and the convergence of the EM-Lq. We start with the following theorem.

Theorem 3. For any Ψ, we have the lower bound of the Lq-likelihood function

$$\sum_{i=1}^{n} L_q(p(x_i;\Psi)) \geq B(\Psi, \Psi^{\text{old}}) + C(\Psi^{\text{old}}, \Psi^{\text{old}}). \tag{9}$$

When $\Psi = \Psi^{\text{old}}$, we have

$$\sum_{i=1}^{n} L_q(p(x_i;\Psi^{\text{old}})) = B(\Psi^{\text{old}}, \Psi^{\text{old}}) + C(\Psi^{\text{old}}, \Psi^{\text{old}}).$$

Proof. See Section 9 for proof. □

From Theorem 3, we know that, at each M step, as long as we can find Ψ^{new} that increases B, i.e., $B(\Psi^{\text{new}}, \Psi^{\text{old}}) > B(\Psi^{\text{old}}, \Psi^{\text{old}})$, we can guarantee that the Lq-likelihood will also increase, i.e., $\sum_{i=1}^{n} L_q(x_i; \Psi^{\text{new}}) > \sum_{i=1}^{n} L_q(x_i; \Psi^{\text{old}})$. It is because that

$$\begin{aligned}\sum_{i=1}^{n} L_q(p(x_i; \Psi^{\text{new}})) &\geq B(\Psi^{\text{new}}, \Psi^{\text{old}}) + C(\Psi^{\text{old}}, \Psi^{\text{old}}) \\ &> B(\Psi^{\text{old}}, \Psi^{\text{old}}) + C(\Psi^{\text{old}}, \Psi^{\text{old}}) \\ &= \sum_{i=1}^{n} L_q(p(x_i; \Psi^{\text{old}})).\end{aligned}$$

Thus, we have proved the monotonicity of our EM-Lq algorithm.

Based on Theorem 3, we can further derive the following theorem.

Theorem 4. For our EM-Lq algorithm, when A, B and the Lq-likelihood are differentiable with respect to Ψ, it holds that

$$\begin{aligned}\frac{\partial}{\partial \Psi} \sum_{i=1}^{n} L_q(p(x_i; \Psi))\Big|_{\Psi=\Psi^{\text{old}}} &= \frac{\partial}{\partial \Psi} B(\Psi, \Psi^{\text{old}})\Big|_{\Psi=\Psi^{\text{old}}} = \frac{\partial}{\partial \Psi} A(\Psi, \Psi^{\text{old}})\Big|_{\Psi=\Psi^{\text{old}}}, \\ \sum_{i=1}^{n} L_q(P(x_i; \Psi))\Big|_{\Psi=\Psi^{\text{old}}} &= A(\Psi, \Psi^{\text{old}})\Big|_{\Psi=\Psi^{\text{old}}}.\end{aligned}$$

Proof. See Section 9 for proof. □

It becomes clear that A is not only just a good approximation of, but also the first order approximation of, $\sum_{i=1}^{n} L_q(p(x_i; \Psi))$.

One good thing following from the property of the first order approximation is that, when we have a fixed point, meaning that $A(\Psi^{\text{old}}, \Psi^{\text{old}}) = \max_{\Psi} A(\Psi, \Psi^{\text{old}})$, then we know $\frac{\partial}{\partial \Psi} A(\Psi, \Psi^{\text{old}})\Big|_{\Psi=\Psi^{\text{old}}} = \frac{\partial}{\partial \Psi} \sum_{i=1}^{n} L_q(p(x_i; \Psi))\Big|_{\Psi=\Psi^{\text{old}}} = 0$, which means that $\sum_{i=1}^{n} L_q(p(x_i; \Psi))$ takes its local maximum at the same place that $A(\Psi, \Psi)$ does. So as long as we achieve the maximum of A, we simultaneously maximize the incomplete Lq-likelihood $\sum_{i=1}^{n} L_q(p(x_i; \Psi))$.

By Theorem 4, we know that, as long as $\frac{\partial}{\partial \Psi} B(\Psi, \Psi^{\text{old}})\Big|_{\Psi=\Psi^{\text{old}}} \neq 0$, we can always find a Ψ^{new}, such that $\sum_{i=1}^{n} L_q(p(x_i; \Psi^{\text{new}})) > \sum_{i=1}^{n} L_q(p(x_i; \Psi^{\text{old}}))$. Hence our EM-L$q$ can be considered as a generalized EM algorithm (GEM) for Lq-likelihood. Wu (1983) has proved the convergence of the GEM from a pure optimization approach (Global Convergence Theorem, Theorem 1 and Theorem 2 of Wu, 1983, pp. 97 - 98), which we can directly use to prove the convergence of the EM-Lq.

In our simulation results, the converging point of the EM-Lq is always the same as the true maximizer of the Lq-likelihood which is obtained from the optimization package fmincon() in Matlab. We also try to move a small step away from the solution given by the EM-Lq to check whether the Lq-likelihood decreases. It shows that a small step in any directions will cause the Lq-likelihood to decrease, which numerically demonstrates that the solution is a local maximizer.

5 EM-Lq ALGORITHM FOR MIXTURE MODELS

5.1 EM-Lq for Mixture Models

Returning to our mixture model, suppose the observed data $x_1, ..., x_n$ are generated from a mixture model $f(x; \Psi) = \sum_{j=1}^{k} \pi_j f_j(x; \theta_j)$ with parameter $\Psi = (\pi_1, \ldots, \pi_{k-1}, \theta_1, \ldots, \theta_k)$. The missing data are the component labels $[\mathbf{z_1}, ..., \mathbf{z_n}]$, where $\mathbf{z_i} = (z_{i1}, ..., z_{ik})$ is a k dimensional component label vector with each element z_{ij} being 0 or 1 and $\sum_{j=1}^{k} z_{ij} = 1$.

In this situation, we have

$$p(x, \mathbf{z}; \Psi) = \prod_{j=1}^{k} (\pi_j f_j(x; \theta_j))^{z_j}, \tag{10}$$

$$p(\mathbf{z}|x; \Psi) = \prod_{j=1}^{k} p(z_j|x; \Psi)^{z_j} = \prod_{j=1}^{k} \Big(\frac{\pi_j f_j(x; \theta_j)}{f(x; \Psi)} \Big)^{z_j}, \tag{11}$$

where x is an observed data point, and $\mathbf{z} = (z_1, ..., z_k)$ is a component label vector. Substituting these into B and reorganizing the formula, we have

Theorem 5. In the mixture model case, B can be expressed as

$$B(\Psi, \Psi^{\text{old}}) = \sum_{i=1}^{n} \sum_{j=1}^{k} \tau_j(x_i, \Psi^{\text{old}})^q L_q(\pi_j f_j(x_i; \theta_j)),$$

where $\tau_j(x_i, \Psi^{\text{old}}) = E_{\Psi^{\text{old}}}[Z_{ij}|X = x_i]$, i.e., the soft label in the traditional EM.

Proof. See Section 9 for proof. □

We define new binary random variables $\tilde{Z}_{ij}$ whose expectation is $\tilde{\tau}_j(x_i, \Psi^{\text{old}}) = E_{\Psi^{\text{old}}}[\tilde{Z}_{ij}|X = x_i] = E_{\Psi^{\text{old}}}[Z_{ij}|X = x_i]^q$. $\tilde{Z}_{ij}$ can be considered as a distorted label as its probability distribution is distorted (i.e., $P_{\Psi^{\text{old}}}(\tilde{Z}_{ij} = 1|x_i) = P_{\Psi^{\text{old}}}(Z_{ij} = 1|x_i)^q$). Please note that, for $\tilde{Z}_{ij}$, we no longer have $\sum_{j=1}^{k} \tilde{\tau}_j(x_i, \Psi^{\text{old}}) = 1$. After the replacement, B becomes

$$B(\Psi, \Psi^{\text{old}}) = \sum_{i=1}^{n} \sum_{j=1}^{k} \tilde{\tau}_j(x_i, \Psi^{\text{old}}) L_q(\pi_j f_j(x_i; \theta_j)).$$

To maximize B, we apply the first order condition and obtain the following theorem.

Theorem 6. The first order condition of B with respect to θ_j and π_j yields

$$0 = \frac{\partial}{\partial \theta_j} B(\Psi, \Psi^{\text{old}}) \Rightarrow 0 = \sum_{i=1}^{n} \tilde{\tau}_j(x_i, \Psi^{\text{old}}) \frac{\frac{\partial}{\partial \theta_j} f_j(x_i; \theta_j)}{f_j(x_i; \theta_j)} f_j(x_i; \theta_j)^{1-q}, \tag{12}$$

$$0 = \frac{\partial}{\partial \pi_j} B(\Psi, \Psi^{\text{old}}) \Rightarrow \pi_j \propto \Big[\sum_{i=1}^{n} \tilde{\tau}_j(x_i, \Psi^{\text{old}}) f_j(x_i; \theta_j)^{1-q} \Big]^{\frac{1}{q}}. \tag{13}$$

Proof. See Section 9 for proof. □

Recall that the M step in the traditional EM solves a similar set of equations,

$$0 = \frac{\partial}{\partial \theta_j} J(\Psi, \Psi^{\text{old}}) \Rightarrow 0 = \sum_{i=1}^{n} \tau_j(x_i, \Psi^{\text{old}}) \frac{\frac{\partial}{\partial \theta_j} f_j(x_i; \theta_j)}{f_j(x_i; \theta_j)}, \tag{14}$$

$$0 = \frac{\partial}{\partial \pi_j} J(\Psi, \Psi^{\text{old}}) \Rightarrow \pi_j \propto \sum_{i=1}^{n} \tau_j(x_i, \Psi^{\text{old}}). \tag{15}$$

Comparing equations (14) and (15) with equations (12) and (13), we see that (1) θ_j^{new} of the EM-Lq satisfies a weighted likelihood equation, where the weights contain both the distorted soft label $\tilde{\tau}_j(x_i, \Psi^{\text{old}})$ and the power transformation of the individual component density function, $f_j(x_i; \theta_j)^{1-q}$; and (2) π_j is proportional to the summation of the distorted soft label $\tilde{\tau}_j(x_i, \Psi^{\text{old}})$ adjusted by the individual density function.

5.2 EM-Lq for Gaussian Mixture Models

For a Gaussian mixture model with parameter $\Psi = (\pi_1, \ldots, \pi_{k-1}, \mu_1, \ldots, \mu_k, \sigma_1^2, \ldots, \sigma_k^2)$. At each E step, we calculate $\tilde{\tau}_j(x_i, \Psi^{\text{old}}) = \left[\frac{\pi_j^{\text{old}} \varphi(x_i; \mu_j^{\text{old}}, \sigma_j^{2\text{old}})}{f(x_i, \Psi_{\text{old}})}\right]^q$. At each M step, we solve equations (12) and (13) to yield

$$\mu_j^{\text{new}} = \frac{1}{\sum_{i=1}^{n} \tilde{w}_{ij}} \sum_{i=1}^{n} \tilde{w}_{ij} x_i, \tag{16}$$

$$\sigma_j^{2\text{new}} = \frac{1}{\sum_{i=1}^{n} \tilde{w}_{ij}} \sum_{i=1}^{n} \tilde{w}_{ij} (x_i - \mu_j^{\text{new}})^2, \tag{17}$$

$$\pi_j^{\text{new}} \propto \left[\sum_{i=1}^{n} \tilde{\tau}_j(x_i, \Psi^{\text{old}}) \varphi(x_i; \mu_j^{\text{new}}, \sigma_j^{2\text{new}})^{1-q}\right]^{\frac{1}{q}},$$

where $\tilde{w}_{ij} = \tilde{\tau}_j(x_i, \Psi^{\text{old}}) \varphi(x_i; \mu_j^{\text{new}}, \sigma_j^{2\text{new}})^{1-q}$. The same iterative re-weighting algorithm designed for solving equations (2) and (3) can be used to solve equations (16) and (17). Details of the algorithm is shown in Section 9.

At each M step, it is feasible to replace $\tilde{w}_{ij}$ with $\tilde{w}_{ij}^* = \tilde{\tau}_j(x_i, \Psi^{\text{old}}) \varphi(x_i; \mu_j^{\text{old}}, \sigma_j^{2\text{old}})^{1-q}$,

which only depends on the Ψ^{old}, to improve the efficiency of the algorithm. Thus we can avoid the re-weighting algorithm at each M step. This replacement will simplify the EM-Lq algorithm significantly. We have done the simulation to show that this modified version of the algorithm also gives same solutions as the original EM-Lq algorithm.

5.3 Convergence Speed

We have compared the convergence speeds of the EM-Lq and the EM algorithm using a Gaussian Mixture Model with complexity of 2 (2GMM), $f(x) = 0.4\varphi(x; 1, 2) + 0.6\varphi(x; 5, 2)$, whose two components are in-separable because of the overlap. Surprisingly, the convergence of the EM-Lq is on average slightly faster than that of the EM.

The comparison of the convergence speed is based on r, which is defined as

$$r = \frac{\|\Psi^{(k)} - \Psi^{(k-1)}\|}{\|\Psi^{(k-1)} - \Psi^{(k-2)}\|},$$

where k is the last iteration of the EM-Lq or the EM algorithm. The smaller r is, the faster the convergence is.

We simulate 1000 data sets according to the 2GMM, use the EM-Lq ($q = 0.8$) and the EM to fit the data, and record the convergence speed difference $r_{\text{ML}q\text{E}} - r_{\text{MLE}}$. The average convergence speed difference is -0.012 with a standard error of 0.002, which means the negative difference in the convergence speed is statistically significant.

However, if we change the 2GMM to a gross error model of 3GMM: $f(x) = 0.4(1 - \epsilon)\varphi(x; 1, 2) + 0.6(1 - \epsilon)\varphi(x; 5, 2) + \epsilon\varphi(x; 3, 40)$, where the third component is an outlier component, and still use a 2GMM to fit, the comparison of the convergence speed becomes unclear. We have not fully understood the properties of the convergence speed for the EM-Lq yet. However, we do believe the convergence speed is important, and is an interesting topic for future research.

The fact that the convergence of the EM-Lq is a little faster than that of the EM is closely related to the concept of the information ratio mentioned in Redner and Walker (1984) and Windham and Cutler (1992), where the convergence speed is connected to the missing information ratio. In Lq-likelihood, since the two in-separable components are pushed apart by the weights $\tilde{w}_{ij}$, the corresponding concept of the missing information ratio for the Lq-likelihood is relatively lower, thus, we have a faster convergence.

Although the convergence speed is faster for the EM-Lq, it is not necessary that the EM-Lq takes less computer time than the EM. This is because that, at each M step in the EM-Lq, we need to do another iterative algorithm to obtain Ψ^{new} (i.e., the algorithm explained in Section 9), whereas the EM needs only one step to obtain the new parameter estimate.

The advantage of the convergence speed of the EM-Lq has been hinted by another algorithm called q-Parameterized Deterministic Annealing EM algorithm (q-DAEM) previously proposed by Guo and Cui (2008) in the signal processing and statistical mechanics context. The q-DAEM can successfully maximize the log-likelihood at a faster convergence speed, by using a different but similar M steps as in our EM-Lq. Their M step includes setting $q > 1$ and $\beta > 1$ and dynamically pushing $q \to 1$ and $\beta \to 1$ (β is an additional parameter for their deterministic annealing procedure). On the other hand, our EM-Lq maximizes the Lq-likelihood with a fixed $q < 1$. Although the objective functions are different for these two algorithms, it is obvious that the advantages on the convergence speed are due to the tuning parameter q. It turns out that $q > 1$(along with $\beta > 1$ in the q-DAEM) and $q < 1$ (in the EM-Lq) both help with the convergence speed, even though they have different convergence points. We have proved the first order approximation property in Theorem 4, which leads to the proof of the monotonicity and the convergence for the EM-Lq. For the q-DAEM, because $q \downarrow 1$ and $\beta \downarrow 1$ make it reduce to the traditional EM, it also converges. When $\beta = 1$ and $q = 1$, both algorithms reduce to the traditional EM algorithm.

6 NUMERICAL RESULTS AND VALIDATION

Now we compare the performance of two estimators on mixture models: 1) the MLqE from the EM-Lq; 2) the MLE from the EM. We set $q = 0.95$ throughout this section.

6.1 Kullback Leibler Distance Comparison

We simulate data using a three component Gaussian mixture model (3GMM)

$$f_0^*(x; \epsilon, \sigma_c^2) = 0.4(1-\epsilon)\varphi(x; 1, 2) + 0.6(1-\epsilon)\varphi(x; 5, 2) + \epsilon\varphi(x; 3, \sigma_c^2). \tag{18}$$

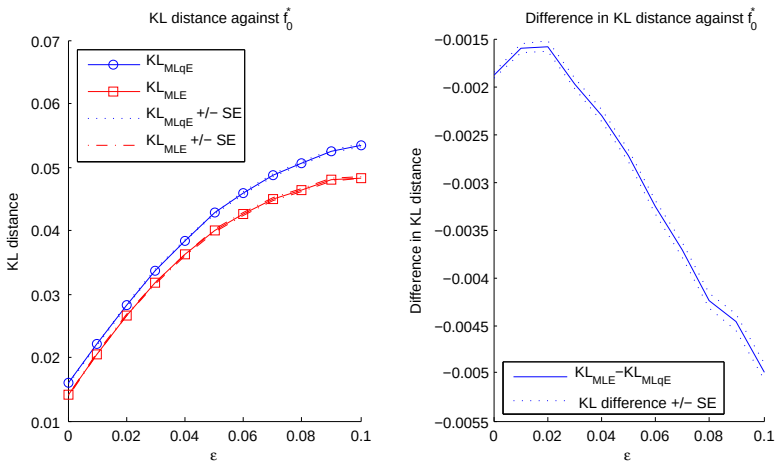
This is a gross error model, where the third term is the outlier component (or contamination component, or measurement error component); ϵ is the contamination ratio ($\epsilon \leq 0.1$); σ_c^2 is the variance of the contamination component, and is usually very large (i.e., $\sigma_c^2 > 10$). Equation (18) can be considered as a small deviation from the 2GMM: $f_0(x) = 0.4\varphi(x; 1, 2) + 0.6\varphi(x; 5, 2)$.

As we mentioned in Section 3, there are two approaches for estimating f_0 based on data generated by f_0^*. We will investigate them individually.

6.1.1 Direct Approach

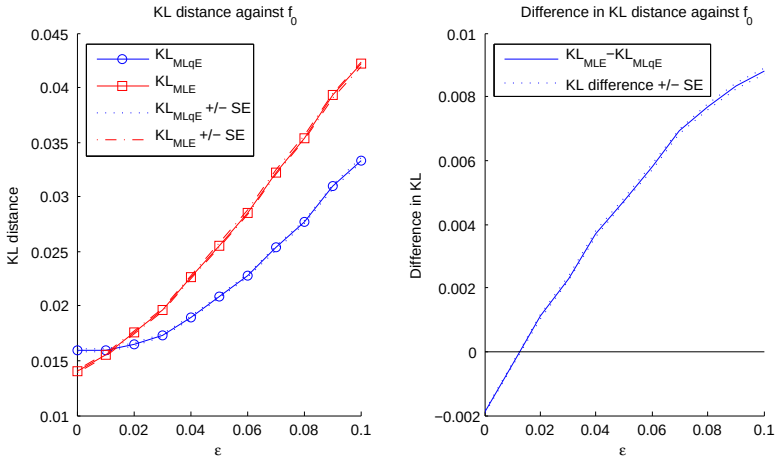
We start with the direct approach. First, we simulate data with sample size $n = 200$ according to equation (18), $f_0^*(x; \epsilon, \sigma_c^2 = 20)$, at different contamination levels $\epsilon \in [0, 0.1]$. We fit the 2GMM using the MLqE and the MLE. We repeat this procedure 10,000 times and then calculate (1) the average KL distance between the estimated 2GMM and f_0^*, and (2) the average KL distance between the estimated 2GMM and f_0. We summarize the results in Figure 2 (KL against f_0^*) and Figure 3 (KL against f_0).

In Figure 2a, we see that both $\mathrm{KL}_{\mathrm{ML}q\mathrm{E}}$ and $\mathrm{KL}_{\mathrm{MLE}}$ increase as ϵ increases, which means the performance of both MLqE and MLE degrades as more measurement errors are present.



(a) (b)

Figure 2: Comparison between the MLqE and the MLE in terms of KL distances against f_0^*: (a) shows the KL distances themselves, (b) shows their difference.



(a) (b)

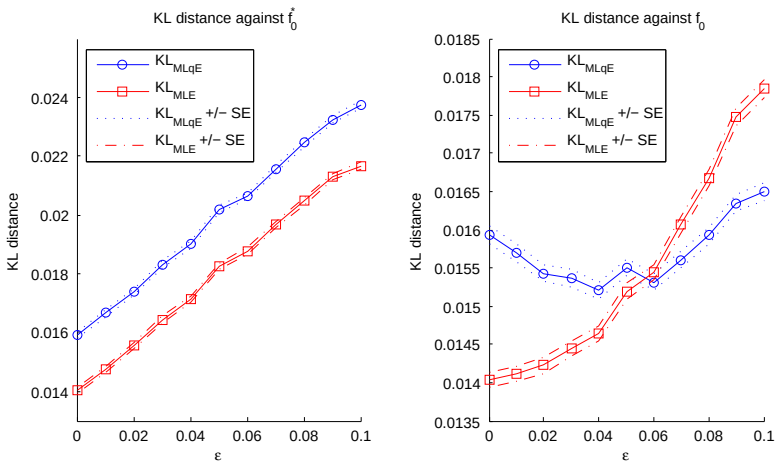
Figure 3: Comparison between the MLqE and the MLE in terms of KL distances against f_0: (a) shows the KL distances themselves, (b) shows their difference.

$\mathrm{KL}_{\mathrm{ML}q\mathrm{E}}$ is always larger than and increases slightly faster than $\mathrm{KL}_{\mathrm{MLE}}$. It implies that the ML$q$E performs worse than, and degrades faster than the MLE. Figure 2b shows their difference $\mathrm{KL}_{\mathrm{MLE}} - \mathrm{KL}_{\mathrm{ML}q\mathrm{E}}$ which is negative and decreasing. This phenomena is reasonable because, when estimating f_0^* using data generated by f_0^*, the MLE is the best estimator (in terms of KL distance) by definition. The MLqE's bias-variance trade off does not gain anything compared to the MLE.

On the other hand, Figure 3 shows an interesting phenomena. In Figure 3a, we see that both $\mathrm{KL}_{\mathrm{ML}q\mathrm{E}}$ and $\mathrm{KL}_{\mathrm{MLE}}$ still increase as ϵ increases. However, when estimating the non-measurement error components f_0, $\mathrm{KL}_{\mathrm{ML}q\mathrm{E}}$ increases more slowly than $\mathrm{KL}_{\mathrm{MLE}}$. The former starts above the latter but eventually ends up below the latter as ϵ increases, which means the MLE degrades faster than the MLqE. Figure 3b shows their difference $\mathrm{KL}_{\mathrm{MLE}} - \mathrm{KL}_{\mathrm{ML}q\mathrm{E}}$ which starts in negative and increases gradually to positive (changes sign at around $\epsilon = 0.025$). This means that our MLqE performs better than the MLE in terms of estimating f_0 when there are more measurement errors in the data. Hence, we gain robustness from the MLqE.

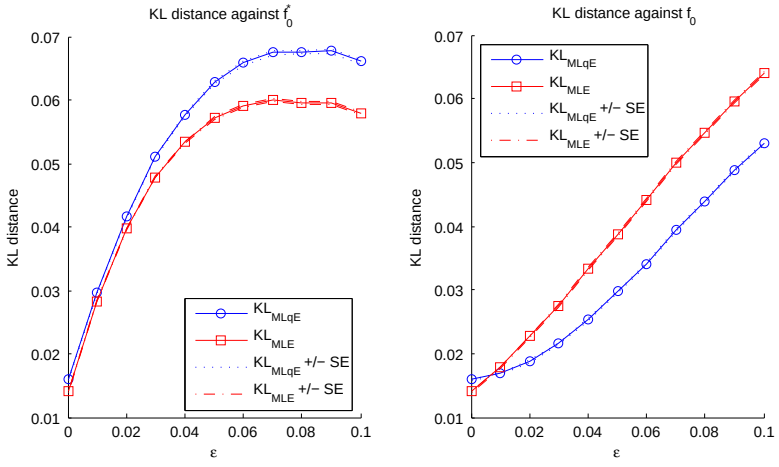
The above simulation is done using the model $f_0^*(x; \epsilon, \sigma_{\mathrm{c}}^2 = 20)$. To illustrate the effect of σ_{c}^2 on the performance of the MLqE, we change model to $f_0^*(x; \epsilon, \sigma_{\mathrm{c}}^2 = 10)$ and $f_0^*(x; \epsilon, \sigma_{\mathrm{c}}^2 = 30)$, and repeat the above calculation. The results are shown in Figures 4 and 5.

As we can see, σ_{c}^2 has a big impact on the performance of the estimator. As σ_{c}^2 gets larger (i.e., more serious measurement error problems), both the MLqE and the MLE degrade faster as the contamination ratio increases. This is why the slopes of the KL distance curves become steeper with the higher σ_{c}^2. However, the advantage of the MLqE over the MLE is more obvious with the larger σ_{c}^2. The point where two KL distance curves insect (in Figure 4b and 5b) moves to the left as σ_{c}^2 increases, which means the MLqE will beat the MLE at the lower contamination ratio when the higher σ_{c}^2 is used (i.e., the higher variance of the measurement errors).



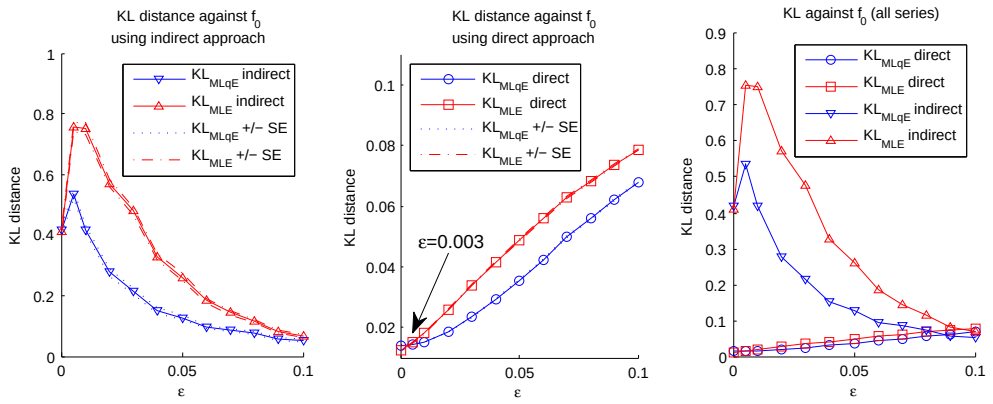
(a) (b)

Figure 4: Comparison between the MLqE and the MLE in terms of KL distances against f_0^* (left panel) and f_0 (right panel) with the third component variance σ_c^2 being 10.



(a) (b)

Figure 5: Comparison between the MLqE and the MLE in terms of KL distances against f_0^* (left panel) and f_0 (right panel) with the third component variance σ_c^2 being 30.



(a) (b) (c)

Figure 6: Comparison between the MLqE and the MLE in terms of KL distances against f_0: (a) shows KL distances obtained from the indirect approach, (b) shows KL distances obtained from the direct approach, (c) shows both these two kinds of KL distances together in order to compare their magnitude.

6.1.2 Indirect Approach

Now, let us take the indirect approach, which is to estimate f_0^* first and project it onto the 2GMM space. In this experiment, we let the data to be generated by $f_0^*(x; \epsilon, \sigma_c^2 = 40)$ which has an even higher variance of the measurement error component than the previous section. We use a sample size of $n = 200$. We simulate data according to $f_0^*(x; \epsilon, \sigma_c^2 = 40)$, use the ML$q$E and the MLE to fit the 3GMM, take out its component with the largest variance and normalize the weights to get our estimate for f_0. We repeat this procedure 10,000 times, and calculate the average KL distance between our estimates (both the MLqE and the MLE) and f_0. For the comparison purpose, we repeat the calculation using the direct approach on this simulation data as well, and summarize the results in Figure 6.

In Figure 6a, we see that, as ϵ increases, KL distances of the indirect approach first increase and then decrease. The increasing part suggests that a few outliers will hurt the estimation of the non-measurement error component. The decreasing part means that, after the contamination increases beyond certain level ($\epsilon = 0.5\%$), the more contamination there

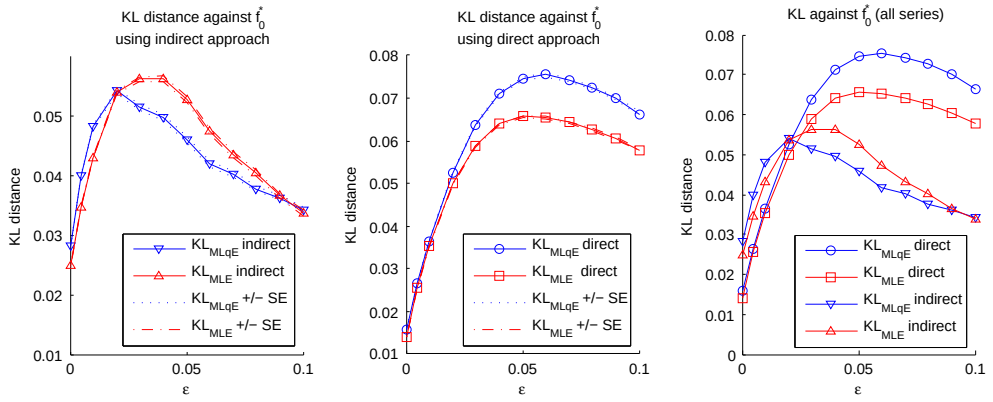
is, the more accurate our estimates are. This is because that, when the contamination ratio is small, it is hard to estimate the measurement error component as there are very few outliers. As the contamination ratio gets larger, the indirect approach can more accurately estimate the measurement error component, hence provide better estimates of the non-measurement error components. Please note that our MLqE is still doing better than the MLE in this case. The reason is that the MLqE successfully trades bias for variance to gain in the overall performance. However, as ϵ increases, the advantage of the MLqE gradually disappears. It is because that when the contamination is obvious, the MLE will be more powerful and efficient than the MLqE under the correctly specified model.

In Figure 6b, we present the results for the direct approach, which is consistent with Figure 3a. We notice that, when f_0^* has a larger variance for the measure error component, the MLqE beats the MLE at a lower contamination ratio ($\epsilon = 0.003$). In other words, as f_0^* is further deviated from f_0 (in terms of the variance of measurement error component), the advantage of the MLqE becomes more significant.

In Figure 6c, we plot KL distances of both approaches. It is obvious that the indirect approach is not even comparable to the direct approach until ϵ raises above 0.08. It is because that we estimate more parameters and have more estimation variance for the indirect approach. Although our model is correctly specified, the estimation variance is so big that it dominates the overall performance. To sum up, with the small contamination ratio, we would be better off using the direct approach with the misspecified model. When the contamination ratio is large, we should use the indirect approach with the correctly specified model.

The above comparison is done based on the KL distance against f_0. We repeat the above calculation to obtain the corresponding results for the KL distance against f_0^*. Note that all the calculation is the same except we do not need to do the projection from 3GMM to 2GMM, because f_0^* is 3GMM. The results are shown in Figure 7.

As we can see from Figure 7a, when the contamination ratio increases, the KL distances



(a) (b) (c)

Figure 7: Comparison between the MLqE and the MLE in terms of KL distances against f_0^*: (a) shows KL distances obtained from the indirect approach, (b) shows KL distances obtained from the direct approach, (c) shows both these two kinds of KL distances together in order to compare their magnitude.

against f_0^* (for both the MLqE and the MLE) increase first and then decrease. It means that as outliers are gradually brought into the data, they first undermine the estimation for the non-measurement error components, and then help the estimation of the measurement error component. The MLqE starts slightly above the MLE. When outliers become helpful for the estimation ($\epsilon > 2\%$), the MLqE goes below the MLE. As ϵ increases beyond 2%, the advantage of the MLqE over the MLE first increases and then diminishes. Figure 7b is also consistent with what we found in Figures 2a and 4a and 5a. In Figure 7c, we see that the direct and indirect approaches are in about the same range. They intersect at around $\epsilon = 2\%$, which suggests that, when estimating f_0^*, we prefer the direct approach for the mildly contaminated data, and prefer the indirect approach for the heavily contaminated data.

6.2 Relative Efficiency

We can also compute the relative efficiency between the MLE and the MLqE using the same model (equation (18)), $f_0^*(x; \epsilon, \sigma_c^2 = 20)$.

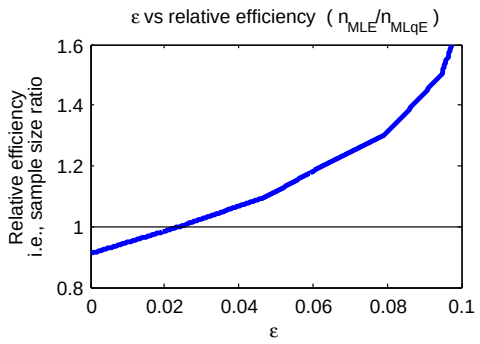


Figure 8: Comparison of the MLE and the MLqE based on relative efficiency.

At each level $\epsilon \in [0, 0.1]$, we generate 3,000 samples with sample size $n = 100$ according to equation (18), $f_0^*(x; \epsilon, \sigma_c^2 = 20)$, fit the 2GMM to the data using the MLqE, and calculate the average KL against f_0. We try the same procedure for the MLE, and find the sample size $n_{\text{MLE}}(\epsilon)$ at which the same average KL is obtained by the MLE. We plot the ratio of these two sample sizes $n_{\text{MLE}}(\epsilon)/100$ in Figure 8.

As we can see, the relative efficiency starts below 1, which means, when the contamination ratio is small, it takes the MLE fewer samples than the MLqE to achieve the same performance. However, as the contamination ratio increases, the relative efficiency climbs substantially above 1, meaning that the MLE will need more data than the MLqE to achieve the same performance.

6.3 Gamma Chi-Square Mixture Model

We take a small digression and consider estimating a Gamma Chi-square mixture model,

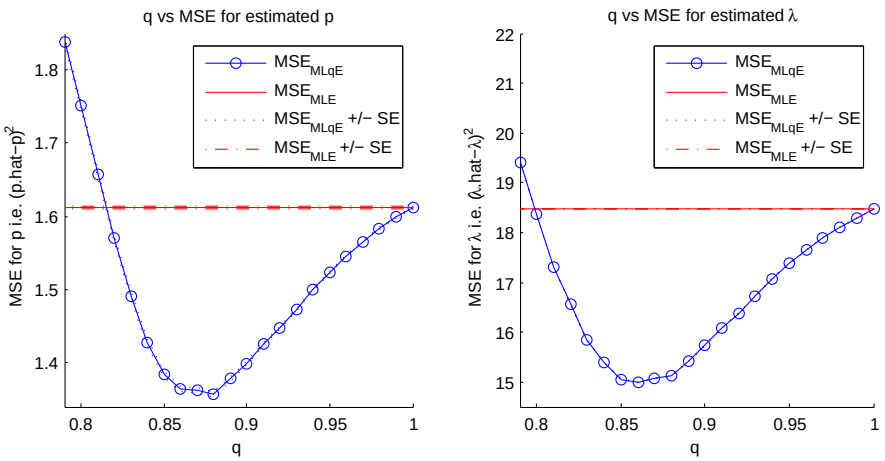
$$f_0^*(x) = (1 - \epsilon)\text{Gamma}(x; p, \lambda) + \epsilon\chi^2(x; d), \tag{19}$$

where the second component is the measurement error component. We can think of our data being generated from the Gamma distribution but contaminated with the Chi-square gross error. In this section, we consider two scenarios:

Scenario 1: $p = 2$, $\lambda = 5$, $d = 5$, $\epsilon = 0.2$, $n = 20$

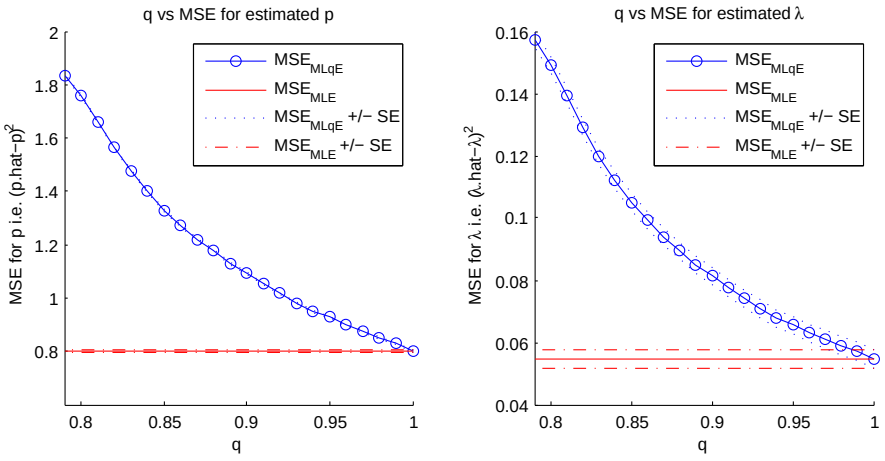
Scenario 2: $p = 2$, $\lambda = 0.5$, $d = 5$, $\epsilon = 0.2$, $n = 20$

In each scenario, we generate 50,000 samples according to equation (19), fit the Gamma distribution using both the MLqE and the MLE, and compare these two estimators based on their mean square error (MSE) for p and λ. For the MLqE, we adjust q to examine the effect of the bias-variance trade off. The results are summarized in Figure 9 (scenario 1) and Figure 10 (scenario 2). In Figure 9, We see that, by setting $q < 1$, we can successfully trade bias for variance and obtain better estimation. In scenario 1, since the Gamma distribution and the Chi-square distribution are sharply different, the bias-variance trade off leads to a significant reduction on the mean square error by partially ignoring the outliers. However, in scenario 2, these two distributions are similar (the mean and variance of the Gamma distribution are 4 and 8, the mean and variance of the Chi-square distribution are 5 and 10). In this situation, partially ignoring the data points on the tails will not help much, which is why the MSE of the MLqE is always larger than the MSE of the MLE.



(a) (b)

Figure 9: Comparison of the MLE and the MLqE in terms of the MSE for $\hat{p}$ (Figure a) and $\hat{\lambda}$ (Figure b) in scenario 1 ($p = 2$, $\lambda = 5$, $d = 5$, $\epsilon = 0.2$, $n = 20$).



(a) (b)

Figure 10: Comparison of the MLE and the MLqE in terms of the MSE for $\hat{p}$ (Figure a) and $\hat{\lambda}$ (Figure b) in scenario 2 ($p = 2$, $\lambda = 0.5$, $d = 5$, $\epsilon = 0.2$, $n = 20$).

6.4 Old Faithful Geyser Eruption Data

We consider the Old Faithful geyser eruption data from Silverman (1986). The original data is obtained from the R package "tclust". The data is univariate eruption time length with sample size of 272. We sort these eruption lengths by their times of occurrences, and lag these lengths by one occurrence to form 271 pairs; thus we have two dimensional data (i.e., current eruption length and previous eruption length). This is the same procedure as described in Garcia-Escudero and Gordaliza (1999). For this two dimensional data, they have suggested three clusters. Since the "short followed by short" eruptions are not usual, Garcia-Escudero and Gordaliza (1999) identify these points in the lower left corner as outliers.

We plot the original data in Figure 11, fit the MLqE ($q = 0.8$) and the MLE to the data, and plot the 2 standard deviation ellipsoids. q is selected based on clustering outcome. As we can see, there are a few outliers in the lower left corner. The MLE is obviously affected by the outliers. The lower right component of the MLE is dragged to the left to accommodate these outliers, and thus misses the center of the cluster. Other components of the MLE are also mildly affected. The MLqE, on the other hand, overcomes this difficulty and correctly

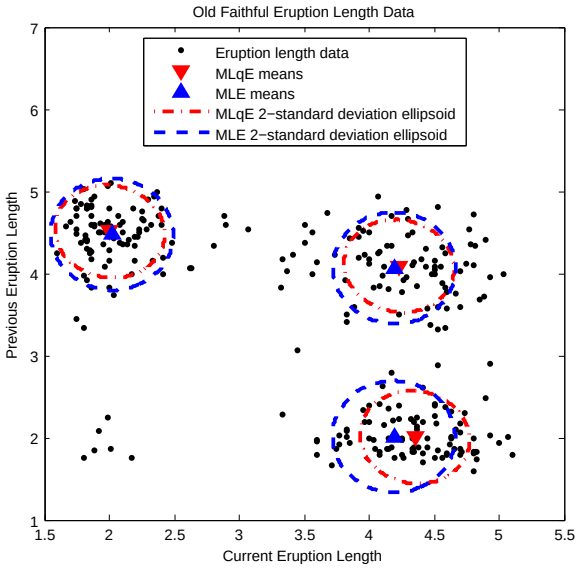


Figure 11: Comparison between the MLqE and the MLE for the Old Faithful geyser data: red triangles: MLqE means; red dashed lines: MLqE two standard deviation ellipsoids; blue triangles: MLE means; blue dashed lines: MLE two standard deviation ellipsoids.

identifies the center of each component. This improvement is especially obvious for the lower right component: the fitted MLqE lies in the center whereas the MLE is shifted to the left and has a bigger 2 standard deviation ellipsoid.

7 SELECTION OF q

So far in this article, we have fixed q in all the analysis. In this section, we will discuss about the selection of q.

The tuning parameter q governs the sensitivity of the estimator against outliers. The smaller q is, the less sensitive the MLqE is to outliers. If the contamination becomes more serious (i.e., larger ϵ and/or σ_c^2), we should use a smaller q to protect against measurement errors. There is no analytical relation between the level of contamination and q, because it depends on the properties of the non-measurement error components, the contamination ratio and the variance of the contamination component. Furthermore, there is no guarantee

that the measurement error component is a Normal distribution. Since all these assumptions can be easily violated, it is impossible to establish an analytical relationship for q and the contamination level.

When $q \neq 1$, the MLqE is an inconsistent estimator. Ferrari and Yang (2010) let $q \to 1$ as $n \to \infty$ in order to force the consistency. In our case, we allow the MLqE to be inconsistent because our data is contaminated. We are no longer after the true underlying distribution f_0^* that generates the data, but are more interested in estimating the non-measurement error components f_0 using the contaminated data. Since the goal is not to estimate f_0^*, being consistent will not help the estimator in terms of robustness.

Generally, it is very hard to choose q analytically. Currently, there is no universal way to do so. Instead, we here present an example to illustrate the idea of selecting q. We generate one data set using equation (18) $f_0^*(x; \epsilon = 0.1, \sigma_c^2 = 40)$ with the sample size $n = 200$. We will demonstrate how to select q for this particular data set.

First, we fit a 3GMM $\hat{f}_{3\text{GMM}}$ to the data using the MLE. We identify the component with the largest variance in $\hat{f}_{3\text{GMM}}$ as the contamination component. We extract the non-measurement error components and renormalize weights to get $\hat{f}_{3\text{GMM}\to 2\text{GMM}}$, which can be considered as the projection from 3GMM to the 2GMM space. We go back to $\hat{f}_{3\text{GMM}}$, utilize it to perform a parametric bootstrap by generating many bootstrap samples, and fit 2GMM to these data sets using MLqE ($\hat{f}^{b\,\text{ML}q\text{E}}_{2\text{GMM}}$) with q varying between 0.7 and 1. We take the q that minimizes the average KL distance between $\hat{f}_{3\text{GMM}\to 2\text{GMM}}$ and the estimated 2GMM $\hat{f}^{b\,\text{ML}q\text{E}}_{2\text{GMM}}$ from the bootstrap samples. The average KL distance against q is shown in Figure 12. From the figure, we estimate q to be 0.82.

This is a very simple way to select q. It is straightforward and easy. However, there is a drawback of this method. When the contamination ratio is very low (e.g., 1% or 2%) and the sample size is small ($n < 100$), the estimated 3GMM $\hat{f}_{3\text{GMM}}$ will not be able to estimate the measurement error component correctly since there are very few outliers. Thus, the

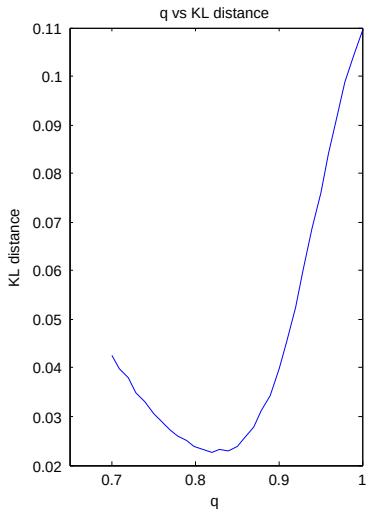


Figure 12: Selection of q based on average KL distance from the bootstrap samples.

parametric bootstrap approach following that will become unreliable. We have not found an effective way of selecting q with the small contamination ratio.

In Ferrari and Yang (2010), they have mentioned using asymptotic variance and asymptotic efficiency as criteria for selecting q. However, obtaining asymptotic variance in the mixture model case is also problematic and unreliable when sample size is small.

To obtain an analytical solution for q is hard. Currently, we have only some remedies under a few situations, and are still looking for a universal way. However, we believe that selecting q is a very important question and is one of major future research directions.

8 DISCUSSION

In this article, we have introduced a new estimation procedure for mixture models, namely the MLqE, along with the EM-Lq algorithm. Our new algorithm provides a more robust estimation for mixture models when measurement errors are present in the data. Simulation results show superior performance of the MLqE over the MLE in terms of estimating the non-measurement error components. Relative efficiency is also studied and shows superiority

of the MLqE. Note that when $q = 1$, the MLqE becomes the MLE, so the MLqE can be considered as a generalization of the MLE.

Throughout this article, we see that the MLqE works well with mixture models in the EM framework. There is a fundamental reason for such a phenomena. Note that the M step of the traditional EM solves a set of weighted likelihood equations with weights being the soft labels. Meanwhile, the MLqE solves a different set of weighted likelihood equations with weights being f^{1-q}. Therefore, incorporating the MLqE in the EM framework comes down to determining the new weights that are consistent with both the soft labels and f^{1-q}. Furthermore, we conjecture that, for any new types of estimators, as long as they only involve solving sets of weighted likelihood equations, they should be able to be smoothly incorporated in the mixture model estimation using the EM framework.

In order to achieve consistency for the MLqE, we need the distortion parameter q to approach 1 as the sample size n goes to infinity. However, letting q converge to 1 will affect the bias-variance trade off. So what is the optimal rate at which q tends to 1 as $n \to \infty$? Meanwhile, how to select q at different sample size is also an interesting topic. The distortion parameter q adjusts how aggressive or conservative we are towards eliminating the effect of outliers. Tuning of the distortion parameter q will be a fruitful direction for future research.

9 APPENDIX

Lemma 1. $\forall m \in \mathbb{R}$ and $\forall a, b \in \mathbb{R}^+$, it holds that

(i) $L_q(ab) = L_q(a) + L_q(b) + (1-q)L_q(a)L_q(b) = L_q(a) + a^{1-q}L_q(b)$.

(ii) $L_q(a^m) = L_q(a)\frac{1-(a^{1-q})^m}{1-a^{1-q}}$.

(iii) $L_q\big(\frac{a}{b}\big) = (\frac{1}{b})^{1-q}(L_q(a) - L_q(b))$.

(iv) $L_q(a)$ is a concave function and $L_q(a) \leq a - 1$.

Proof. (i) We know that $L_q(ab) = \frac{(a^{1-q}-1)+(b^{1-q}-1)+(a^{1-q}-1)(b^{1-q}-1)}{1-q}$, which proves (i).

(ii) $L_q(a^m) = \frac{a^{1-q}-1}{1-q}\frac{(a^{1-q})^m-1}{a^{1-q}-1} = L_q(a)\frac{1-(a^{1-q})^m}{1-a^{1-q}}$.

(iii) By (i), we have $L_q(a/b) = L_q(a)/b^{1-q} + L_q(1/b) = [L_q(a) - L_q(b)]/b^{1-q}$.

(iv) We have $\partial^2 L_q(a)/\partial a^2 = -qa^{-q-1} < 0$, hence, $L_q(a)$ is concave. By the mean value theorem of concave function: $L_q(a) - L_q(1) \leq (a-1)\frac{\partial L_q(x)}{\partial x}\Big|_{x=1} \quad \Rightarrow \quad L_q(a) \leq a - 1.$ $\square$

Re-weighting Algorithm for MLqE: The re-weighting algorithm for solving the MLqE in general is described as follows.

To obtain $\hat{\theta}_{\mathrm{ML}q\mathrm{E}}$, we start with an initial estimate $\theta^{(1)}$ which could be any sensible estimate, we usually use $\hat{\theta}_{\mathrm{MLE}}$ as the starting point. For each new iteration t ($t > 1$), $\theta^{(t+1)}$ is computed via

$$\theta^{(t+1)} = \left\{\theta : 0 = \sum_{i=1}^{n} U(x_i;\theta) f(x_i;\theta^{(t)})^{1-q}\right\},$$

where $U(x;\theta) = \nabla_\theta \log f(x;\theta) = f'_\theta(x;\theta)/f(x;\theta)$. The algorithm is stopped when a certain convergence criterion is satisfied, for example, the change in $\theta^{(t)}$ is sufficiently small.

To obtain the MLqE for a normal distribution, the above algorithm is simplified as follows:

$$\hat{\mu}^{(t+1)} = \frac{1}{\sum_{i=1}^{n} w_i^{(t)}} \sum_{i=1}^{n} w_i^{(t)} x_i,$$
$$\hat{\sigma^2}^{(t+1)} = \frac{1}{\sum_{i=1}^{n} w_i^{(t)}} \sum_{i=1}^{n} w_i^{(t)} (x_i - \hat{\mu}^{(t+1)})^2,$$

where $w_i^{(t)} = \varphi(x_i;\hat{\mu}^{(t)},\hat{\sigma^2}^{(t)})^{1-q}$ and φ is a normal probability density function.

In the M step of the EM-Lq algorithm, the above algorithm is further modified as follows:

$$\mu_j^{(t+1)} = \frac{1}{\sum_{i=1}^{n} \tilde{w}_{ij}^{(t)}} \sum_{i=1}^{n} \tilde{w}_{ij}^{(t)} x_i,$$
$$\sigma_j^{2(t+1)} = \frac{1}{\sum_{i=1}^{n} \tilde{w}_{ij}^{(t)}} \sum_{i=1}^{n} \tilde{w}_{ij}^{(t)} (x_i - \mu_j^{(t+1)})^2,$$

where $\tilde{w}_{ij}^{(t)} = \tilde{\tau}_j(x_i, \Psi^{\text{old}})\varphi(x_i; \mu_j^{(t)}, \sigma_j^{2(t)})^{1-q}$. We iterate the above calculation until $\mu_j^{(t)}$ and $\sigma_j^{2(t)}$ converge, and assign them to μ_j^{new} and $\sigma_j^{2\text{new}}$.

Proof of Theorem 3:

$$\begin{aligned}
\sum_{i=1}^{n} L_q(p(x_i;\Psi)) &= \sum_{i=1}^{n} L_q(\sum_z p(x_i, z; \Psi)) \\
&= \sum_{i=1}^{n} L_q(\sum_z p(z|x_i;\Psi^{\text{old}})\frac{p(x_i,z;\Psi)}{p(z|x_i;\Psi^{\text{old}})}) \qquad (20)\\
&\geq \sum_{i=1}^{n}\sum_z p(z|x_i;\Psi^{\text{old}}) L_q\Big(\frac{p(x_i,z;\Psi)}{p(z|x_i;\Psi^{\text{old}})}\Big) \qquad (21)\\
&= \sum_{i=1}^{n}\sum_z p(z|x_i;\Psi^{\text{old}})\frac{L_q(p(x_i,z;\Psi)) - L_q(p(z|x_i;\Psi^{\text{old}}))}{p(z|x_i;\Psi^{\text{old}})^{1-q}} \\
&= \sum_{i=1}^{n} E_{\Psi^{\text{old}}}\Big[\frac{L_q(p(X,Z;\Psi))}{p(Z|X;\Psi^{\text{old}})^{1-q}} - \frac{L_q(p(Z|X;\Psi^{\text{old}}))}{p(Z|X;\Psi^{\text{old}})^{1-q}}\Big| X = x_i\Big] \\
&= B(\Psi,\Psi^{\text{old}}) + C(\Psi^{\text{old}},\Psi^{\text{old}}),
\end{aligned}$$

where, from equation (20) to (21), we have used Jensen's inequality on the L_q function due to its concavity (Lemma 1, part (iv) in Section 9). When $\Psi = \Psi^{\text{old}}$, we have

$$B(\Psi^{\text{old}},\Psi^{\text{old}}) + C(\Psi^{\text{old}},\Psi^{\text{old}}) = A(\Psi^{\text{old}},\Psi^{\text{old}}) = \sum_{i=1}^{n} L_q(p(x_i;\Psi^{\text{old}}))$$

Proof of Theorem 4: Define

$$D(\Psi) = \sum_{i=1}^{n} L_q(p(x_i;\Psi)) - \big(B(\Psi,\Psi^{\text{old}}) + C(\Psi^{\text{old}},\Psi^{\text{old}})\big) \geq 0. \qquad (22)$$

By theorem 3, we know that $D(\Psi^{\text{old}}) = 0$ and $D(\Psi) \geq 0$, so $D(\Psi)$ obtains its minimum at $\Psi = \Psi^{\text{old}}$, i.e.,

$$\frac{\partial}{\partial \Psi} D(\Psi)\Big|_{\Psi=\Psi^{\text{old}}} = 0.$$

Take the derivative of both sides of (22), we have the first part of the theorem. Together with equation (7) and (8), we prove the rest of the theorem.

Proof of Theorem 5: For mixture models, we plug equation (10) and (11) in B,

$$\begin{aligned}
B(\Psi, \Psi^{\text{old}}) &= \sum_{i=1}^{n} E_{\Psi^{\text{old}}}\Big[\frac{L_q(\prod_{j=1}^{k}(\pi_j f_j(X;\theta_j))^{Z_j})}{(\prod_{j=1}^{k}(\frac{\pi_j^{\text{old}} f_j(X;\theta_j^{\text{old}})}{f(X;\Psi^{\text{old}})})^{Z_j})^{1-q}}\Big| X = x_i\Big] \\
&= \sum_{i=1}^{n}\sum_{j=1}^{k} L_q(\pi_j f_j(x_i;\theta_j)) \cdot \frac{p(Z_j = 1, Z_{-j} = 0 | X = x_i; \Psi^{\text{old}})}{(\frac{\pi_j^{\text{old}} f_j(x_i;\theta_j^{\text{old}})}{f(x_i;\Psi^{\text{old}})})^{1-q}} \\
&= \sum_{i=1}^{n}\sum_{j=1}^{k} \tau_j(x_i, \Psi^{\text{old}})^q L_q(\pi_j f_j(x_i;\theta_j)).
\end{aligned}$$

Proof of Theorem 6: Apply the first order condition on B (note $\pi_k = 1 - \sum_{j=1}^{k-1}\pi_j$),

$$\frac{\partial}{\partial \theta_j} B(\Psi, \Psi^{\text{old}}) = \sum_{i=1}^{n} \tilde{\tau}_j(x_i, \Psi^{\text{old}}) \frac{\frac{\partial}{\partial\theta_j} f_j(x_i;\theta_j)}{f_j(x_i;\theta_j)} (\pi_j f_j(x_i;\theta_j))^{1-q}, \tag{23}$$

$$\begin{aligned}
\frac{\partial}{\partial \pi_j} B(\Psi, \Psi^{\text{old}}) &= \sum_{i=1}^{n} \tilde{\tau}_j(x_i, \Psi^{\text{old}}) \frac{f_j(x_i;\theta_j)}{(\pi_j f_j(x_i;\theta_j))^q} - \sum_{i=1}^{n} \tilde{\tau}_k(x_i, \Psi^{\text{old}}) \frac{f_k(x_i;\theta_k)}{(\pi_k f_k(x_k;\theta_k))^q} \\
&= \sum_{i=1}^{n} \frac{\tilde{\tau}_j(x_i, \Psi^{\text{old}}) f_j(x_i;\theta_j)^{1-q}}{\pi_j^q} - \sum_{i=1}^{n} \frac{\tilde{\tau}_k(x_i, \Psi^{\text{old}}) f_k(x_i;\theta_k)^{1-q}}{\pi_k^q},
\end{aligned} \tag{24}$$

$$\Rightarrow 0 = \sum_{i=1}^{n} \tilde{\tau}_j(x_i, \Psi^{\text{old}}) \frac{\frac{\partial}{\partial\theta_j} f_j(x_i;\theta_j)}{f_j(x_i;\theta_j)} f_j(x_i;\theta_j)^{1-q} \quad \text{and} \quad \pi_j \propto \Big[\sum_{i=1}^{n} \tilde{\tau}_j(x_i, \Psi^{\text{old}}) f_j(x_i;\theta_j)^{1-q}\Big]^{\frac{1}{q}}.$$

Proof of Theorem 4 for the mixture model case: By equations (23) and (24), the derivatives of B at $\Psi = \Psi^{\text{old}}$ are,

$$\begin{aligned}
\frac{\partial}{\partial \theta_j} B(\Psi, \Psi^{\text{old}})\Big|_{\Psi=\Psi^{\text{old}}} &= \sum_{i=1}^{n} \Big(\frac{\pi_j^{\text{old}} f_j(x_i;\theta_j^{\text{old}})}{f(x_i;\Psi^{\text{old}})}\Big)^q \cdot \frac{\frac{\partial}{\partial\theta_j} f_j(x_i;\theta_j)\big|_{\theta_j=\theta_j^{\text{old}}}}{f_j(x_i;\theta_j^{\text{old}})} (\pi_j^{\text{old}} f_j(x_i;\theta_j^{\text{old}}))^{1-q} \\
&= \sum_{i=1}^{n} \frac{\pi_j^{\text{old}} \big(\frac{\partial}{\partial\theta_j} f_j(x_i;\theta_j)\big|_{\theta_j=\theta_j^{\text{old}}}\big)}{f(x_i;\Psi^{\text{old}})^q}, \\
\frac{\partial}{\partial \pi_j} B(\Psi, \Psi^{\text{old}})\Big|_{\Psi=\Psi^{\text{old}}} &= \sum_{i=1}^{n} \Big(\frac{\pi_j^{\text{old}} f_j(x_i;\theta_j^{\text{old}})}{f(x_i;\Psi^{\text{old}})}\Big)^q \frac{f_j(x_i;\theta_j^{\text{old}})^{1-q}}{(\pi_j^{\text{old}})^q} - \sum_{i=1}^{n} \Big(\frac{\pi_k^{\text{old}} f_k(x_i;\theta_k^{\text{old}})}{f(x_i;\Psi^{\text{old}})}\Big)^q \frac{f_k(x_i;\theta_k^{\text{old}})^{1-q}}{(\pi_k^{\text{old}})^q} \\
&= \sum_{i=1}^{n} \frac{f_j(x_i;\theta_j^{\text{old}}) - f_k(x_i;\theta_k^{\text{old}})}{f(x_i;\Psi^{\text{old}})^q}.
\end{aligned}$$

We calculate the derivatives of $\sum_{i=1}^{n} L_q(p(x_i;\Psi))$ at $\Psi = \Psi^{\text{old}}$,

$$\begin{aligned}
\frac{\partial}{\partial \theta_j} \sum_{i=1}^{n} L_q(p(x_i;\Psi))\Big|_{\Psi=\Psi^{\text{old}}} &= \sum_{i=1}^{n} \frac{\pi_j^{\text{old}} \big(\frac{\partial}{\partial\theta_j} f_j(x_i;\theta_j)\big|_{\theta_j=\theta_j^{\text{old}}}\big)}{f(x_i;\Psi^{\text{old}})^q}, \\
\frac{\partial}{\partial \pi_j} \sum_{i=1}^{n} L_q(p(x_i;\Psi))\Big|_{\Psi=\Psi^{\text{old}}} &= \sum_{i=1}^{n} \frac{f_j(x_i;\theta_j^{\text{old}}) - f_k(x_i;\theta_k^{\text{old}})}{f(x_i;\Psi^{\text{old}})^q}.
\end{aligned}$$

By comparing the formulas above, we obtain the first equation of the theorem. Together with equation (7) and (8), we prove the rest of the theorem.

References

Bickel, P. J. and Doksum, K. A. (2007). *Mathematical Statistics Basic Ideas and Selected Topics Volume I.* Pearson Prentice Hall, second edition.

Cuesta-Albertos, J. A., Matran, C., and Mayo-Iscar, A. (2008). Robust estimation in the normal mixture model based on robust clustering. *Journal of the Royal Statistical Society: Series B*, 70:779–802.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B*, 39:1–38.

Ferrari, D. and Yang, Y. (2010). Maximum Lq-likelihood estimation. *Annals of Statistics*, 38:753–783.

Garcia-Escudero, L. A. and Gordaliza, A. (1999). Robustness properties of k-means and trimmed k-means. *Journal of the American Statistical Association*, 94:956–969.

Guo, W. and Cui, S. (2008). A q-parameterized deterministic annealing em algorithm based on nonextensive statistical mechanics. *IEEE Transactions on Signal Processing*, 56 (7):3069–3080.

Lange, K., Hunter, D. R., and Yang, I. (2000). Optimization transfer using surrogate objective functions. *Journal of Computational and Graphical Statistics*, 9:1–20.

McLachlan, G. J., Ng, S.-K., and Bean, R. (2006). Robust cluster analysis via mixture models. *Austrian Journal of Statistics*, 35:157–174.

Meng, X.-L. and Rubin, D. B. (1993). Maximum likelihood from incomplete data via the em algorithm. *Biometrika*, 80 (2):267–278.

Peel, D. and McLachlan, G. J. (2000). Robust mixture modelling using the t distribution. *Statistics and Computing*, 10:339–348.

Redner, R. A. and Walker, H. F. (1984). Mixture densities, maximum likelihood and the em algorithm. *Society for Industrial and Applied Mathematics Review*, 26 (2):195–239.

Silverman, B. W. (1986). *Density Estimation for Statistics and Data Analysis*. Chapman and Hall.

Tadjudin, S. and Landgrebe, D. A. (2000). Robust parameter estimation for mixture model. *IEEE Transactions on Geoscience and Remote Sensing*, 38:439–445.

Windham, M. P. and Cutler, A. (1992). Information ratios for validating mixture analyses. *Journal of the American Statistical Association*, 87:1188–1192.

Wu, C. F. J. (1983). On the convergence properties of the em algorithm. *Annals of Statistics*, 11:95–103.