Estimating Local Optimums in EM Algorithm over Gaussian Mixture Model

Zhenjie Zhang ZHENJIE@COMP.NUS.EDU.SG
Bing Tian Dai DAIBINGT@COMP.NUS.EDU.SG
Anthony K.H. Tung ATUNG@COMP.NUS.EDU.SG
School of Computing, National University of Singapore, Computing 1, Law Link, 117590, Singapore

Abstract

EM algorithm is a very popular iteration-based method to estimate the parameters of Gaussian Mixture Model from a large observation set. However, in most cases, EM algorithm is not guaranteed to converge to the global optimum. Instead, it stops at some local optimums, which can be much worse than the global optimum. Therefore, it is usually required to run multiple procedures of EM algorithm with different initial configurations and return the best solution. To improve the efficiency of this scheme, we propose a new method which can estimate an upper bound on the logarithm likelihood of the local optimum, based on the current configuration after the latest EM iteration. This is accomplished by first deriving some region bounding the possible locations of local optimum, followed by some upper bound estimation on the maximum likelihood. With this estimation, we can terminate an EM algorithm procedure if the estimated local optimum is definitely worse than the best solution seen so far. Extensive experiments show that our method can effectively and efficiently accelerate conventional multiple restart EM algorithm.

1. Introduction

Gaussian Mixture Model (GMM) (McLachlan & Peel, 2000) is a powerful tool in unsupervised learning to model unlabelled data in a multi-dimensional space. However, given an observation data set, estimating the parameters of the underlying Gaussian Mixture Model of the data is not a trivial task, especially when the dimensionality or the number of components is large. Usually, this model estimation problem is transformed to a new problem, which try to find parameters maximizing the likelihood probability on the observations from the Gaussian distributions. In the past decades, EM algorithm (Dempster et al., 1977) has become the most widely method used in the problem of learning Gaussian Mixture Model (Ma et al., 2001; Jordan & Xu, 1995; McLachlan & Krishnan, 1996).

Although EM algorithm can converge in finite iterations, there is no guarantee on the convergence to global optimum. Instead, it usually stops at some local optimum, which can be arbitrarily worse than the global optimum. Although there have been extensive studies on how to avoid bad local optimums, it is still required to run EM algorithm with different random initial configurations and the best local optimum is returned as final result. This leads to a great waste of computation resource since most of the calculations do not have any contribution to the final result.

In this paper, we propose a fast stopping method to overcome the problem of trapping into bad local optimums. Given any current configuration after an EM iteration, our method can estimate an upper bound on the final likelihood of the local optimum current configuration is leading to. Therefore, if the estimated local optimum is definitely not better than the best local optimum achieved in previous runs, current procedure can be terminated immediately.

To facilitate such local optimum estimation, we first prove that a region in the parameter space can definitely cover the unknown local optimum. If a region covers the current configuration and any configuration on the boundary of the region gives lower likelihood than the current one does, we can show that the local optimum is "trapped" in the region; and we call such region as a maximal region. In this paper, we adopt a

Appearing in *Proceedings of the 25th International Conference on Machine Learning*, Helsinki, Finland, 2008. Copyright 2008 by the author(s)/owner(s).

special type of maximal region, which can be computed efficiently. Since the best likelihood of any configuration in a maximal region can be estimated in relatively short time, it can be decided immediately on whether current procedure still has potential to achieve a better local optimum. In our experiments, such method is shown to greatly improve the efficiency of original EM algorithm for GMM, on both synthetic and real data sets.

The rest of the paper is organized as follows. We first introduce the definitions and related works on Gaussian Mixture Model and EM algorithm in Section 2 . Section 3 proves the local trapping property of EM algorithm on GMM; and Section 4 presents our study on maximal region of local optimum. We propose our algorithm on estimating the likelihood of a local optimum in Section 5 . Section 6 shows some experimental result. Finally, section 7 concludes this paper.

2. Model and Related Works

In this section, we review the basic models of Gaussian Mixture Model, EM algorithm, and some acceleration method proposed for a special type of Gaussian Mixture Model (K-Means Algorithm).

2.1. Gaussian Mixture Model

In GMM model (McLachlan & Peel, 2000), there exist k underlying components $\{\omega_1, \omega_2, \ldots, \omega_k\}$ in a d-dimensional data set. Each component follows some Gaussian distribution in the space. The parameters of the component ω_j include $\boldsymbol{\Theta}_j = \{\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j, \pi_j\}$, in which $\boldsymbol{\mu}_j = (\boldsymbol{\mu}_j[1], \ldots, \boldsymbol{\mu}_j[d])$ is the center of the Gaussian distribution, $\boldsymbol{\Sigma}_j$ is the covariance matrix of the distribution and π_j is the probability of the component ω_j. Based on the parameters, the probability of a point coming from component ω_j appearing at $\boldsymbol{x} = (x[1], \ldots, x[d])$ can be represented by

$$\Pr(\boldsymbol{x}|\boldsymbol{\Theta}_j) = \frac{|\boldsymbol{\Sigma}_j^{-1}|^{1/2}}{(2\pi)^{d/2}} \exp\left\{-\frac{1}{2}(x - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1}(x - \boldsymbol{\mu}_j)\right\}$$

Thus, given the component parameter set $\boldsymbol{\Theta} = \{\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2, \ldots, \boldsymbol{\Theta}_k\}$ but without any component information on an observation point $\boldsymbol{x}$, the probability of observing $\boldsymbol{x}$ is estimated by

$$\Pr(\boldsymbol{x}|\boldsymbol{\Theta}) = \sum_{j=1}^{k} \Pr(\boldsymbol{x}|\boldsymbol{\Theta}_j)\pi_j$$

The problem of learning GMM is estimating the parameter set $\boldsymbol{\Theta}$ of the k component to maximize the likelihood of a set of observations $D = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$, which is represented by

$$\Pr(D|\boldsymbol{\Theta}) = \prod_{i=1}^{n} \Pr(\boldsymbol{x}_i|\boldsymbol{\Theta}) \qquad (1)$$

Based on the parameters of the GMM model, the posterior probability of $\boldsymbol{x}_i$ from component ω_j (or the weight of $\boldsymbol{x}_i$ in component j), τ_{ij}, can be calculated as follows.

$$\tau_{ij} = \frac{\Pr(\boldsymbol{x}_i|\boldsymbol{\Theta}_j)\pi_j}{\sum_{l=1}^{k} \Pr(\boldsymbol{x}_i|\boldsymbol{\Theta}_l)\pi_l} \qquad (2)$$

To simplify the notations, we use $\boldsymbol{\Phi}$ to denote the set of all τ_{ij} for any pair of i, j, and use $\Psi(\boldsymbol{\Theta})$ to denote the corresponding $\boldsymbol{\Phi}$ based on current configuration $\boldsymbol{\Theta}$. For ease of analysis, the original optimization problem on equation (1), is usually transformed to an equal maximization problem on the following variable, called log likelihood.

$$\begin{aligned} L(\boldsymbol{\Theta}, \boldsymbol{\Phi}) &= \sum_{i=1}^{n} \sum_{j=1}^{k} \tau_{ij} (\ln \frac{\pi_j}{\tau_{ij}} + \frac{\ln |\boldsymbol{\Sigma}_j^{-1}|}{2} - \\ & \quad \frac{(\boldsymbol{x}_i - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_j)}{2}) \end{aligned} \qquad (3)$$

L is actually a function over $\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$, the latter of which is usually optimized according to $\boldsymbol{\Theta}$. Thus, the problem of learning GMM is finding an optimal parameter set $\boldsymbol{\Theta}^*$ which can maximize the function $L(\boldsymbol{\Theta}^*, \Psi(\boldsymbol{\Theta}^*))$.

2.2. EM Algorithm

EM algorithm (Dempster et al., 1977) is a widely used technique for probabilistic parameter estimation. To estimate $\boldsymbol{\Theta} = \{\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_k\}$, it starts with a randomly chosen initial parameter configuration $\boldsymbol{\Theta}^0$. Then, it keeps invoking iterations to recompute $\boldsymbol{\Theta}^{t+1}$ based on $\boldsymbol{\Theta}^t$. Every iteration consists of two steps, E-step and M-step. In E-step, the algorithm computes the expected value of τ_{ij} for each pair of i and j based on $\boldsymbol{\Theta}^t = \{\boldsymbol{\Theta}_1^t, \ldots, \boldsymbol{\Theta}_k^t\}$ and equation (2).

In M-step, the algorithm finds a new group of parameters $\boldsymbol{\Theta}^{t+1}$ to maximize L based on $\boldsymbol{\Phi}^t = \{\tau_{ij}^t\}$ and $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$. The details of the update process for $\boldsymbol{\mu}_j$, $\boldsymbol{\Sigma}_j$ and π_j are listed below.

$$\boldsymbol{\mu}_j^{t+1}[l] = \frac{\sum_{i=1}^{n} \tau_{ij}^t \boldsymbol{x}_i[l]}{\sum_{i=1}^{n} \tau_{ij}^t} \qquad (4)$$

$$\boldsymbol{\Sigma}_j^{t+1} = \frac{\sum_{i=1}^n \tau_{ij}^t (\boldsymbol{x}_i - \boldsymbol{\mu}_j^{t+1})(\boldsymbol{x}_i - \boldsymbol{\mu}_j^{t+1})^T}{\sum_{i=1}^n \tau_{ij}^t} \qquad (5)$$

$$\pi_j^{t+1} = \frac{\sum_{i=1}^n \tau_{ij}^t}{n} \qquad (6)$$

The iteration process stops only when $\boldsymbol{\Theta}^N = \boldsymbol{\Theta}^{N-1}$ after N iterations. We note that both E-step and M-step always improve the objective function, i.e. $L(\boldsymbol{\Theta}^t, \boldsymbol{\Phi}^t) \geq L(\boldsymbol{\Theta}^t, \boldsymbol{\Phi}^{t-1}) \geq L(\boldsymbol{\Theta}^{t-1}, \boldsymbol{\Phi}^{t-1})$. Based on this property, EM-algorithm will definitely converge to some local optimum. The convergence properties of EM algorithm over GMM have been extensively studied in (Xu & Jordan, 1996; Ma et al., 2001).

2.3. K-Means Algorithm and Its Acceleration

K-Means algorithm can be considered as a special problem of GMM learning with several constraints. First, the covariance matrix for each component must be identity matrix. Second, the posterior probability τ_{ij} can only be 0 or 1. Therefore, in E-step of the algorithm, each point is assigned to the closest center under Euclidean distance; whereas in M-step, the set of geometric center of each cluster is used to replace the old set.

With the problem simplification from GMM to K-Means, there have been many methods proposed to accelerate the multiple restart EM algorithm for K-Means. In (Kanungo et al., 2002), for example, Kanungo et al. applied indexing technique to achieve a much more efficient implementation of E-step. In (Elkan, 2003), Elkan accelerated both E-step and M-step by employing triangle inequality of Euclidean distance to reduce the time for distance computations. In (Zhang et al., 2006), Zhang et al. introduced a lower bound estimation on the k-means local optimums to efficiently cut the procedures not leading to good solutions. However, all these methods proposed for k-means algorithm cannot be directly extended to the general GMM. As far as we know, our paper is the first study on acceleration of the multiple restart EM algorithm with robustness guarantee.

To improve the readability of the paper, we summarize all notations in Table 1.

Table 1: Table of Notations

Notation	Description
n	number of points in data
d	dimensionality of data space
k	number of components
ω_j	component j
$\boldsymbol{\Theta}_j$	parameter set of ω_j
$\boldsymbol{\mu}_j$	center of ω_j
$\boldsymbol{\Sigma}_j$	covariance matrix of ω_j
π_j	probability of ω_j
$\boldsymbol{\Theta}$	configuration of all components
$\boldsymbol{x}_i$	ith point in the data
τ_{ij}	posterior probability $\Pr(\omega_j \vert \boldsymbol{x}_i)$
$\boldsymbol{\Phi}$	the set of all τ_{ij}
$\Psi(\boldsymbol{\Theta})$	the optimal $\boldsymbol{\Phi}$ with $\boldsymbol{\Theta}$
$\mathcal{S}$	solutions space for configurations
$L(\boldsymbol{\Theta}, \boldsymbol{\Phi})$	objective log likelihood function
Δ	a parameter for a maximal region

3. Local Trapping Property

In this section, we prove the local trapping property of EM algorithm on GMM. To derive the analysis, we first define a solution space $\mathcal{S}$, containing $(d^2 + d + 1)k$ dimensions where d is the dimensionality of the original data space. Any configuration $\boldsymbol{\Theta}$, either valid or invalid, can be represented by a point in $\mathcal{S}$. Without loss of generality, we use $\boldsymbol{\Theta}$ to denote the configuration as well as the corresponding point in solution space $\mathcal{S}$. The rest of the section will be spent to prove the following theorem.

Theorem 1 *Given a closed region R in the solution space $\mathcal{S}$ covering current configuration $\boldsymbol{\Theta}^t$, EM algorithm converges to a local optimum in R if every configuration $\boldsymbol{\Theta}$ on the boundary of R has $L(\boldsymbol{\Theta}, \Psi(\boldsymbol{\Theta})) < L(\boldsymbol{\Theta}^t, \boldsymbol{\Phi}^t)$*

Given two configurations $\boldsymbol{\Theta}^t$ and $\boldsymbol{\Theta}^{t+1}$ across one EM iteration, we define a path between $\boldsymbol{\Theta}^t$ and $\boldsymbol{\Theta}^{t+1}$ in $\mathcal{S}$ as follows. This path consists of two parts, called P^1 and P^2 respectively. P^1 starts at $\boldsymbol{\Theta}^t$ and ends at $\boldsymbol{\Theta}^{\#}$, where $\boldsymbol{\Theta}^{\#} = \{\boldsymbol{\Theta}_j^{\#}\}$. Here $\boldsymbol{\Theta}_j^{\#} = \{\boldsymbol{\mu}_j^t, \boldsymbol{\Sigma}_j^{\#}, \pi_j^t\}$, and $\boldsymbol{\Sigma}_j^{\#} = \sum_i \tau_{ij}^t (\boldsymbol{x}_i - \boldsymbol{\mu}_j^t)(\boldsymbol{x}_i - \boldsymbol{\mu}_j^t)^T / \sum_i \tau_{ij}^t$. An intermediate configuration between $\boldsymbol{\Theta}^t$ and $\boldsymbol{\Theta}^{\#}$ is defined as $\boldsymbol{\Theta}^{\alpha}$, in which $\boldsymbol{\mu}_j^t$ and π_j^t remain the same, while $\boldsymbol{\Sigma}_j^{\alpha}$ in $\boldsymbol{\Theta}^{\alpha}$ is $((1-\alpha)(\boldsymbol{\Sigma}^t)^{-1} + \alpha(\boldsymbol{\Sigma}^{\#})^{-1})^{-1}$. When α increases from 0 to 1, we can move from $\boldsymbol{\Theta}^t$ to $\boldsymbol{\Theta}^{\#}$ in the solutions space $\mathcal{S}$. The second part of the path starts at $\boldsymbol{\Theta}^{\#}$ and ends at $\boldsymbol{\Theta}^{t+1}$. Any intermediate configuration $\boldsymbol{\Theta}^{\beta} = \{\boldsymbol{\Theta}_j^{\beta}\}$, where $\boldsymbol{\mu}_j^{\beta} = (1-\beta)\boldsymbol{\mu}_j^t + \beta\boldsymbol{\mu}_j^{t+1}$, $\boldsymbol{\Sigma}_j^{\beta} = \sum_i \tau_{ij}^t (\boldsymbol{x}_i - \boldsymbol{\mu}_j^{\beta})(\boldsymbol{x}_i - \boldsymbol{\mu}_j^{\beta})^T / \sum_i \tau_{ij}^t$, and $\pi_j^{\beta} = (\pi_j^t)^{1-\beta}(\pi_j^{t+1})^{\beta}$. Similarly, a continuous movement from $\boldsymbol{\Theta}^{\#}$ to $\boldsymbol{\Theta}^{t+1}$ can be made by increasing β from 0 to 1. The following lemmas prove that any intermediate configuration on the path is a better solution than $\boldsymbol{\Theta}^t$.

Lemma 1 *Given any intermediate configuration $\mathbf{\Theta}^{\alpha}$ between $\mathbf{\Theta}^{t}$ and $\mathbf{\Theta}^{\#}$, we have $L(\mathbf{\Theta}^{\alpha}, \Psi(\mathbf{\Theta}^{\alpha})) \geq L(\mathbf{\Theta}^{t}, \mathbf{\Phi}^{t})$.*

Proof: By the optimality property of $\Psi(\mathbf{\Theta}^{\alpha})$, we have $L(\mathbf{\Theta}^{\alpha}, \Psi(\mathbf{\Theta}^{\alpha})) \geq L(\mathbf{\Theta}^{\alpha}, \mathbf{\Phi}^{t})$.

Since $\mathbf{\Sigma}_j^{\#} = \sum_i \tau_{ij}^t (\boldsymbol{x}_i - \boldsymbol{\mu}_j^t)(\boldsymbol{x}_i - \boldsymbol{\mu}_j^t)^T / \sum_i \tau_{ij}^t$ is the optimal choice for $\mathbf{\Sigma}_j$ if τ_{ij}^t, $\boldsymbol{\mu}_j^t$ and π_j^t are fixed, we also have $L(\mathbf{\Theta}^{\#}, \mathbf{\Phi}^{t}) \geq L(\mathbf{\Theta}^{t}, \mathbf{\Phi}^{t})$.

By the definition of $\mathbf{\Theta}^{\alpha}$ and the property of $\mathbf{\Theta}^{\#}$ above, the following equations can be easily derived.

$$\begin{aligned} & L(\mathbf{\Theta}^{\alpha}, \mathbf{\Phi}^{t}) \\ = \ & (1-\alpha) L(\mathbf{\Theta}^{t}, \mathbf{\Phi}^{t}) + \alpha L(\mathbf{\Theta}^{\#}, \mathbf{\Phi}^{t}) \\ \geq \ & L(\mathbf{\Theta}^{t}, \mathbf{\Phi}^{t}) \end{aligned}$$

Therefore, it is straightforward to reach the conclusion that $L(\mathbf{\Theta}^{\alpha}, \Psi(\mathbf{\Theta}^{\alpha})) \geq L(\mathbf{\Theta}^{t}, \mathbf{\Phi}^{t})$. □

Lemma 2 *Given any intermediate configuration $\mathbf{\Theta}^{\beta}$ between $\mathbf{\Theta}^{\#}$ and $\mathbf{\Theta}^{t+1}$, we have $L(\mathbf{\Theta}^{\beta}, \Psi(\mathbf{\Theta}^{\alpha})) \geq L(\mathbf{\Theta}^{t}, \mathbf{\Phi}^{t})$.*

Proof: Again, the basic inequality $L(\mathbf{\Theta}^{\beta}, \Psi(\mathbf{\Theta}^{\beta})) \geq L(\mathbf{\Theta}^{\beta}, \mathbf{\Theta}^{t})$ holds. Based on this, we can prove the lemma by showing $L(\mathbf{\Theta}^{\beta}, \mathbf{\Phi}^{t}) \geq L(\mathbf{\Theta}^{\#}, \mathbf{\Phi}^{t})$, since $L(\mathbf{\Theta}^{\#}, \mathbf{\Phi}^{t}) \geq L(\mathbf{\Theta}^{t}, \mathbf{\Phi}^{t})$.

If $\mathbf{\Sigma}_j^{\beta} = \sum_i \tau_{ij}^t (\boldsymbol{x}_i - \boldsymbol{\mu}_j^{\beta})(\boldsymbol{x}_i - \boldsymbol{\mu}_j^{\beta})^T / \sum_i \tau_{ij}^t$, a very interesting result is that $\sum_j \sum_i \tau_{ij}^t (\boldsymbol{x}_i - \boldsymbol{\mu}_j^{\beta})^T (\mathbf{\Sigma}_j^{\beta})^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_j^{\beta})$ remains constant for any β, as is shown below.

$$\sum_j \sum_i \tau_{ij}^t (\boldsymbol{x}_i - \boldsymbol{\mu}_j^{\beta})^T (\mathbf{\Sigma}_j^{\beta})^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_j^{\beta}) = nd$$

Therefore, for any $\mathbf{\Theta}^{\beta}$, we only need to consider the sum $\sum_i \sum_j \tau_{ij}^t \left(\ln(\pi_j^{\beta} / \tau_{ij}^t) - \ln(|\mathbf{\Sigma}_j^{\beta}|)/2 \right)$.

By the definition of π_j^{β}, since $\pi_j^{\beta} = (\pi_j^t)^{1-\beta} (\pi_j^{t+1})^{\beta}$, we have $\ln \pi_j^{\beta} = (1-\beta) \ln \pi_j^t + \beta \ln \pi_j^{t+1}$. Then,

$$\sum_{i=1}^{n} \tau_{ij}^t \ln \frac{\pi_j^{\beta}}{\tau_{ij}^t} = (1-\beta) \sum_{i=1}^{n} \tau_{ij}^t \ln \frac{\pi_j^t}{\tau_{ij}^t} + \beta \sum_{i=1}^{n} \tau_{ij}^t \ln \frac{\pi_j^{t+1}}{\tau_{ij}^t} \tag{7}$$

Therefore, $\sum_i \sum_j \tau_{ij}^t \ln \frac{\pi_j^{\beta}}{\tau_{ij}^t} \geq \sum_i \sum_j \tau_{ij}^t \ln \frac{\pi_j^t}{\tau_{ij}^t}$, since $\sum_{i=1}^{n} \tau_{ij}^t \ln \frac{\pi_j^{t+1}}{\tau_{ij}^t} \geq \sum_{i=1}^{n} \tau_{ij}^t \ln \frac{\pi_j^t}{\tau_{ij}^t}$.

On the other hand, based on the definition of $\mathbf{\Sigma}_j^{\beta}$, we can prove that

$$\begin{aligned} \mathbf{\Sigma}_j^{\beta} = & \sum_{i=1}^{n} \tau_{i,j}^t (\boldsymbol{x}_i - \boldsymbol{\mu}_j^t)(\boldsymbol{x}_i - \boldsymbol{\mu}_j^t)^T + \\ & (\beta^2 - 2\beta)(\sum_{i=1}^{n} \tau_{i,j}^t)(\boldsymbol{\mu}_j^{t+1} - \boldsymbol{\mu}_j^t)(\boldsymbol{\mu}_j^{t+1} - \boldsymbol{\mu}_j^t)^T \end{aligned}$$

Since $\beta^2 - 2\beta \leq 0$ for any β between 0 and 1, $\ln |\mathbf{\Sigma}_j^{\beta}| \leq \ln |\mathbf{\Sigma}_j^{\#}|$. And thus, we have $-\ln |\mathbf{\Sigma}_j^{\beta}|/2 \geq -\ln |\mathbf{\Sigma}_j^{\#}|/2$.

Combing the results above, we reach the conclusion that $L(\mathbf{\Theta}^{\beta}, \mathbf{\Phi}^{t}) \geq L(\mathbf{\Theta}^{\#}, \mathbf{\Phi}^{t})$, leading to the correctness of the lemma. □

Proof for Theorem 1

Proof: We prove the theorem by contradiction. If R satisfies the boundary condition but EM algorithm converges to some configuration out of R in $\mathcal{S}$, there is at least one pair of configurations $\{\mathbf{\Theta}^{s}, \mathbf{\Theta}^{s+1}\}$ that $\mathbf{\Theta}^{s}$ is in R but $\mathbf{\Theta}^{s+1}$ is not. By setting up the path $\{\mathbf{\Theta}^{\alpha}\} \cap \{\mathbf{\Theta}^{\beta}\}$ between $\mathbf{\Theta}^{s}$ and $\mathbf{\Theta}^{s+1}$ as defined above, we know there is at least one $\mathbf{\Theta}^{\alpha}$ ($\mathbf{\Theta}^{\beta}$) that $\mathbf{\Theta}^{\alpha}$ ($\mathbf{\Theta}^{\beta}$) is exactly on the boundary of R. By Lemma 1 (Lemma 2), we know $L(\mathbf{\Theta}^{\alpha}, \Psi(\mathbf{\Theta}^{\alpha})) \geq L(\mathbf{\Theta}^{s}, \mathbf{\Phi}^{s})$ ($L(\mathbf{\Theta}^{\beta}, \Psi(\mathbf{\Theta}^{\beta})) \geq L(\mathbf{\Theta}^{s}, \mathbf{\Phi}^{s})$). On the other hand, any $\mathbf{\Theta}^{\alpha}$ or $\mathbf{\Theta}^{\beta}$ is better than $\mathbf{\Theta}^{t}$ by the definition of R. This leads to the contradiction, since $L(\mathbf{\Theta}^{s}, \mathbf{\Phi}^{s}) > L(\mathbf{\Theta}^{t}, \mathbf{\Phi}^{t})$. □

4. Maximal Region

Based on Theorem 1, we define the concept of *Maximal Region* in GMM as follows. Given the current configuration $\mathbf{\Theta}^{t}$, a region R in S is the maximal region for $\mathbf{\Theta}^{t}$, if (1) R covers $\mathbf{\Theta}^{t}$, and (2) any boundary configuration $\mathbf{\Theta}$ of R has $L(\mathbf{\Theta}, \Psi(\mathbf{\Theta})) < L(\mathbf{\Theta}^{t}, \mathbf{\Phi}^{t})$, by Theorem 1, EM algorithm converges to some local optimum in R.

Given the current configuration $\mathbf{\Theta}^{t}$, there are infinite number of valid maximal regions in the solution space, most of which are hard to verify and manipulate. To facilitate efficient computation, we further propose a special class of maximal regions. Given $\mathbf{\Theta}^{t}$ and a positive real value $\Delta < 1$, we define a closed region $R(\mathbf{\Theta}^{t}, \Delta) \subseteq S$ as the union of any configuration $\mathbf{\Theta}$, each $\theta_j = \{\boldsymbol{\mu}_j, \mathbf{\Sigma}_j, \pi_j\}$ of which satisfies all of the conditions below: (1) $(1-\Delta)\pi_j^t \leq \pi_j \leq (1+\Delta)\pi_j^t$; (2) $-\Delta \leq tr(\mathbf{\Sigma}_j^{-1}(\mathbf{\Sigma}_j^t) - I) \leq \Delta$; and (3) $(\boldsymbol{\mu}_j - \boldsymbol{\mu}_j^t)^T (\mathbf{\Sigma}_j^t)^{-1} (\boldsymbol{\mu}_j - \boldsymbol{\mu}_j^t) \leq \Delta^2$; where $tr(M)$ denotes the trace of the matrix M and I is the identity matrix of dimension d.

Theorem 2 *Any configuration* $\mathbf{\Theta}$ *on the boundary of* $R(\mathbf{\Theta}^t, \Delta)$ *has* $L(\mathbf{\Theta}, \mathbf{\Phi}^{t-1}) \leq L(\mathbf{\Theta}^t, \mathbf{\Phi}^{t-1}) - n \min_j \pi_j^t \Delta^2/6$.

Proof: Given any $R(\mathbf{\Theta}^t, \Delta)$, any configuration on the boundary must satisfy one of the following conditions for at least one j $(1 \leq j \leq k)$: (1) $(1-\Delta)\pi_j^t = \pi_j$; (2) $\pi_j^t = (1+\Delta)\pi_j^t$; (3) $|tr(\mathbf{\Sigma}_j^{-1}(\mathbf{\Sigma}_j^t) - I)| = \Delta$; and (4) $(\boldsymbol{\mu}_j - \boldsymbol{\mu}_j^t)^T(\mathbf{\Sigma}_j^t)^{-1}(\boldsymbol{\mu}_j - \boldsymbol{\mu}_j^t) = \Delta^2$.

If $\mathbf{\Theta}$ satisfies condition (1) for some component l, $L(\mathbf{\Theta}, \mathbf{\Phi}^{t-1})$ is maximized if $\boldsymbol{\mu}_j^t$ and $\mathbf{\Sigma}_j^t$ remain unchanged for all j, while $\pi_j = \frac{1-(1-\Delta)\pi_l^t}{1-\pi_l^t}\pi_j^t$ for all $j \neq l$. Therefore, we have the following upper bound.

$$\begin{aligned}
& L(\mathbf{\Theta}, \mathbf{\Phi}^{t-1}) - L(\mathbf{\Theta}^t, \mathbf{\Phi}^{t-1}) \\
\leq \quad & n\pi_l^t \ln(1-\Delta) + n(1-\pi_l^t) \ln \frac{1-(1-\Delta)\pi_l^t}{1-\pi_l^t} \\
= \quad & n\pi_l^t \ln(1-\Delta) + n(1-\pi_l^t) \ln \left(1 + \frac{\Delta\pi_l^t}{1-\pi_l^t}\right) \\
\leq \quad & n\pi_l^t \left(-\Delta - \frac{\Delta^2}{2}\right) + n(1-\pi_l^t)\frac{\Delta\pi_l^t}{1-\pi_l^t} \\
= \quad & -\frac{n\pi_l^t\Delta^2}{2}
\end{aligned}$$

The second inequality from the bottom is achieved by applying Taylor expansion on $\ln(1-\Delta)$. By iterating l with all k components, we have $L(\mathbf{\Theta}, \mathbf{\Phi}^{t-1}) \leq L(\mathbf{\Theta}^t, \mathbf{\Phi}^{t-1}) - \min_j n\pi_l^t\Delta^2/2$.

If $\mathbf{\Theta}$ satisfies condition (2) for some component l, $L(\mathbf{\Theta}, \mathbf{\Phi}^{t-1})$ can be maximized similarly. We have

$$\begin{aligned}
& L(\mathbf{\Theta}, \mathbf{\Phi}^{t-1}) - L(\mathbf{\Theta}^t, \mathbf{\Phi}^{t-1}) \\
\leq \quad & n\pi_l^t \ln(1+\Delta) + n(1-\pi_l^t) \ln \frac{1-(1+\Delta)\pi_l^t}{1-\pi_l^t} \\
= \quad & n\pi_l^t \ln(1+\Delta) + n(1-\pi_l^t) \ln \left(1 - \frac{\Delta\pi_l^t}{1-\pi_l^t}\right) \\
\leq \quad & n\pi_l^t \left(\Delta - \frac{\Delta^2}{2} + \frac{\Delta^3}{3}\right) + n(1-\pi_l^t)\frac{\Delta\pi_l^t}{1-\pi_l^t} \\
\leq \quad & -\frac{n\pi_l^t\Delta^2}{6}
\end{aligned}$$

Again, the third inequality from the bottom is due to Taylor expansion of $\ln(1+\Delta)$. The last inequality is because $\Delta^3 \leq \Delta^2$ for any $0 \leq \Delta \leq 1$.

If $\mathbf{\Theta}$ satisfies condition (3) for some component l, L is maximized if all other parameters remain the same. Thus,

$$\begin{aligned}
& L(\mathbf{\Theta}, \mathbf{\Phi}^{t-1}) - L(\mathbf{\Theta}^t, \mathbf{\Phi}^{t-1}) \\
\leq \quad & \frac{n\pi_l^t}{2} \left(\ln|\mathbf{\Sigma}_l^{-1}\mathbf{\Sigma}_l^t| - tr\left(\left(\mathbf{\Sigma}_l^{-1} - (\mathbf{\Sigma}_l^t)^{-1}\right)\mathbf{\Sigma}_l^t\right)\right) \\
= \quad & \frac{n\pi_l^t}{2} \left(tr\left(\log\left(\mathbf{\Sigma}_l^{-1}\mathbf{\Sigma}_l^t\right)\right) - tr\left(\mathbf{\Sigma}_l^{-1}\mathbf{\Sigma}_l^t - I\right)\right) \\
\leq \quad & \frac{n\pi_l^t}{2} \left(-\frac{tr(\mathbf{\Sigma}_l^{-1}\mathbf{\Sigma}_l^t - I)^2}{2}\right) \\
= \quad & -\frac{n\pi_l^t\Delta^2}{4}
\end{aligned}$$

The fourth equality is derived by the definitions of $\mathbf{\Sigma}_l^t$ and π_l^t. And the second inequality from bottom is due to the taylor expansion on the logarithm matrix.

Finally, if $\mathbf{\Theta}$ satisfies condition (4) for some component l, L is maximized if $\mathbf{\Sigma}_l = \frac{\sum \tau_{il}(\boldsymbol{x}_i - \boldsymbol{\mu}_l)(\boldsymbol{x}_i - \boldsymbol{\mu}_l)^T}{\sum \tau_{il}}$. In this case, $\sum_i \tau_l^{t-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_j)^T\mathbf{\Sigma}_j^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_j) = \sum_i \tau_l^{t-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_j^t)^T(\mathbf{\Sigma}_j^t)^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_j^t) = n\pi_j d$. Thus, the only difference on the log likelihood function L stems from the change on the determinant of the covariance matrix.

$$\begin{aligned}
& L(\mathbf{\Theta}, \mathbf{\Phi}^{t-1}) - L(\mathbf{\Theta}^t, \mathbf{\Phi}^{t-1}) \\
\leq \quad & \sum_{i=1}^{n} \frac{\tau_{il}}{2}\left(-\ln|\mathbf{\Sigma}_l| + \ln|\mathbf{\Sigma}_l^t|\right) \\
= \quad & \sum_{i=1}^{n} \frac{\tau_{il}}{2}\left(-\ln|\mathbf{\Sigma}_l^t + (\boldsymbol{\mu}_l - \boldsymbol{\mu}_l^t)(\boldsymbol{\mu}_l - \boldsymbol{\mu}_l^t)^T)| + \ln|\mathbf{\Sigma}_l^t|\right) \\
\leq \quad & \sum_{i=1}^{n} \frac{\tau_{il}}{2}\left(-\ln\left(|\mathbf{\Sigma}_l^t| + |(\boldsymbol{\mu}_l - \boldsymbol{\mu}_l^t)(\boldsymbol{\mu}_l - \boldsymbol{\mu}_l^t)^T|\right) + \ln|\mathbf{\Sigma}_l^t|\right) \\
\leq \quad & \sum_{i=1}^{n} \frac{\tau_{il}}{2}\left(-\ln\left(|\mathbf{\Sigma}_l^t| + \Delta^2|\mathbf{\Sigma}_l^t|\right) + \ln|\mathbf{\Sigma}_l^t|\right) \\
= \quad & -\sum_{i=1}^{n} \frac{\tau_{il}\ln(1+\Delta^2)}{2} \\
\leq \quad & -\frac{n\pi_l^t\Delta^2}{2}
\end{aligned}$$

The fourth inequality applies the property of positive definite matrices that $|A+B| > |A| + |B|$ (Lutkepohl, 1996) .

In all of the four cases, the reduction on the likelihood function L is at least $-\frac{n \min_j \pi_j^t \Delta^2}{6}$. This completes the proof of the theorem. □

Last theorem implies that $\mathbf{\Theta}$ will reduce the log likelihood function by at least $n \min_j^t \pi_j \Delta^2/6$ if $\mathbf{\Phi}$ remains $\mathbf{\Phi}^{t-1}$. The following question is how much we can increase the likelihood if we use the optimal $\Psi(\mathbf{\Theta})$ instead of $\mathbf{\Phi}^{t-1}$.

Lemma 3 *Given* $\boldsymbol{\Theta} \in R(\boldsymbol{\Theta}^t, \Delta)$*,* $Pr(\boldsymbol{x}_i|\theta_j)\pi_j$ *is no larger than* $(1+\Delta)^{1.5}\frac{|(\boldsymbol{\Sigma}_j^t)^{-1}|^{1/2}}{(2\pi)^{d/2}}\exp\{-(1-\Delta)M(\boldsymbol{x},\boldsymbol{\Theta}_j)^2/2\}\pi_j^t$*, where* $M(\boldsymbol{x},\boldsymbol{\Theta}_j)$ *is*

$$\max\left\{\sqrt((x-\boldsymbol{\mu}_j^t)^T(\boldsymbol{\Sigma}_j^t)^{-1}(x-\boldsymbol{\mu}_j^t))-\Delta, 0\right\}$$

Lemma 4 *Given* $\boldsymbol{\Theta} \in R(\boldsymbol{\Theta}^t, \Delta)$*,* $Pr(\boldsymbol{x}_i|\theta_j)\pi_j$ *is no smaller than* $(1-\Delta)^{1.5}\frac{|(\boldsymbol{\Sigma}_j^t)^{-1}|^{1/2}}{(2\pi)^{d/2}}\exp\{-(1+\Delta)N(\boldsymbol{x},\boldsymbol{\Theta}_j)^2/2\}\pi_j^t$*, where* $N(\boldsymbol{x},\boldsymbol{\Theta}_j)$ *is*

$$\sqrt((x-\boldsymbol{\mu}_j^t)^T(\boldsymbol{\Sigma}_j^t)^{-1}(x-\boldsymbol{\mu}_j^t))+\Delta$$

.

The proofs of Lemma 3 and Lemma 4 are available in (Zhang et al., 2008).

Lemma 5 *Given a region* $R(\boldsymbol{\Theta}^t, \Delta)$ *as defined above, an upper bound,* U_{ij}*, on* $\tau_{ij} \in \Psi(\boldsymbol{\Theta})$ *for any* $\boldsymbol{\Theta} \in R(\boldsymbol{\Theta}^t, \Delta)$ *can be calculated in constant time.*

Proof: For any configuration $\boldsymbol{\Theta}$ on the boundary of $R(\boldsymbol{\Theta}^t, \Delta)$, the optimal value of τ_{ij} can be calculated by equation (2). By Lemma 3 and Lemma 4, we can compute $\max_{\boldsymbol{\Theta}} \Pr(\boldsymbol{x}_i|\omega_j)\pi_j$ and $\min_{\boldsymbol{\Theta}} \Pr(\boldsymbol{x}_i|\omega_j)\pi_j$. Therefore,

$$\tau_{ij} \le U_{ij} = \frac{\max_{\boldsymbol{\Theta}} \Pr(\boldsymbol{x}_i|\omega_j)\pi_j}{\sum_l \min_{\boldsymbol{\Theta}} \Pr(\boldsymbol{x}_i|\omega_l)\pi_j}$$

$$\tau_{ij} \ge L_{ij} = \frac{\min_{\boldsymbol{\Theta}} \Pr(\boldsymbol{x}_i|\omega_j)\pi_j}{\sum_l \max_{\boldsymbol{\Theta}} \Pr(\boldsymbol{x}_i|\omega_l)\pi_j}$$

The calculations can be finished in constant time with the two sums pre-computed. □

By Lemma 5, the increase upper bound from $L(\boldsymbol{\Theta}, \boldsymbol{\Phi}^{t-1})$ to $L(\boldsymbol{\Theta}, \Psi(\boldsymbol{\Theta}))$ can be calculated by the following equation.

$$\begin{aligned} & L(\boldsymbol{\Theta}, \Psi(\boldsymbol{\Theta})) - L(\boldsymbol{\Theta}, \boldsymbol{\Phi}^{t-1}) \\ \le\ & \ln \sum\sum U_{ij} \max_{\boldsymbol{\Theta}} \Pr(\boldsymbol{x}_i|\omega_j)\pi_j - \\ & \ln \sum\sum \tau_{ij} \max_{\boldsymbol{\Theta}} \Pr(\boldsymbol{x}_i|\omega_j)\pi_j \qquad (8) \end{aligned}$$

The following theorem gives a sufficient condition on a maximal region $R(\boldsymbol{\Theta}^t, \Delta)$ for some positive value Δ.

Theorem 3 $R(\boldsymbol{\Theta}^t, \Delta)$ *is a maximal region for* $\boldsymbol{\Theta}^t$ *if* $\ln \frac{\sum_i \sum_j U_{ij} \max_{\boldsymbol{\Theta}} \Pr(\boldsymbol{x}_i|\omega_j)\pi_j}{\sum_i \sum_j \tau_{ij} \max_{\boldsymbol{\Theta}} \Pr(\boldsymbol{x}_i|\omega_j)\pi_j} - n \min \pi_j^t \Delta^2/6 < L(\boldsymbol{\Theta}^t, \boldsymbol{\Phi}^t) - L(\boldsymbol{\Theta}^t, \boldsymbol{\Phi}^{t-1})$

Proof: By the definition of L, we have

$$L(\boldsymbol{\Theta}, \Psi(\boldsymbol{\Theta})) - L(\boldsymbol{\Theta}^t, \boldsymbol{\Phi}^{t-1}) \le \ln \frac{\sum_i \sum_j U_{ij} \Pr(\boldsymbol{x}_i|\omega_j)\pi_j}{\sum_i \sum_j \tau_{ij} \Pr(\boldsymbol{x}_i|\omega_j)\pi_j}$$

It is not hard to verify that the derivative of $L(\boldsymbol{\Theta}, \Psi(\boldsymbol{\Theta})) - L(\boldsymbol{\Theta}^t, \boldsymbol{\Phi}^{t-1})$ to $\Pr(\boldsymbol{x}_i|\omega_j)\pi_j$ is always positive. Therefore, the equation above can be maximized if we employ the maximum value of $\Pr(\boldsymbol{x}_i|\omega_j)\pi_j$. Based on the analysis above, we know that

$$\begin{aligned} & L(\boldsymbol{\Theta}, \Psi(\boldsymbol{\Theta})) - L(\boldsymbol{\Theta}^t, \boldsymbol{\Phi}^{t-1}) \\ \le\ & \ln \frac{\sum_i \sum_j U_{ij} \max_{\boldsymbol{\Theta}} \Pr(\boldsymbol{x}_i|\omega_j)\pi_j}{\sum_i \sum_j \tau_{ij} \max_{\boldsymbol{\Theta}} \Pr(\boldsymbol{x}_i|\omega_j)\pi_j} \end{aligned}$$

By Theorem 2, $L(\boldsymbol{\Theta}, \Phi^{t-1}) - L(\boldsymbol{\Theta}^t, \Phi^{t-1}) \le n \min_j \pi_j^t \Delta^2/6$. Therefore, by Theorem 1, $L(\boldsymbol{\Theta}, \Psi(\boldsymbol{\Theta})) < L(\boldsymbol{\Theta}^t, \boldsymbol{\Phi}^t)$ if the condition of the theorem is satisfied. □

For any local optimum $\boldsymbol{\Theta}^*$ in the maximal region $R(\boldsymbol{\Theta}^t, \Delta)$, the following theorem upper bound the likelihood function $L(\boldsymbol{\Theta}^*, \Psi(\boldsymbol{\Theta}^*))$.

Theorem 4 *Given a valid maximal region* $R(\boldsymbol{\Theta}^t, \Delta)$*, if EM algorithm converges to local optimum* $\boldsymbol{\Theta}^*$*,* $L(\boldsymbol{\Theta}^*, \Psi(\boldsymbol{\Theta}^*)) \le L(\boldsymbol{\Theta}^t, \boldsymbol{\Phi}^t) + n \min \pi_j^t \Delta^2/6$*.*

Proof: Since $\sum_i \sum_j (U_{ij} - \tau_{ij}^t) \max_{\boldsymbol{\Theta}} \Pr(\omega_j|\boldsymbol{x}_i) < n \min \pi_j^t \Delta^2/6$ by Theorem 3 and $L(\boldsymbol{\Theta}, \Psi(\boldsymbol{\Theta})) - L(\boldsymbol{\Theta}, \boldsymbol{\Phi}^t) \le \sum_i \sum_j (U_{ij} - \tau_{ij}^t) \max_{\boldsymbol{\Theta}} \Pr(\omega_j|\boldsymbol{x}_i)$, we have $L(\boldsymbol{\Theta}, \Psi(\boldsymbol{\Theta})) - L(\boldsymbol{\Theta}^t, \boldsymbol{\Phi}^t) \le L(\boldsymbol{\Theta}, \Psi(\boldsymbol{\Theta})) - L(\boldsymbol{\Theta}, \boldsymbol{\Phi}^t) \le n \min \pi_j^t \Delta^2/6$. □

5. Algorithm

Theorem 3 provides an easy way to verify whether $R(\boldsymbol{\Theta}^t, \Delta)$ is a valid maximal region. On the other hand, Theorem 4 implies that a smaller Δ can lead to tighter bound on the likelihood function L. However, it is not necessary to get the tightest bound on local optimum in our algorithm, since the goal of our algorithm is estimating whether the current configuration can lead to better solution. Instead, we set Δ as $\min\left\{1, \sqrt{\frac{6(L^* - L(\boldsymbol{\Theta}^t, \boldsymbol{\Phi}^t))}{n \min \pi_j^t}}\right\}$, where L^* is the best result we have seen so far. This Δ is the maximal one of all Δ values, which are able to prune the current EM procedure by Theorem 4

The details of the algorithm are summarized in Algo 1. In this algorithm, conventional M-step and E-step

Algorithm 1 New Iteration(Data Set D, Current $\mathbf{\Theta}^{t-1}$, current $\mathbf{\Phi}^{t-1}$, component number k, sample number m, current best result L^*)

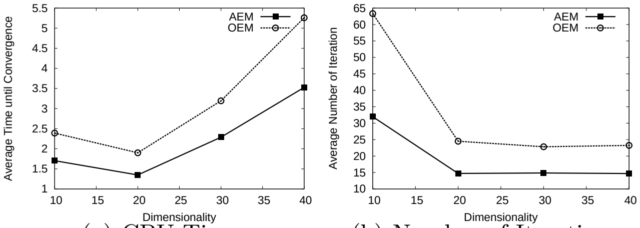
```
Compute new $\mathbf{\Theta}^t$ by M-Step.
Compute new $\mathbf{\Phi}^t$ by E-Step.
if $L(\mathbf{\Theta}^t, \mathbf{\Phi}^t) < L^*$ then
    Let $\Delta = \min \left\{ 1, \sqrt{\frac{6(L^* - L(\mathbf{\Theta}^t, \mathbf{\Phi}^t))}{n \min \pi_j^t}} \right\}$
    $S = X = 0$
    for each $\boldsymbol{x}_i$ do
        for each dimension $j$ do
            Get $l_{ij} = \max_{\mathbf{\Theta}} \Pr(\boldsymbol{x}_i|\theta_j)\pi_j$ by Lemma 3.
            Get $s_{ij} = \min_{\mathbf{\Theta}} \Pr(\boldsymbol{x}_i|\theta_j)\pi_j$ by Lemma 4.
            Get $U_{ij}$ by Lemma 5.
            $S+ = U_{ij} * l_{ij}$
            $X+ = \tau_{ij}^{t-1} * l_{ij}$
        end for
    end for
    if $\ln S - \ln X - n \min \pi_j \Delta^2/6 < L(\mathbf{\Theta}^t, \mathbf{\Phi}^t) - L(\mathbf{\Theta}^t, \mathbf{\Phi}^{t-1})$ then
        Stop the current procedure of EM algorithm.
    end if
else
    Return $(\mathbf{\Theta}^t, \mathbf{\Phi}^t)$
end if
```

are invoked first. If the current configuration is better than the best solution we have seen before, there is no need to test the upper bound of the local optimum. Otherwise, the value of Δ is set according to $\min \pi_j^t$, L^* and $L(\mathbf{\Theta}^T, \mathbf{\Phi}^t)$. For each point and each component, l_{ij}, s_{ij} and U_{ij} are collected according to Lemma 3, Lemma 4 and Lemma 5 respectively. With the information collected from each point, the condition of Theorem 1 can be tested. If this condition is satisfied, we can assert that current local optimum can never be better than L^*, leading to the termination of the current procedure.

6. Experiments

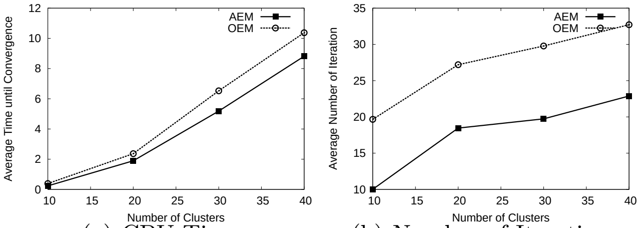
In this section, we report the experimental results on the comparison of our accelerated EM algorithm (AEM) and the conventional EM algorithm (OEM). We note that in our implementation, either AEM or OEM will be stopped if it does not converge after 100 iterations.

We employ both synthetic and real data sets in our empirical studies. The synthetic data sets are generated in a d-dimensional unit cube. There are k components in the space. Each component follows some Gaussian distribution. The center, size and covariance matrix of each component are randomly generated independently. Two real data sets are also tested, including *Cloud* and *Spam*, both of which are available on UCI Machine Learning Repository. The *Cloud* data set consists of 1024 points in 10-dimensional space, while *Spam* data set has 4601 points in 58 dimensions. Both of the real data set are normalized before being used in our experiments.



(a) CPU Time (b) Number of Iterations

Figure 1: Performance vs. varying dimensionality



(a) CPU Time (b) Number of Iterations

Figure 2: Performance vs. varying component number

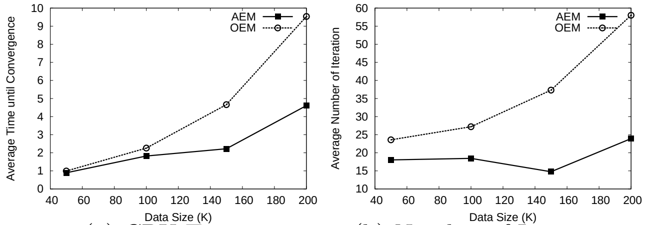
Two performance measurements are recorded in our experiments, including CPU time and number of iterations. An algorithm is supposed to be better if it spends less CPU time and invokes less time of iterations. All of the experiments are compiled and run on a Fedora Core 6 linux machine with 3.0 GHz Processor, 1GB of memory and GCC 4.1.2.

In the experiments on the data sets, we test the performances of the algorithms with varying dimensionality D, number of components k, and the number of points in the data S. The default setting of our experiments is $D = 20$, $k = 20$, and $S = 100$K. The time of EM restart is fixed at 100 in all tests. More experimental results are available in the technical report (Zhang et al., 2008).

6.1. Results on Synthetic Data

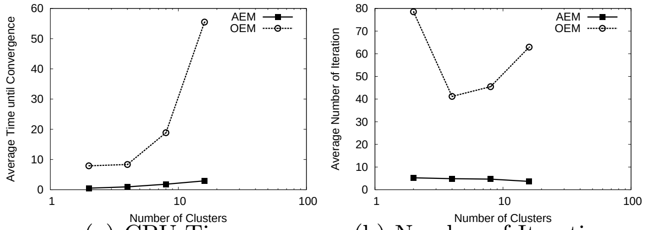
In Figure 1(a) and Figure 1(b), we present the experimental result by varying the dimensionality from 10 to 40. The results show that AEM is much more efficient than OEM. On data set with low dimensionality, AEM is almost two times faster than OEM, both on the CPU time and the number of iterations. The advantage is very obvious, even on high dimensional space.

The results of our experiments on varying component



(a) CPU Time (b) Number of Iterations

Figure 3: Performance vs. varying data size



(a) CPU Time (b) Number of Iterations

Figure 4: Performance vs. varying component number on *Spam* data

number are summarized in Figure 2(a) and Figure 2(b). From the figures, we can see the performance advantage of AEM is stable, with the increase of component number. The CPU time and number of iterations on AEM is only about half of those of OEM.

As is shown in Figure 3(a), Figure 3(b) AEM has much better performance than OEM when we increase the data size from 50K to 200K. AEM can detect those worse local optimums much earlier, if there are more data available. The number of iterations invoked by AEM is almost the same, even when the data has been doubled. The ratio of CPU time is more stable when the data size is larger.

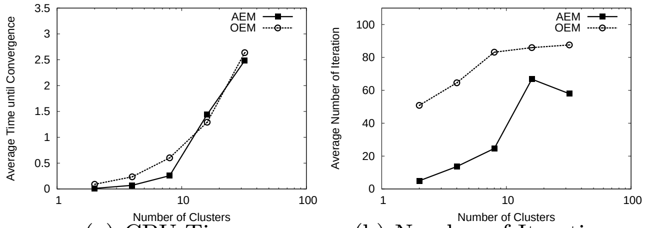
6.2. Results on Real Data

On *Spam* data set, AEM also show great advantage over OEM, on CPU time (Figure 4(a)) and on the number of iterations (Figure 4(b)). AEM is more efficient than OEM by one magnitude, independent to the number of components k.

However, the experiments on *Cloud* data set show quite different results than the pervious results, where AEM has very limited advantage. We believe the difference on the results stems from normalization problem.

7. Conclusion

In this paper, we propose a new acceleration method for multiple restart EM algorithm over Gaussian Mixture Model. We derive an upper bound on the local optimum of the likelihood function in the solution



(a) CPU Time (b) Number of Iterations

Figure 5: Performance vs. varying component number on *Cloud* data

space. This upper bound computation turns out to be both efficient and effective in pruning un-promising procedures of EM algorithm.

Acknowledgement: The authors would like to acknowledge the valuable comments from the reviewers of ICML 2008.

References

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

Elkan, C. (2003). Using the triangle inequality to accelerate k-means. *ICML* (pp. 147–153).

Jordan, M. I., & Xu, L. (1995). Convergence results for the em approach to mixtures of experts architectures. *Neural Networks*, *8*, 1409–1431.

Kanungo, T., Mount, D., Netanyahu, N., Piatko, C., Silverman, R., & Wu, A. (2002). An efficient k-means clustering algorithm: analysis and implementation.

Lutkepohl, H. (1996). *Handbook of matrices.* John Wiley & Sons Ltd.

Ma, J., Xu, L., & Jordan, M. I. (2001). Asymptotic convergence rate of the em algorithm for gaussian mixtures. *Neural Computation*, *12*, 2881–2907.

McLachlan, G., & Krishnan, T. (1996). *The em algorithm and extensions.* Wiley-Interscience.

McLachlan, G., & Peel, D. (2000). *Finite mixture models.* Wiley-Interscience.

Xu, L., & Jordan, M. I. (1996). On convergence properties of the em algorithm for gaussian mixtures. *Neural Computation*, *8*, 129–151.

Zhang, Z., Dai, B. T., & Tung, A. K. H. (2006). On the lower bound of local optimums in k-means algorithm. *ICDM* (pp. 775–786).

Zhang, Z., Dai, B. T., & Tung, A. K. H. (2008). Estimating local optimums in em algorithm over gaussian mixture model. *http://www.comp.nus.edu.sg/∼zhangzh2/papers/em.pdf.*