ISSN: 0128-7680
Pertanika J. Sci. & Technol. **17 (2)**: 231 – 243 (2009)

The Performance of Expectation Maximization (EM) Algorithm in Gaussian Mixed Models (GMM)

Mohd. Izhan Mohd. Yusoff[1*], Mohd. Rizam Abu Bakar[2] and Abu Hassan Shaari Mohd. Nor[3]

[1]*Operation Support Systems Program, Applied Research Division, Telekom Research & Development Sdn Bhd, TMR&D Innovation Centre, Lingkaran Teknokrat Timur, 63000 Cyberjaya, Selangor, Malaysia*

[2]*Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia*

[3]*Faculty of Economics and Business, Universiti Kebangsaan Malaysia, 43600 UKM, Bangi, Selangor, Malaysia*

[*]*E-mail: izhan@tmrnd.com.my*

ABSTRACT

Expectation Maximization (EM) algorithm has experienced a significant increase in terms of usage in many fields of study. In this paper, the performance of the said algorithm in finding the Maximum Likelihood for the Gaussian Mixed Models (GMM), a probabilistic model normally used in fraud detection and recognizing a person's voice in speech recognition field, is shown and discussed. At the end of the paper, some suggestions for future research works will also be given.

Keywords: Expectation Maximization (EM), Gaussian Mixed Models (GMM), Box and Muller Transformation

INTRODUCTION

Every year, telecommunication companies register heavy loses due to fraud activities amounting to million of dollars. Vendors, seeing the above as an opportunity not to be missed, compete to provide data mining applications which could detect the said activity effectively using methods such as OLAP, deviation based outlier detection, Hidden Markov Model, and the model which became the focal area of this paper, the Gaussian Mixed Models (GMM).

GMM is best known in providing a robust speaker representation for the difficult task of speaker identification on *short-time speech spectra*, which is a cosine, transformed of log energy filter outputs from processed magnitude spectrum from a 20 ms short time segment of speech, by simulated me-scale filter-bank (Reynolds *et al*., 1995). Its function is further extended to detect fraud activities on daily number of calls and length of calls occurring during the office hours, the evening hours and the night hours for both national and international calls (Mohd Yusoff *et al*., 2006; Tanigushi *et al*., 1998).

Maximum likelihood estimation for GMM is difficult to find and the solution is Expectation Maximization (EM) algorithm. The EM algorithm was first introduced by Dempster *et al*. (1977) and since then, there has been a significant increase in terms of its usage, particularly in finding the

Received: 5 September 2007
Accepted: 27 February 2009
[*]Corresponding Author

Maximum Likelihood for probabilistic models (such as missing data, grouping, censoring, truncation, and finite mixtures). The main issue with respect to the EM algorithm is finding the right choice of initial parameters and the number of components. This particular issue or problem is illustrated in several examples in this paper.

The subsequent sections provide a brief introduction of the EM algorithm and GMM, generate the simulation univariate and multivariate data with clear and hidden components, present the results gathered from the GMM and EM algorithm where the emphasis given on the choice of the initial parameters and the number of components, and some suggestions for future research works.

The Gaussian Mixed Models (GMM) and Expectation Maximization (EM) Algorithm

Let $x \in R^d$ and K be the number of components where each component having its own prior probability (weight) a and probability density function with the mean μ and covariance Σ. All of them are mixed resulting in the following formula, which is also known as the Gaussian Mixed Models (GMM):

$$\sum_{i=1}^{K} a_i \phi(\mathrm{x} \mid \mu_i, \Sigma_i) = \sum_{i=1}^{K} a_i \frac{1}{\sqrt{(2\pi)^d \mid \Sigma_i \mid}} \exp\left(\frac{-(\mathrm{x}-\mu_i)^t \Sigma_i^{-1} (\mathrm{x}-\mu_i)}{2} \right) \tag{1}$$

where prior probability (weight) of component i that is a_i satisfy the constraint $\sum_{i=1}^{K} a_i = 1$.

For the case of voice recognition, assuming there are n number of speakers. The m number of samples were collected from each speaker. Equation (1) is derived for each sample, where its parameters were kept in the database for comparison purposes. Fraud detection would follow similar steps.

From equation (1), the likelihood function and log likelihood function by $L(\mathrm{X} \mid \theta) = \prod_{j=1}^{n} f(\mathrm{x}_j \mid \theta)$ and, $l(\mathrm{X} \mid \theta) = \log L(\mathrm{X} \mid \theta) = \sum_{j=1}^{n} \log\left(\sum_{i=1}^{K} a_i \phi(\mathrm{x}_j \mid \mu_i, \Sigma_i) \right)$ were defined, respectively. The maximum likelihood estimation (m.l.e) aimed at finding $\hat{\theta}$ which maximized $l(x|\theta)$, with respect to θ (Mardia *et al*., 1979). The expression $\log\left(\sum_{i=1}^{K} a_i \phi(\mathrm{x}_j \mid \mu_i, \Sigma_i) \right)$ in the log likelihood function is difficult to solve, and in order to overcome this problem, the Expectation Maximization (EM) algorithm was used.

In the EM algorithm, the distribution of X needs to be estimated in the sample space χ, but X can only be observed indirectly through Y in the sample space Y. In many cases, there is a mapping $x \rightarrow y(x)$ from χ o Y, and x is only known to lie in a subset of χ, denoted by $\chi(y)$, which is determined by equation $y = y(x)$. The distribution of X is parameterized by a family of distributions $f(x|\theta)$, with parameters $\theta \in \Omega$ or x. The distribution of Y, $g(y|\theta)$ is therefore:

$$g(y \mid \theta) = \int_{\chi(y)} f(x \mid \theta) dx \tag{2}$$

The EM algorithm aims at finding θ which maximizes $g(y|\theta)$ given an observed y. Let the function

$$Q(\theta' \mid \theta) = E(\log f(x \mid \theta') \mid y, \theta) \tag{3}$$

be the expected value of $\log f(x|\ \theta')$ given y and θ. The expectation was assumed to exist for all the pairs (θ', θ). In particular, it was assumed that $f(x|\theta) > 0$ for $\theta \in \Omega$.

EM Iteration

E-Step: Compute $Q(\theta|\ \theta^{(p)})$

M-step: Choose $\theta^{(p+1)}$ to be a value of $\theta \in \Omega$ that maximizes $Q(\theta|\theta^{(p)})$ (Dempster *et al*., 1977). In the case of GMM, it was defined that $Q(\theta'|\ \theta) = E\left[\log \prod_{i=1}^{n} a'_{y_i} \phi(\mathrm{x}_i | \mu'_{y_i}, \Sigma'_{y_i}) \middle| \mathrm{X}, \theta\right]$, where $y_i \in \{1,2,\ldots, K\}$, $y_i = k$ if the i^{th} sample was generated by the k^{th} mixture component. It was simplified using (among other) the Bayes formula which is $f(\theta|x) \propto f(x|\theta)P(\theta)$, where $f(\theta|x)$ = posterior probability, $f(x|\theta)$ = likelihood function, and $P(\theta)$ = prior probability (Tsay, 2005; Bilmes, 1997) to the following equations:

$$Q(\theta'|\theta) = \sum_{i=1}^{n}\sum_{k=1}^{K} p_{i,k} \log a'_k + \sum_{i=1}^{n}\sum_{k=1}^{K} p_{i,k} \log \phi(\mathrm{x}_i | \mu'_k, \Sigma'_k) \tag{4}$$

where

$$p_{ik} = \frac{a_k \phi(\mathrm{x}_i | \mu_k, \Sigma_k)}{\sum_l a_l \phi(\mathrm{x}_i | \mu_l, \Sigma_l)} \tag{5}$$

and

$$\phi(\mathrm{x}_i | \mu_k, \Sigma_k) = \frac{1}{\sqrt{(2\pi)^d |\Sigma_k|}} \exp\left(\frac{-(\mathrm{x}_i - \mu_k)^t \Sigma_k^{-1} (\mathrm{x}_i - \mu_k)}{2}\right) \tag{6}$$

The EM Iteration (for GMM)

E-Step:

Equation (5) is calculated.

M-Step:

The following formulas (derived from the Lagrange multipliers, $\frac{\partial Q}{\partial \mu_j} = 0$ and $\frac{\partial Q}{\partial \Sigma_j^{-1}} = 0$, respectively) are calculated. Further details are given in Appendix A.2.

$$a_j = \frac{1}{n}\sum_i p_{ij} \tag{7}$$

$$\mu_j = \frac{\sum_i p_{ij} \mathrm{x}_i}{\sum_i p_{ij}} \tag{8}$$

$$\Sigma_j = \frac{\sum_i p_{ij}(\mathrm{x}_i - \mu_j)(\mathrm{x}_i - \mu_j)^t}{\sum_i p_{ij}} \tag{9}$$

The above steps (i.e. E-step and M-step) were repeated until a convergence was achieved.

SIMULATION DATA

A program called "Simulate" was developed (using C++ language) to generate simulation data for equation (1) with the parameters as per given in Table 1 (taken from Everitt *et al*., 1981, with modifications) using Box and Muller Transformation (Box *et al*., 1958) and Equation (10). The simulation data were then labelled as "Sample1", "Sample2", "Sample3" and "Sample4", and their histograms are shown in *Figs. 1* and *2*.

$$z_j = \mu + (-2\sigma^2 \log u_j)^{\frac{1}{2}} \cos 2\pi u_{j+1}$$

$$z_{j+1} = \mu + (-2\sigma^2 \log u_j)^{\frac{1}{2}} \sin 2\pi u_{j+1}, u_j, u_{j+1} \sim U(0,1) \tag{10}$$

The "Simulate" program would generate one random number, denoted by U_1, from the uniform distribution *U(0,1)*, and check whether it was less than say a_i (i=1,2). If the answer is "yes", the two random numbers, denoted by U_2 and U_3, are generated from the uniform distribution *U(0,1)* and used in the computing equation (10), along with the corresponding μ_i and σ_{ii}, taken from Table 1. In this study, these steps were repeated until 1000 observations were obtained. For "Sample 4", apart from equation (10), the formulas given in Appendix A.1 (in the matrix format) were also used.

In *Fig. 1.1*, two humps are observed and these represent two components: $(\mu_1, \sigma_{11}) = (0.0, 1.0)$ and $(\mu_2, \sigma_{22}) = (2.0, 0.25)$. Both of them are well-separated, in which the observations for the latter component are grouped around the mean.

One would never expect to find the two components in *Fig. 1.2*. The histogram is dominated by the component $(\mu_1, \sigma_{11}) = (0.0, 1.0)$ due to the fact that $a_1 = 0.85$.

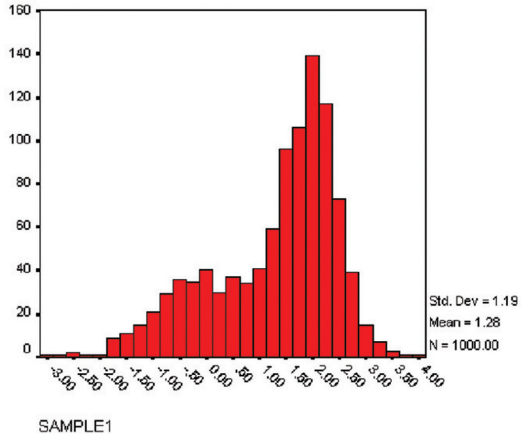
In *Fig. 1.3*, two humps are vividly displayed and they represent two components: $(\mu_2, \sigma_{22}) = (-1.0, 0.25)$ and $(\mu_3, \sigma_{33}) = (4.0, 4.0)$. The third component, $(\mu_1, \sigma_{11}) = (0.0, 1.0)$, is hidden from the view by the two components indicated earlier. The observations are grouped around the mean for the component $(\mu_2, \sigma_{22}) = (-1.0, 0.25)$.

The histograms in *Figs. 2.1* and *2.2* appear to split into two representing components $(\mu_{11}, \mu_{21}) = (5.01, 5.91)$ and $(\mu_{22}, \mu_{32}) = (2.78, 2.95)$, respectively; whereas *Figs. 2.3 and 2.4* into three representing components $(\mu_{13}, \mu_{23}, \mu_{33}) = (1.46, 4.2, 5.48)$ and $(\mu_{14}, \mu_{24}, \mu_{34}) = (0.25, 1.3, 1.98)$, respectively.

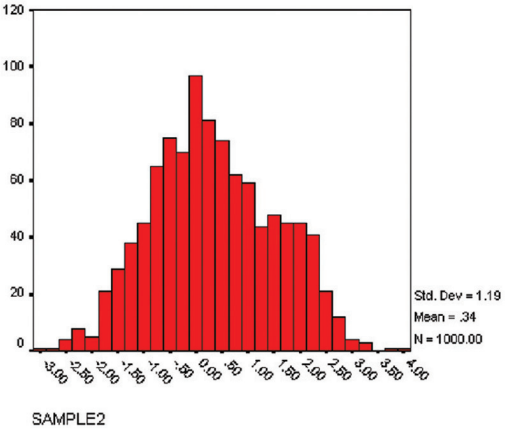
TABLE 1

a's, μ's, $\boldsymbol{\mu}$'s, $\boldsymbol{\sigma}$'s and $\boldsymbol{\Sigma}$'s for each sample used in the "Simulate" program. The number of observations generated by the program is given in the bracket

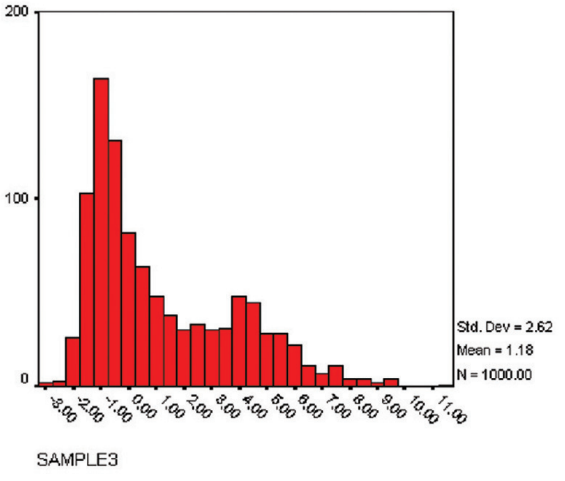
"Sample1" (N=1000)	$a_1=0.4$ $a_2=0.6$	$\mu_1=0.0$ $\mu_2=2.0$	$\sigma_{11}=1.0$ $\sigma_{22}=0.25$
"Sample2" (N=1000)	$a_1=0.85$ $a_2=0.15$	$\mu_1=0.0$ $\mu_2=2.0$	$\sigma_{11}=1.0$ $\sigma_{22}=0.25$
"Sample3" (N=1000)	$a_1=0.33$ $a_2=0.33$ $a_3=0.34$	$\mu_1=0.0$ $\mu_2=-1.0$ $\mu_3=4.0$	$\sigma_{11}=1.0$ $\sigma_{22}=0.25$ $\sigma_{33}=4.0$
"Sample4" (N=1000)	$a_1=0.33$	$\mu_1 = \begin{bmatrix} 5.01 \\ 3.43 \\ 1.46 \\ 0.25 \end{bmatrix}$	$\Sigma_1 = \begin{bmatrix} 0.12 & 0.1 & 0.02 & 0.01 \\ & 0.14 & 0.01 & 0.13 \\ & & 0.03 & 0.01 \\ & & & 0.3 \end{bmatrix}$
	$a_2=0.30$	$\mu_2 = \begin{bmatrix} 5.91 \\ 2.78 \\ 4.2 \\ 1.3 \end{bmatrix}$	$\Sigma_2 = \begin{bmatrix} 0.27 & 0.1 & 0.18 & 0.05 \\ & 0.09 & 0.09 & 0.04 \\ & & 0.2 & 0.06 \\ & & & 0.03 \end{bmatrix}$
	$a_3=0.37$	$\mu_3 = \begin{bmatrix} 6.54 \\ 2.95 \\ 5.48 \\ 1.98 \end{bmatrix}$	$\Sigma_3 = \begin{bmatrix} 0.38 & 0.09 & 0.3 & 0.06 \\ & 0.11 & 0.08 & 0.05 \\ & & 0.32 & 0.07 \\ & & & 0.08 \end{bmatrix}$



(1.1)



(1.2)



(1.3)

Fig. 1: The histograms of "Sample1" (with overall mean and standard deviation equal to 1.28 and 1.19, respectively); "Sample2" (with overall mean and standard deviation equal to 0.34 and 1.19, respectively); and "Sample3" (with overall mean and standard deviation equal to 1.18 and 2.62, respectively)

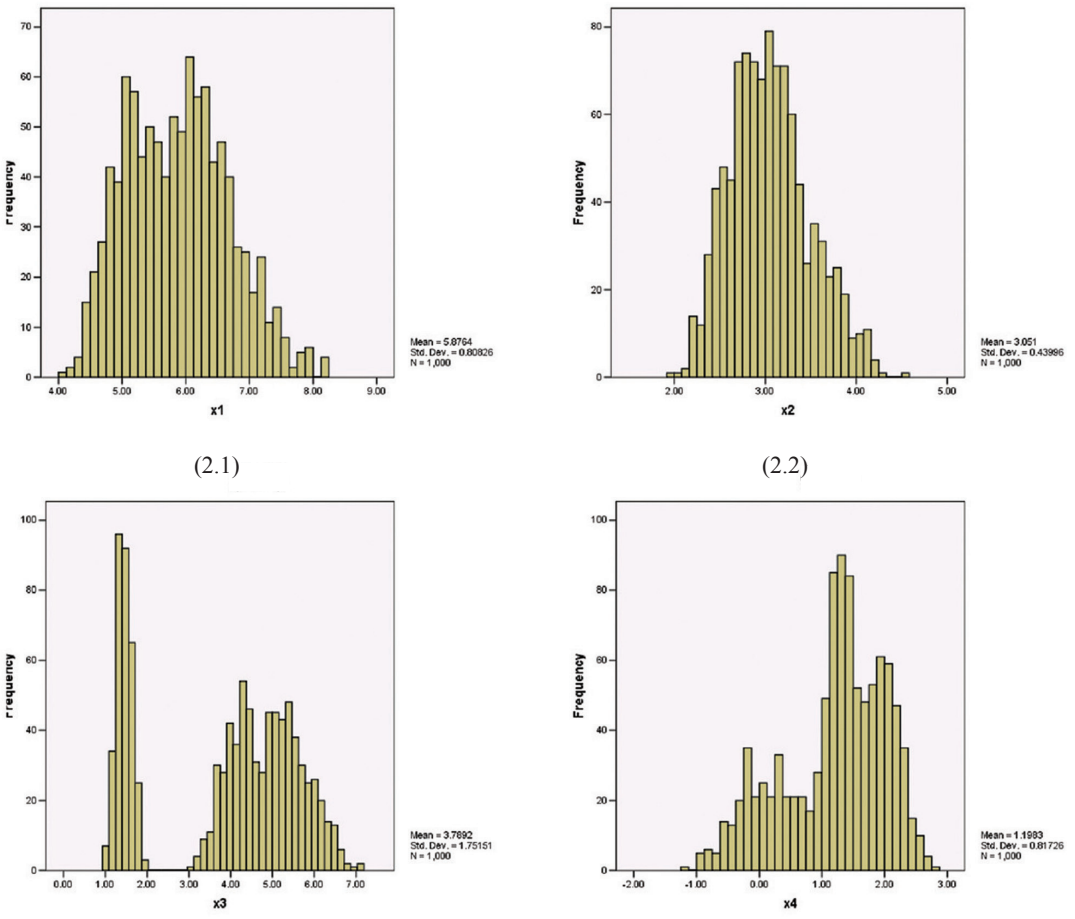


Fig. 2: The histograms of "Sample4" x_1 (with overall mean and standard deviation equal to 5.88 and 0.81, respectively); x_2 (with overall mean and standard deviation equal to 3.05 and 0.44, respectively); x_3 (with overall mean and standard deviation equal to 3.79 and 1.75, respectively); and x_4 (with overall mean and standard deviation equal to 1.2 and 0.82, respectively)

RESULTS

A program known as the "GMM" was developed using the Java language to find the parameters of equation (1) by employing the EM algorithm, where iteration is stopped when $|\theta^{t+1}-\theta^{t}|<0.000001$. Other methods involved in the calculation of EM algorithm include the Cholesky method (Mardia *et al*., 1979). In this section, two scenarios are therefore presented.

Scenario 1: In Table 2, with the exception of "Sample4" (where initial parameters were taken from Everitt *et al*., 1981), the initial parameters for "Sample1", "Sample2" and "Sample3" were determined using visual inspection of the histograms given in *Fig. 1.* This was done by concentrating on the observation(s) that gave the highest frequency, as shown by the components which were clearly displayed.

TABLE 2
a's, μ's, $\boldsymbol{\mu}$'s, σ*'s and* $\boldsymbol{\Sigma}$*'s* for each sample used in the GMM program, where they were treated as the initial parameters

"Sample1"	a_1=*0.5*	μ_1=0.0	σ_{11}=1.0
	a_2=*0.5*	μ_2=2.0	σ_{22}=1.0
"Sample2"	a_1=*0.5*	μ_1=0.0	σ_{11}=1.0
	a_2=*0.5*	μ_2=1.5	σ_{22}=1.0
"Sample3"	a_1=*0.33*	μ_1=0.0	σ_{11}=1.0
	a_2=*0.33*	μ_2=-1.0	σ_{22}=1.0
	a_3=*0.34*	μ_3=4.0	σ_{33}=1.0
"Sample4"	a_1=0.33	$\mu_1 = \begin{bmatrix} 4 \\ 4 \\ 2 \\ 1 \end{bmatrix}$	$\Sigma_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ & 1 & 0 & 0 \\ & & 1 & 0 \\ & & & 1 \end{bmatrix}$
	a_2=0.30	$\mu_2 = \begin{bmatrix} 7 \\ 2 \\ 3 \\ 2 \end{bmatrix}$	$\Sigma_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ & 1 & 0 & 0 \\ & & 1 & 0 \\ & & & 1 \end{bmatrix}$
	a_3=0.37	$\mu_3 = \begin{bmatrix} 8 \\ 4 \\ 5 \\ 3 \end{bmatrix}$	$\Sigma_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ & 1 & 0 & 0 \\ & & 1 & 0 \\ & & & 1 \end{bmatrix}$

The values given in Table 2 were used by the "GMM" program as the initial parameters to find the final ones for the simulation data, as shown in *Figs. 1* and *2*. The results are as tabulated below.

TABLE 3
$\hat{a}$'s, $\hat{u}$'s, $\hat{u}$'s, $\hat{\sigma}$'s and $\hat{\Sigma}$'s for each sample produced by the "GMM" program, using (Table 2) as the initial parameters. The "GMM" program converged is given in the bracket

"Sample1" *(Converged at iteration no 156)*	$\hat{a}_1$=0.34 $\hat{a}_2$=0.66	$\hat{\mu}_1$=-0.05 $\hat{\mu}_2$=1.98	$\hat{\sigma}_{11}$=0.85 $\hat{\sigma}_{22}$=0.28
"Sample2" *(Converged at iteration no 389)*	$\hat{a}_1$=0.82 $\hat{a}_2$=0.18	$\hat{\mu}_1$=-0.02 $\hat{\mu}_2$=1.97	$\hat{\sigma}_{11}$=0.93 $\hat{\sigma}_{22}$=0.30
"Sample3" *(Converged at iteration no 316)*	$\hat{a}_1$=0.29 $\hat{a}_2$=0.31 $\hat{a}_3$=0.40	$\hat{\mu}_1$=-0.1 $\hat{\mu}_2$=-1.01 $\hat{\mu}_3$=3.76	$\hat{\sigma}_{11}$=0.89 $\hat{\sigma}_{22}$=0.22 $\hat{\sigma}_{33}$=4.71
"Sample4" *(converged at iteration no 32)*	$\hat{a}_1$=0.32 $\hat{a}_2$=0.33 $\hat{a}_3$=0.35	$\hat{\boldsymbol{\mu}}_1 = \begin{bmatrix} 5.05 \\ 3.46 \\ 1.45 \\ 0.23 \end{bmatrix}$ $\hat{\boldsymbol{\mu}}_2 = \begin{bmatrix} 5.93 \\ 2.77 \\ 4.22 \\ 1.30 \end{bmatrix}$ $\hat{\boldsymbol{\mu}}_3 = \begin{bmatrix} 6.58 \\ 2.94 \\ 5.53 \\ 1.99 \end{bmatrix}$	$\hat{\boldsymbol{\Sigma}}_1 = \begin{bmatrix} 0.13 & 0.11 & 0.03 & 0.02 \\ & 0.15 & 0.02 & 0.14 \\ & & 0.03 & 0.01 \\ & & & 0.31 \end{bmatrix}$ $\hat{\boldsymbol{\Sigma}}_2 = \begin{bmatrix} 0.26 & 0.09 & 0.17 & 0.05 \\ & 0.08 & 0.09 & 0.04 \\ & & 0.21 & 0.06 \\ & & & 0.03 \end{bmatrix}$ $\hat{\boldsymbol{\Sigma}}_3 = \begin{bmatrix} 0.36 & 0.08 & 0.28 & 0.06 \\ & 0.10 & 0.08 & 0.05 \\ & & 0.29 & 0.06 \\ & & & 0.08 \end{bmatrix}$

It is crucial to note that for the univariate samples, the convergence was achieved with more than 100 iterations, while for the multivariate samples, less than 100 iterations were required. The choice of the initial parameters might play an important role in making the convergence process faster, as illustrated by the latter.

The "GMM" program managed to find (final) parameters even in cases where the components were hidden from the view, but this is provided that the number of components and the observations which give the highest frequency for the identifiable components are known.

Scenario 2: Great care should be taken when choosing the initial parameters (to start the EM algorithm) as well as the number of components, where wrong choice will lead to the situation exemplified in Table 4. Other examples can be found in Everitt *et al*. (1981) and Reynolds *et al*. (1995).

TABLE 4
The initial (1st row) and final (2nd row) parameters for "Sample 3" (chosen for having hidden components), where: two components were used for (4.1), four components were used for (4.2), and six components were used for (4.3) and (4.4). The actual number of the components is three

"Sample3"	a_1=0.5 a_2=0.5	μ_1=-1.0 μ_2=4.0	σ_{11}=1.0 σ_{22}=1.0
"Sample3" (Converged at iteration no 99)	$\hat{a}_1$=0.51 $\hat{a}_2$=0.49	$\hat{\mu}_1$=-0.75 $\hat{\mu}_2$=3.21	$\hat{\sigma}_{11}$=0.45 $\hat{\sigma}_{22}$=5.58

(4.1)

"Sample3"	a_1=0.25 a_2=0.25 a_3=0.25 a_4=0.25	μ_1=-1.0 μ_2=4.0 μ_3=0.0 μ_4=0.0	σ_{11}=1.0 σ_{22}=1.0 σ_{33}=1.0 σ_{44}=1.0
"Sample3" (Converged at iteration no 186)	$\hat{a}_1$=0.31 $\hat{a}_2$=0.41 $\hat{a}_3$=0.14 $\hat{a}_4$=0.14	$\hat{\mu}_1$=-1.01 $\hat{\mu}_2$=3.76 $\hat{\mu}_3$=-0.1 $\hat{\mu}_4$=-0.1	$\hat{\sigma}_{11}$=0.22 $\hat{\sigma}_{22}$=4.71 $\hat{\sigma}_{33}$=0.89 $\hat{\sigma}_{44}$=0.89

(4.2)

"Sample3"	a_1=0.17 a_2=0.17 a_3=0.17 a_4=0.17 a_5=0.17 a_6=0.17	μ_1=-1.0 μ_2=4.0 μ_3=0.0 μ_4=0.0 μ_5=0.0 μ_6=0.0	σ_{11}=1.0 σ_{22}=1.0 σ_{33}=1.0 σ_{44}=1.0 σ_{55}=1.0 σ_{66}=1.0
"Sample3" (Converged at iteration no 312)	$\hat{a}_1$=0.31 $\hat{a}_2$=0.41 $\hat{a}_3$=0.07 $\hat{a}_4$=0.07 $\hat{a}_5$=0.07 $\hat{a}_6$=0.07	$\hat{\mu}_1$=-1.01 $\hat{\mu}_2$=3.76 $\hat{\mu}_3$=-0.1 $\hat{\mu}_4$=-0.1 $\hat{\mu}_5$=-0.1 $\hat{\mu}_6$=-0.1	$\hat{\sigma}_{11}$=0.22 $\hat{\sigma}_{22}$=4.71 $\hat{\sigma}_{33}$=0.89 $\hat{\sigma}_{44}$=0.89 $\hat{\sigma}_{55}$=0.89 $\hat{\sigma}_{66}$=0.89

(4.3)

(4.3)

"Sample3"	a_1=0.17 a_2=0.17 a_3=0.17 a_4=0.17 a_5=0.17 a_6=0.17	μ_1=0.0 μ_2=0.0 μ_3=0.0 μ_4=0.0 μ_5=0.0 μ_6=0.0	σ_{11}=1.0 σ_{22}=1.0 σ_{33}=1.0 σ_{44}=1.0 σ_{55}=1.0 σ_{66}=1.0
"Sample3" (Converged at iteration no 2)	$\hat{a}_1$=0.17 $\hat{a}_2$=0.17 $\hat{a}_3$=0.17 $\hat{a}_4$=0.17 $\hat{a}_5$=0.17 $\hat{a}_6$=0.17	$\hat{\mu}_1$=1.18 $\hat{\mu}_2$=1.18 $\hat{\mu}_3$=1.18 $\hat{\mu}_4$=1.18 $\hat{\mu}_5$=1.18 $\hat{\mu}_6$=1.18	$\hat{\sigma}_{11}$=6.88 $\hat{\sigma}_{22}$=6.88 $\hat{\sigma}_{33}$=6.88 $\hat{\sigma}_{44}$=6.88 $\hat{\sigma}_{55}$=6.88 $\hat{\sigma}_{66}$=6.88

(4.4)

Notice that Table 4.2's $\hat{\mu}_3 = \hat{\mu}_4 = -0.1$ and $\hat{\sigma}_{33} = \hat{\sigma}_{44} = 0.89$ and if $\hat{a}_3 + \hat{a}_4$ were computed, 0.28 would therefore be obtained, and this is no far different from the ones given in Table 3. Table 4.3 also shows similar results, where $\hat{u}_3 = \ldots = \hat{\mu}_6 = -0.1$, $= \hat{\sigma}_{33} = \ldots = \hat{\sigma}_{66} = 0.89$ and $\hat{a}_3 = \ldots = \hat{a}_6 = 0.07$ where $\hat{a}_3 + \ldots + \hat{a}_6 = 0.28$. Despite converging at iteration no. 2 (the lowest so far), the final parameters shown in Table 4.4 are completely different from those in Table 3, and this is a direct consequence from ignoring the characteristics shown by the observations in the histograms.

CONCLUSIONS

In the previous sections, "Sample1", "Sample2", "Sample3", and "Sample4" (using a program called "Simulate") were generated with known number of both components and parameters. Using the same information, particularly on the number of components and determining the initial parameters to start the EM algorithm by inspecting the histograms, the final parameters produced from the EM algorithm (using the program known as the "GMM") are similar to the real ones.

Just to show how important the process of choosing the initial parameters is (to start the EM algorithm) and the number of components, "Sample3" was selected for having hidden components, while the process of determining the initial parameters to start EM algorithm (i.e. by inspecting the histograms) and reducing the number of components was repeated; the final parameters produced were incorrect. The same results were also obtained when the number of components was increased; for the initial parameters to start the EM algorithm, let the mean equals to 0 and the standard deviation equals to 1 (a common mistake done by most of the practitioners).

In contrary to the above, when the number of components was increased and the initial parameters to start the EM algorithm was determined by inspecting the histograms and for the rest (especially the hidden components) by letting the mean equals to 0 and standard deviation equals to 1, the final parameters produced (with minor adjustments) were similar to the real ones (a "characteristic" where some might consider it as unimportant and therefore choose to ignore).

The determination of the initial parameters to start the EM algorithm could be made easier and faster using the graphical techniques such as plotting $\log \frac{\phi_{i+1}}{\phi_i}$ against x_i where each approximately straight line, with negative slope represents an area where one component dominates and the kernel

method defined by $f(\hat{t}_k) = \sum_{l=-\frac{m}{2}}^{\frac{m}{2}} \exp\left(-\frac{2\pi ikl}{m}\right)\exp\left(-\frac{1}{2}h^2\left(\frac{2\pi l}{b-a}\right)^2\right)\left(\frac{1}{m}\sum_{k=0}^{m-1}\xi_k \exp\left(\frac{2\pi ikl}{m}\right)\right)$, $m = 2^r$ (Everitt *et al*., 1981; Bhattacharya, 1967; Silverman, 1986). Nevertheless, the main disadvantage of both methods is that they can not be used to detect hidden components.

Appendix A

A.1 "Simulate" program uses the following formulas to produce "Sample4" (where the subscript represents the dimension of the matrix).

$$X_{nx1} = C_{nxn} Z_{nx1} + \mu_{nx1}, \Sigma_{nxn} = C_{nxn} C_{nxn}{}^{t}$$

where $c_{ij} = \begin{cases} \dfrac{\sigma_{ij} - \sum_{k=1}^{j-1} c_{ik}c_{jk}}{\left(\sigma_{jj} - \sum_{k=1}^{j-1} c_{jk}c_{jk}\right)^{\frac{1}{2}}} & ,j \le i \\ 0 & ,i < j \end{cases}$ and z_i, ith component of Z, is as per defined in equation (10), where μ and σ are set/fixed at 0 and 1, respectively.

A.2 Derivation of Equations (7), (8) and (9)

A.2.1 Using Lagrange multipliers defined by *max/min F(x,y,z)* subject to $\Phi(x,y,z)=0$, $G(x,y,z)=F(x,y,z)+\lambda\Phi(x,y,z)$, $\frac{\partial G}{\partial x} = 0, \frac{\partial G}{\partial y} = 0, \frac{\partial G}{\partial Z} = 0$ (Spiegel, 1974) on $max \sum_i \sum_j p_{ij} \log(a_j)$ subject to $\sum_j a_j = 1$ (or $\left(\sum_j a_j - 1\right) = 0$, Equation (7) would be obtained.

A.2.2 From $\frac{\partial}{\partial \mu_j}\left(\frac{1}{2}\sum_i \sum_j p_{ij}(\mathbf{X}_i - \boldsymbol{\mu}_j)^t \Sigma_j^{t-1}(\mathbf{X}_i - \boldsymbol{\mu}_j)\right) = 0$, equation (8) would be obtained using the following matrix properties, $\frac{\partial \mathbf{x}^t \mathbf{A}\mathbf{y}}{\partial \mathbf{x}} = \mathbf{A}\mathbf{y}, \frac{\partial \mathbf{a}^t \mathbf{x}}{\partial \mathbf{x}} = \mathbf{a}$

A.2.3 The first and second expressions of

$\frac{\partial}{\partial \boldsymbol{\Sigma}_j^{-1}}\left(\frac{1}{2}\sum_i \sum_j p_{ij}(\mathbf{X}_i - \boldsymbol{\mu}_j)^t \boldsymbol{\Sigma}_j^{-1}(\mathbf{X}_i - \boldsymbol{\mu}_j)\right) + \frac{\partial}{\partial \boldsymbol{\Sigma}_j^{-1}}\left(\frac{1}{2}\sum_i \sum_j p_{ij} \log\left|\boldsymbol{\Sigma}_j^{-1}\right|\right) = 0$ use the following matrix properties, $\frac{\partial tr(\mathbf{x}\mathbf{y})}{\partial \mathbf{x}} = \mathbf{y} + \mathbf{y}^t - Diag(\mathbf{y})$, and $\sum \mathbf{X}_i^t \mathbf{A}\mathbf{x}_i = tr\left(\mathbf{A}\sum \mathbf{x}_i \mathbf{x}_i^t\right)$ to get equation (9). (Mardia *et al*., 1979).

REFERENCES

Bhattacharya, C.G. (1967). A simple method of resolution of a distribution into Gaussian components. *Biometrics*, *23,* 115-35.

Bilmes, J.A. (1997). A Gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian Mixture and Hidden Markov Models. Technical Report, University of Berkeley, ICSI-TR-97-021.

Box, G.E.P. and Muller, M.E. (1958). A note on the generating of random normal deviates. *Annals of Mathematical Statistics*, *29,* 610-611.

Dempster, A.P., Laird, N.M. and Rubin, D.B. (1977). Maximum Likelihood from incomplete data via the EM algorithm. *Journal of Royal Statistics Society*, *39*(1), 1-21.

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

Mardia, K.V., Kent, J.T. and Bibby, J.M. (1979). *Multivariate Analysis*. London: Academic Press Inc. Ltd.

Mohd Yusoff, M.I., Abu Bakar, M.R. and Mohd Nor, A.H.S. (2007). Fraud detection in telecommunication using Data Mining application. In *Proceedings of 9th Islamic Countries Conference on Statistical Sciences 2007 (ICCS-IX)*.

Reynolds, D.A. and Rose, R.C. (1995). Robust text-independent speaker identification using Gaussian Mixture Speaker Models. *IEEE Transactions on Speech and Audio Processing, 3*(1), January 1995.

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

Spiegel, M.R. (1974). *Shaum's Outline Series: Theory and Problems of Advanced Calculus*: SI (Metric) edition. Mc Graw-Hills, Inc.

Tanigushi, M., Haft, M., Hollmen, J. and Tresp, V. (1998). Fraud detection in communications networks using neural and probabilistic methods. In *Proceeding of the 1998 IEEE International Conference in Acoustics, Speech and Signal Processing (ICASSP'98)*, *II,* 1241-44.

Tsay, R.S. (2005). *Analysis of Financial Time Series: Financial Econometrics*. John Wiley and Sons.