Acceleration of the EM algorithm

Shiro Ikeda
PRESTO, Japan Science and Technology Corporation (JST)

July 25, 2001

Abstract

The EM algorithm is used for many applications including Boltzmann machine, stochastic Perceptron and HMM. This algorithm gives an iterating procedure for calculating the MLE of stochastic models which have hidden random variables. It is simple, but the convergence is slow. We also have "Fisher's scoring method". Its convergence is faster, but the calculation is heavy. We show that by using the EM algorithm recursively, we can connect these two methods and accelerate the EM algorithm. Also Louis, Meng and Rubin showed they can accelerate the EM algorithm, but our algorithm is simpler. We show some numerical simulations with our algorithm.

Keywords EM algorithm, Fisher's scoring method, maximum likelihood estimate, Louis turbo

1 Introduction

The EM (Expectation Maximization) algorithm[9] was originally proposed by Dempster et al.[4] for estimating the MLE(Maximum Likelihood Estimate) of stochastic models which have hidden random variables. The algorithm is now used in many applications such as HMM (Hidden Markov Model)[10] and some neural networks including Boltzmann Machine[2], stochastic Perceptron, and mixture of expert networks[5][6][7].

This algorithm gives an iterative procedure for each model to obtain the MLE. The practical form of each step is usually simple but the convergence speed is slow. There are some works to accelerate the convergence speed of the EM algorithm [8], but the procedure is usually not easy and need a lot of calculations. Meng and Rubin[11] proposed a practical method for realizing the acceleration algorithm but it is still difficult to be carried out.

On the other hand, there is an algorithm which is called the Fisher's scoring method[9]. This algorithm is also used to estimate the MLE with some iterative method. It is know that the Fisher's scoring converges faster than the EM algorithm but the calculation of each step is heavy. For the models like HMM or Neural Networks it is difficult to apply this method.

In this paper, we first show the relation between the EM algorithm and the Fisher's scoring. Based on the result, we propose the way to approximate the Fisher's scoring by applying the EM algorithm recursively. This procedure is simple and gives a fast convergence speed. The algorithm consists of two parts. First, it applies one EM step with the given data set. After that, draw data from the updated model and apply another EM step with this new data set. We show that we can obtain better parameters through the process. We show some results of numerical simulations and the algorithm converges faster than the original EM algorithm.

2 The EM algorithm and the Fisher's scoring

2.1 The EM algorithm

When we estimate the parameters of Boltzmann machine[2] or stochastic Perceptron[1], what makes the estimation difficult is the hidden random variables in the models. Let us define their random variables as $x = (y, z)$, where y is the visible variable (output cells), and z is the hidden variable (output of the hidden layers). With these variables, we can define the model as $p(x; \boldsymbol{\theta}) = p(y, z; \boldsymbol{\theta})$. For the estimation of their parameters, what is available is only an empirical distribution of y, $\{y_1, \cdots, y_N\}$. Let us define this empirical distribution of y as $\hat{q}(y) = \sum_{s=1}^{N} \delta(y_s)/N$. We want to estimate $\boldsymbol{\theta}$ from $\hat{q}(y)$.

The marginal distribution $p(y; \boldsymbol{\theta})$ is defined from $p(x; \boldsymbol{\theta})$ as

$$p(y; \boldsymbol{\theta}) = E_{p(z;\boldsymbol{\theta})}\left[p(x; \boldsymbol{\theta})\right] = \int p(x; \boldsymbol{\theta}) d\mu(z).$$

Let $l(y; \boldsymbol{\theta}) = \log p(y; \boldsymbol{\theta})$, and the log-likelihood function is

$$\begin{aligned} L(Y^N; \boldsymbol{\theta}) &\stackrel{\text{def}}{=} \frac{1}{N} \sum_{s=1}^{N} l(y_s; \boldsymbol{\theta}) \\ &= \int \hat{q}(y) l(y; \boldsymbol{\theta}) d\mu(y) = E_{\hat{q}(y)}\left[l(y; \boldsymbol{\theta})\right]. \end{aligned}$$

The definition of the maximum likelihood estimate is the parameter $\hat{\boldsymbol{\theta}}$ which maximizes this function $L(Y^N; \boldsymbol{\theta})$.

$$\hat{\boldsymbol{\theta}} = \operatorname*{argmax}_{\theta} L(Y^N; \boldsymbol{\theta}). \tag{1}$$

If the model has a hidden variable z, it is difficult to calculate MLE by solving (1) directly. The EM algorithm is useful in such cases.

In this paper, $p(x;\boldsymbol{\theta})$ is an exponential family where the probability density function is written as

$$p(x;\boldsymbol{\theta}) = \exp\left(\sum_{i=1}^{n}\theta^i r_i(x) - k(\boldsymbol{r}(x)) - \psi(\boldsymbol{\theta})\right). \tag{2}$$

where, $\boldsymbol{r}(x) = (r_1(x),\cdots,r_n(x))^T$, $\boldsymbol{\theta} = (\theta^1,\cdots,\theta^n)^T$ (natural parameter), and $\psi(\boldsymbol{\theta})$ is the normalization term which is a function of the natural parameter. Various models are included in the exponential family, such as Boltzmann machine, stochastic Perceptron, HMM, and Gaussian mixture[1][2]. Even when $p(x;\boldsymbol{\theta})$ belongs to the exponential family, the marginal distribution $p(y;\boldsymbol{\theta})$ is not always included in the exponential family.

The EM algorithm is an iterative algorithm to calculate the MLE by generating, from an initial point $\boldsymbol{\theta}_0$, a sequence $\{\boldsymbol{\theta}_t\}$ of estimates, $t = 1,2,3,\cdots$. Each step consists of the following two sub-steps.

- E-step: Given observation $\hat{q}(y)$ and current estimate, evaluate $Q(\boldsymbol{\theta},\boldsymbol{\theta}_t)$, which is defined as,

$$\begin{aligned} Q(\boldsymbol{\theta},\boldsymbol{\theta}_t) &= E_{\hat{q}(y)p(z|y;\boldsymbol{\theta}_t)}\left[l(y,z;\boldsymbol{\theta})\right] \\ &= \sum_{s=1}^{N}\int p(z|y_s;\boldsymbol{\theta}_t)l(y_s,z;\boldsymbol{\theta})d\mu(z). \end{aligned}$$

- M-step: Find the $\boldsymbol{\theta}_{t+1}$ that maximizes $Q(\boldsymbol{\theta},\boldsymbol{\theta}_t)$,

$$\boldsymbol{\theta}_{t+1} = \operatorname*{argmax}_{\boldsymbol{\theta}} Q(\boldsymbol{\theta},\boldsymbol{\theta}_t)$$

After a cycle of E-step and M-step, we obtain $\boldsymbol{\theta}_{t+1}$ from $\boldsymbol{\theta}_t$ and it is shown[4] that

$$L(Y^N;\boldsymbol{\theta}_{t+1}) \geq L(Y^N;\boldsymbol{\theta}_t).$$

By iterating EM steps, the algorithm converges to the parameters which should be the MLE. The difference of the parameters between before and after one EM step can be approximated as (3) (The proof is shown in [9](3.76),[12]). Here, we have to note that this approximation is not true for the curved exponential family(Appendix A).

$$\boldsymbol{\theta}_{t+1} \simeq \boldsymbol{\theta}_t + {G_X}^{-1}(\boldsymbol{\theta}_t)\partial L(Y^N;\boldsymbol{\theta}_t), \tag{3}$$

where $\partial = (\partial_1,\cdots,\partial_n)^T = (\partial/\partial\theta^1,\cdots,\partial/\partial\theta^n)^T$ and $G_X(\boldsymbol{\theta}) = (g_{X\,ij}(\boldsymbol{\theta}))$ is the Fisher information matrix of $p(x;\boldsymbol{\theta})$. The definition is

$$\begin{aligned} g_{X\,ij}(\boldsymbol{\theta}) &= E_{p(x;\boldsymbol{\theta})}\left[\partial_i l(x;\boldsymbol{\theta})\partial_j l(x;\boldsymbol{\theta})\right] \\ &= -E_{p(x;\boldsymbol{\theta})}\left[\partial_i\partial_j l(x;\boldsymbol{\theta})\right]. \end{aligned}$$

This result shows that the EM algorithm is updating the parameter along the steepest decent direction with the metric defined by G_X.

2.2 The Fisher's scoring

In this section, we derive the relation between the Fisher's scoring and the EM algorithm. The Fisher's scoring is also an iterative algorithm to have the MLE. The updating rule is,

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + {G_Y}^{-1}(\boldsymbol{\theta}_t)\partial L(Y^N;\boldsymbol{\theta}_t). \tag{4}$$

It looks similar to the EM algorithm but it is known that the convergence is faster than the EM algorithm. This comes from the difference between $G_X(\boldsymbol{\theta})^{-1}$ in (3) and $G_Y(\boldsymbol{\theta})^{-1}$ in (4). $G_Y(\boldsymbol{\theta}) = (g_{Y\,ij}(\boldsymbol{\theta}))$ is the Fisher information matrix of $p(y;\boldsymbol{\theta})$.

$$\begin{aligned} g_{Y\,ij}(\boldsymbol{\theta}) &= E_{p(y;\boldsymbol{\theta})}\left[\partial_i l(y;\boldsymbol{\theta})\partial_j l(y;\boldsymbol{\theta})\right] \\ &= -E_{p(y;\boldsymbol{\theta})}\left[\partial_i\partial_j l(y;\boldsymbol{\theta})\right]. \end{aligned}$$

There is a following relation between $G_X(\boldsymbol{\theta})$ and $G_Y(\boldsymbol{\theta})$.

$$\begin{aligned} -l(y;\boldsymbol{\theta}) &= -l(x;\boldsymbol{\theta}) + l(z|y;\boldsymbol{\theta}) \\ -E_{p(y;\boldsymbol{\theta})}\left[\partial_i\partial_j l(y;\boldsymbol{\theta})\right] &= -E_{p(x;\boldsymbol{\theta})}\left[\partial_i\partial_j l(x;\boldsymbol{\theta})\right] \\ &\quad +E_{p(x;\boldsymbol{\theta})}\left[\partial_i\partial_j l(z|y;\boldsymbol{\theta})\right] \\ G_Y(\boldsymbol{\theta}) &= G_X(\boldsymbol{\theta}) - G_{Z|Y}(\boldsymbol{\theta}) \end{aligned} \tag{5}$$

$G_{Z|Y} = (g_{Z|Y\,ij}(\boldsymbol{\theta}))$ is also a conditional Fisher information matrix defined as

$$\begin{aligned} g_{Z|Y\,ij}(\boldsymbol{\theta}) &= -E_{p(y;\boldsymbol{\theta})}\left[E_{p(z|y;\boldsymbol{\theta})}\left[\partial_i\partial_j l(z|y;\boldsymbol{\theta})\right]\right] \\ &= E_{p(y;\boldsymbol{\theta})}\left[g_{Z|y\,ij}(\boldsymbol{\theta})\right]. \end{aligned}$$

Generally, G_Y, G_X and $G_{Z|Y}$ are positive definite symmetric matrices.

One step of the Fisher's scoring changes the parameters into the Fisher efficient estimator. However, calculation of ${G_Y}^{-1}$ is intractable in many models. Here, we show the following expansion which is the key of our algorithm.

Theorem 1. *${G_Y}^{-1}$ can be expanded with ${G_X}^{-1}$ and $G_{Z|Y}$ as*

$${G_Y}^{-1} = \left(I + \sum_{i=1}^{\infty}({G_X}^{-1}G_{Z|Y})^i\right){G_X}^{-1}. \tag{6}$$

Proof. (6) is easily obtained be simultaneous diagonalization of G_Y, G_X and $G_{Z|Y}$[9]. □

Using this result, (4) can be rewritten as,

$$\begin{aligned} \boldsymbol{\theta}_{t+1} &= \boldsymbol{\theta}_t + {G_Y}^{-1}\partial L(Y^N;\boldsymbol{\theta}_t) \\ &= \boldsymbol{\theta}_t + {G_X}^{-1}\partial L(Y^N;\boldsymbol{\theta}_t) \\ &\quad +{G_X}^{-1}G_{Z|Y}{G_X}^{-1}\partial L(Y^N;\boldsymbol{\theta}_t) \\ &\quad +({G_X}^{-1}G_{Z|Y})^2{G_X}^{-1}\partial L(Y^N;\boldsymbol{\theta}_t) \\ &\quad +\cdots. \end{aligned} \tag{7}$$

(7) shows that the EM algorithm is the first order approximation of the Fisher's scoring.

3 Proposed algorithm

We have shown that the Fisher's scoring update the parameters into the Fisher efficient direction. However, ${G_Y}^{-1}$ is intractable especially for the model such as neural networks or HMM. Here, we propose an algorithm to approximate the Fisher's scoring by using the tractable EM algorithm recursively.

Suppose the case we have applied one EM step to $\boldsymbol{\theta}_t$ and have obtained an estimate $\boldsymbol{\theta}_{t+1}$. This new estimate $\boldsymbol{\theta}_{t+1}$ gives a probability distribution $p(y;\boldsymbol{\theta}_{t+1})$. Let us draw N' samples from $p(y;\boldsymbol{\theta}_{t+1})$ as $\{\bar{y}_1, \bar{y}_2, \cdots, \bar{y}_{N'}\} \sim p(y;\boldsymbol{\theta}_{t+1})$ and use this data set for estimation. After one EM step applied to $\boldsymbol{\theta}_t$, we have a new estimate $\bar{\boldsymbol{\theta}}_{t+1}$. This parameter is different from $\boldsymbol{\theta}_t$ nor $\boldsymbol{\theta}_{t+1}$. It can be shown that we can make a better estimate with $\boldsymbol{\theta}_t$, $\boldsymbol{\theta}_{t+1}$ and $\bar{\boldsymbol{\theta}}_{t+1}$ (Fig.1). First, we show the following theorem which describes the feature of $\bar{\boldsymbol{\theta}}_{t+1}$.

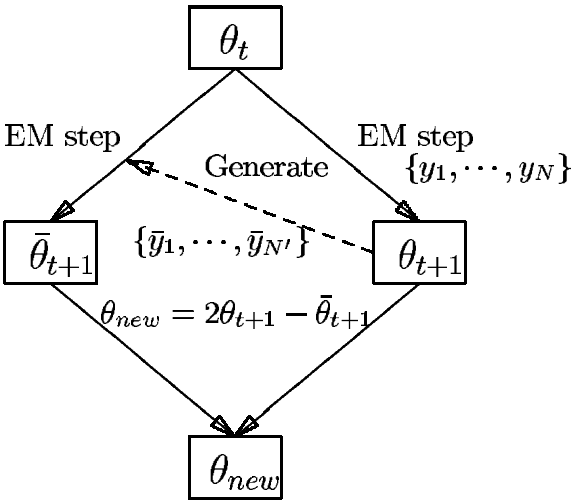


Figure 1: Flowchart of the proposed algorithm

Theorem 2. *Let $\bar{\boldsymbol{\theta}}_{t+1}$ be the parameter estimated from $\boldsymbol{\theta}_t$ taking $p(y;\boldsymbol{\theta}_{t+1})$ as the target distribution and applied one EM step. Then we have the following approximation,*

$$\bar{\boldsymbol{\theta}}_{t+1} - \boldsymbol{\theta}_t \simeq {G_X}^{-1} G_Y {G_X}^{-1} \partial L(Y^N;\boldsymbol{\theta}_t). \quad (8)$$

Proof. See Appendix B. □

From (3), (5) and (8), we can derive

$$\begin{aligned} & \bar{\boldsymbol{\theta}}_{t+1} - \boldsymbol{\theta}_t \\ \simeq\ & {G_X}^{-1}(G_X - G_{Z|Y}){G_X}^{-1}\partial L(Y^N;\boldsymbol{\theta}_t) \\ \simeq\ & (\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t) - {G_X}^{-1} G_{Z|Y} {G_X}^{-1} \partial L(Y^N;\boldsymbol{\theta}_t). \end{aligned} \quad (9)$$

The second order approximation of the Fisher's scoring in (7) is

$$\begin{aligned} & {G_X}^{-1} G_{Z|Y} {G_X}^{-1} \partial L(Y^N;\boldsymbol{\theta}_t) \\ \simeq\ & (\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t) - (\bar{\boldsymbol{\theta}}_{t+1} - \boldsymbol{\theta}_t) = \boldsymbol{\theta}_{t+1} - \bar{\boldsymbol{\theta}}_{t+1}. \end{aligned}$$

Therefore, the approximation of the Fisher's scoring up to the second order is

$$\begin{aligned} \boldsymbol{\theta}' &= 2\boldsymbol{\theta}_{t+1} - \bar{\boldsymbol{\theta}}_{t+1} \\ &= \boldsymbol{\theta}_t + (\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t) + (\boldsymbol{\theta}_{t+1} - \bar{\boldsymbol{\theta}}_{t+1}) \\ &\simeq \boldsymbol{\theta}_t + {G_X}^{-1}(I + G_{Z|Y}{G_X}^{-1})\partial L(Y^N;\boldsymbol{\theta}_t). \end{aligned}$$

We can use the similar process to approximate higher orders of the Fisher's scoring.

Collorary 1. *Apply one EM step to $\boldsymbol{\theta}_t$ and estimate $\bar{\boldsymbol{\theta}}_{t+i}$ where $p(y;\bar{\boldsymbol{\theta}}_{t+i-1})$ is the target distribution ($\bar{\boldsymbol{\theta}}_t = \boldsymbol{\theta}_{t+1}$, $i = 1, 2, \cdots$), $\bar{\boldsymbol{\theta}}_{t+i}$ has the following property.*

$$\begin{aligned} \bar{\boldsymbol{\theta}}_{t+i} - \boldsymbol{\theta}_t &\simeq ({G_X}^{-1}G_Y)^i {G_X}^{-1}\partial L(Y^N;\boldsymbol{\theta}_t) \\ &= (I - {G_X}^{-1}G_{Z|Y})^i {G_X}^{-1}\partial L(Y^N;\boldsymbol{\theta}_t) \end{aligned}$$

Proof. The proof is similar to Theorem 2 (Appendix B) □

This result shows that we can approximate $({G_X}^{-1}G_{Z|Y})^i {G_X}^{-1}\partial L(Y^N;\boldsymbol{\theta}_t)$ and the Fisher's scoring up to ith order by $\bar{\boldsymbol{\theta}}_t, \cdots, \bar{\boldsymbol{\theta}}_{t+i}$ and $\boldsymbol{\theta}_t$. But since we need to use some Monte Carlo method when the target distribution is a continuous distribution, approximation of the order higher than 2 will not be effective.

For a discrete distribution, we can use the density function itself for the EM algorithm and we only have to calculate up to $i = n$ for the higher order approximations since we can calculate the higher orders by linear combinations. Let us define $\boldsymbol{g}_i$ as follows

$$\begin{aligned} \boldsymbol{g}_0 &= {G_X}^{-1}\partial L(Y^N;\boldsymbol{\theta}) \\ &\vdots \\ \boldsymbol{g}_n &= ({G_X}^{-1}G_{Z|Y})^n {G_X}^{-1}\partial L(Y^N;\boldsymbol{\theta}). \end{aligned}$$

Because $\boldsymbol{\theta}$ is an n dimensional vector, $\boldsymbol{g}_1, \cdots, \boldsymbol{g}_n$ are linearly dependent. Therefore, we have the following relation

$$\boldsymbol{g}_n = a_1\boldsymbol{g}_1 + \cdots + a_{n-1}\boldsymbol{g}_{n-1}.$$

$\boldsymbol{g}_{n+1}$ is written with $a_1, \cdots, a_n$ as

$$\begin{aligned} \boldsymbol{g}_{n+1} &= ({G_X}^{-1}G_{Z|Y})^{n+1} {G_X}^{-1}\partial L(Y^N;\boldsymbol{\theta}) \\ &= a_1\boldsymbol{g}_2 + \cdots + a_{n-1}\boldsymbol{g}_n. \end{aligned}$$

And it is the same for any higher orders.

We proposed a new algorithm which uses the EM algorithm recursively. First, we apply one EM step using the original given data set. After that we apply another EM step using the data generated by the model. And finally, we make a better estimate.

4 Numerical Simulations

4.1 Log-linear model

First, we show a result of the proposed algorithm applied to a log-linear Model.

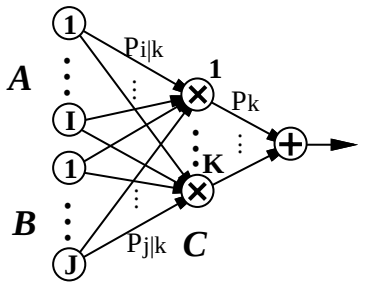


Figure 2: Definition of the model

The model (Fig.2) has three random variables (A, B, C), where A, B and C take values on $\{A_i\}$ $(i = 1, \cdots, I)$, $\{B_j\}$ $(j = 1, \cdots, J)$ and $\{C_k\}$ $(k = 1, \cdots, K)$ respectively. Therefore the density function is discrete. We can observe two variables A, B of them, but cannot observe C (latent variable). We make an assumption that the probability distribution has the form,

$$P(A, B, C) = P(A|C)P(B|C)P(C). \qquad (10)$$

The distributions of A and B is independent conditional to C.

When we observe data, we cannot know the true frequency distribution of A, B and C, but the marginal distribution of A and B. Empirical distribution of A and B is written as, $m_{ij} = n_{ij}/\sum_{i'j'} n_{i'j'}$, where n_{ij} is the frequency of observing $(A = A_i, B = B_j)$. From the assumption, we can write this distribution as $P(A_i, B_j) = \sum_k P_{i|k} P_{j|k} P_k$. We want to estimate $P_{i|k}$ and $P_{j|k}$ and P_k from m_{ij}. Since we have a latent variable P_k, we can apply the EM algorithm. We made a numerical simulation with a model which is $I = J = 5$, and $K = 2$. Therefore, $p(A_i, B_j)$ is multinomial distribution of 25 elements. If we have 24 parameters, we can describe the given distribution precisely, but now we only have $(K - 1) + K(I - 1) + K(J - 1) = 17$ parameters. The target distribution was made at random, and the problem is to estimate the parameter to fit the target distribution.

Fig.3 is the result using the original EM algorithm and the proposed algorithm which approximate the scoring up to the 2nd and the 3rd order. You can see that if we use the 2nd or 3rd order approximation, the convergence speed is much faster than the original EM algorithm.

4.2 Gaussian Mixture

The log-linear model was a discrete distribution, and we did not have to draw data from the distribution. But when the density function is continuous, we need a sampled data set $\{\bar{y}_1, \cdots, \bar{y}_{N'}\}$ drawn from $p(y; \boldsymbol{\theta}_{t+1})$ in order to have $\bar{\boldsymbol{\theta}}_{t+1}$. To test if this Monte Carlo procedure works, we did a simulation using the mixture of 2-dimensional Gaussian distributions[13].

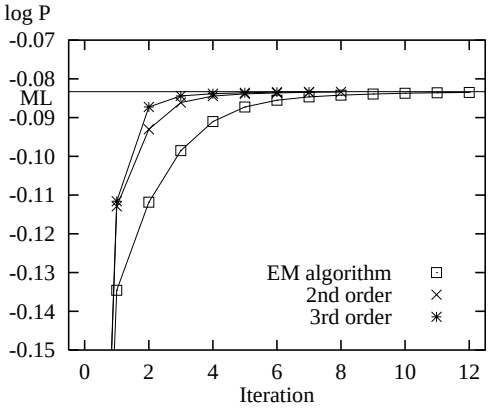
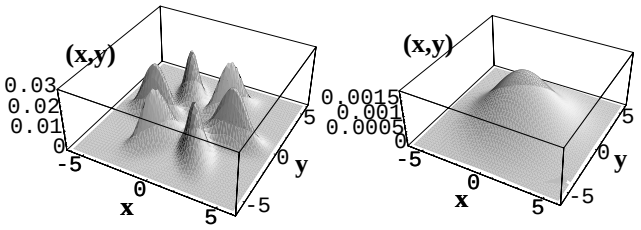


Figure 3: The increase of the log-likelihood



True model Initial model for learning

Figure 4: The true model and the initial model for learning

Fig.4 shows the density functions of the true distribution and the initial model for learning. Both of them consists of 6 Gaussian distributions. In the initial model, since the covariances are large, each of 6 cannot be observed clearly.

We don't show the exact form of the EM algorithm, but it is simple and calculation is not heavy. We applied the proposed algorithm on this model as follows.

1. Prepare 1000 samples from the true distribution. Let the parameter of the initial model be $\boldsymbol{\theta}_0$.
2. Using the data, apply one EM step to $\boldsymbol{\theta}_t$ and calculate $\boldsymbol{\theta}_{t+1}$.
3. Generate 1000 new data according to $p(y; \boldsymbol{\theta}_{t+1})$.
4. Using the newly generated data, apply one EM step to $\boldsymbol{\theta}_t$ and calculate $\bar{\boldsymbol{\theta}}_{t+1}$.
5. Let $\boldsymbol{\theta}_{new} = 2\boldsymbol{\theta}_{t+1} - \bar{\boldsymbol{\theta}}_{t+1}$ and $\boldsymbol{\theta}_t = \boldsymbol{\theta}_{new}$, then go to 2.

Fig.6 shows the estimated models by the EM algorithm and the proposed algorithm. Also the profile of the log-likelihood during the iterations is shown in Fig.5. Since the proposed algorithm includes a sort of Monte Carlo method, it does not converge but keep fluctuating. This is the reason why we did not test any of higher order approximations. The result shows that the proposed algorithm has a better performance.

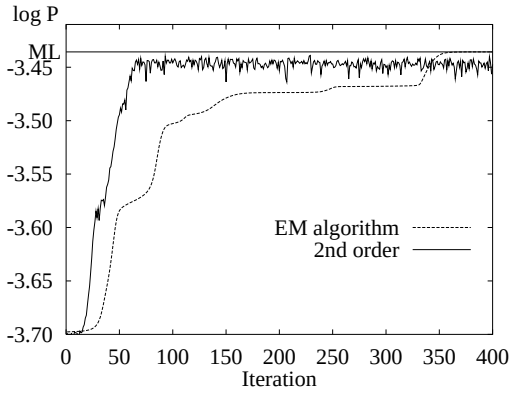


Figure 5: The transition of the log-likelihood according to the iteration

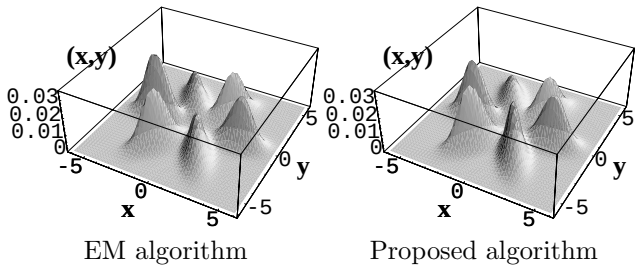


Figure 6: Results of learning

We have to reconsider this result taking the amount of the calculations into account. One step of the proposed algorithm includes two EM steps. When we compare the speed of the convergence, it is better to include this factor. We show the result in Fig.7. This result shows that the proposed algorithm still converges faster than the EM algorithm.

Finally, we add one modification to the proposed algorithm to suppress the fluctuation. The idea is to switch the proposed algorithm to the ordinary EM algorithm. We determine the switching point by a function $\lambda(t)$ which is defined as follows.

$$\begin{aligned} \lambda(t) &= \eta\lambda(t-1) + (1-\eta)L(Y^N;\boldsymbol{\theta}_t), t = 1, \cdots, \\ \lambda(0) &= L(Y^N;\boldsymbol{\theta}_0). \end{aligned} \tag{11}$$

When $\lambda(t)$ decreases, we switch to the original EM algorithm automatically. We defined η as 0.7, and the result is shown in Fig. 8. This result shows that our algorithm converges about 3 times faster than the original EM algorithm.

5 Discussion

It is shown that the proposed algorithm improves the performance of the EM algorithm through numerical simulations. But there is a practical problem. In order to have the second order approximation, we have to use two EM steps. Therefore we hope that the proposed

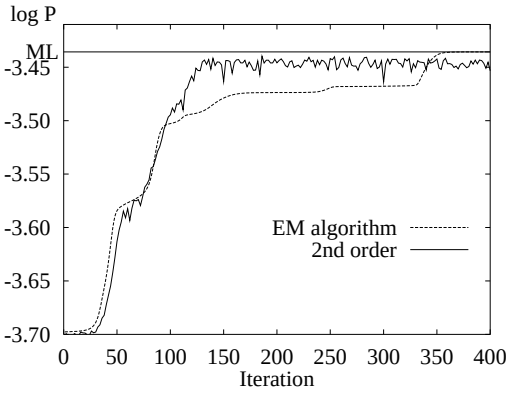


Figure 7: The transition of the log-likelihood according to the iteration considering the amount of the calculation

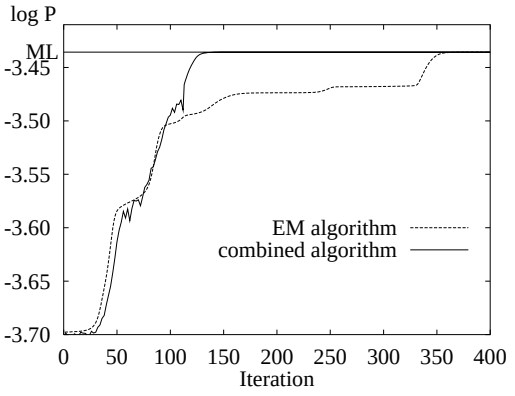


Figure 8: Combining the proposed algorithm and the EM algorithm

algorithm works twice faster than the original EM algorithm. However, it is not always true. For Gaussian mixture, it converges more than two times faster, but for the log-linear model, it is not the case.

Let $\boldsymbol{\theta}_{new}$ be the parameter which is obtained by the proposed algorithm of the second order approximation, and $\boldsymbol{\theta}_{t+2}$ be the parameter obtained after two EM steps. Generally, $\boldsymbol{\theta}_{new} \neq \boldsymbol{\theta}_{t+2}$. We want to compare $L(Y^N;\boldsymbol{\theta}_{new})$ with $L(Y^N;\boldsymbol{\theta}_{t+2})$. When $\boldsymbol{\theta}_t$, $\bar{\boldsymbol{\theta}}_t$, $\boldsymbol{\theta}_{t+1}$ and $\boldsymbol{\theta}_{t+2}$ are close to each others, the result of section 3 gives the following approximation of $L(Y^N;\boldsymbol{\theta}_{new})$

$$\begin{aligned} & L(Y^N;\boldsymbol{\theta}_{new}) = L(Y^N; 2\boldsymbol{\theta}_{t+1} - \bar{\boldsymbol{\theta}}_{t+1}) \\ \simeq\ & L(Y^N;\boldsymbol{\theta}_t) + \partial {L_t}^T G_X(\boldsymbol{\theta}_t)^{-1} \partial L_t \\ & + \partial {L_t}^T G_X(\boldsymbol{\theta}_t)^{-1} G_{Z|Y}(\boldsymbol{\theta}_t) G_X(\boldsymbol{\theta}_t)^{-1} \partial L_t. \end{aligned} \tag{12}$$

Here, $L_t = L(Y^N;\boldsymbol{\theta}_t)$. On the other hand,

$$\begin{aligned} & L(Y^N;\boldsymbol{\theta}_{t+2}) = L(Y^N;\boldsymbol{\theta}_{t+2}-\boldsymbol{\theta}_{t+1}+\boldsymbol{\theta}_{t+1}) \\ \simeq\ & L(Y^N;\boldsymbol{\theta}_t) + \partial {L_t}^T G_X(\boldsymbol{\theta}_t)^{-1}\partial L_t \\ & +\partial {L_t}^T G_X(\boldsymbol{\theta}_{t+1})^{-1}\partial L_t \\ & -\partial {L_t}^T G_X(\boldsymbol{\theta}_{t+1})^{-1}\mathcal{G}_Y(\boldsymbol{\theta}_t)G_X(\boldsymbol{\theta}_t)^{-1}\partial L_t \\ =\ & L(Y^N;\boldsymbol{\theta}_t) + \partial {L_t}^T G_X(\boldsymbol{\theta}_t)^{-1}\partial L_t \\ & +\partial {L_t}^T G_X(\boldsymbol{\theta}_{t+1})^{-1}\mathcal{G}_{Z|Y}(\boldsymbol{\theta}_t)^{-1}G_X(\boldsymbol{\theta}_t)^{-1}\partial L_t. \end{aligned} \tag{13}$$

In the equations, we used the following definitions: $\mathcal{G}_Y(\boldsymbol{\theta}_t) = -\sum_{s=1}^N \partial^2 l(y_s;\boldsymbol{\theta}_t)$, $\mathcal{G}_{Z|Y}(\boldsymbol{\theta}_t) = -\sum_{s=1}^N E_{p(z|y_s;\boldsymbol{\theta}_t)}\left[\partial^2 l(z|y_s;\boldsymbol{\theta}_t)\right]$. It is not clear which of (12) and (13) is larger in general cases. Qualitatively speaking, if the model is close to the convergence point, $G_X(\boldsymbol{\theta}_{t+1}) \simeq G_X(\boldsymbol{\theta}_t)$ and $\mathcal{G}_{Z|Y}(\boldsymbol{\theta}_t) \simeq G_{Z|Y}(\boldsymbol{\theta}_t)$, and two EM steps are almost equivalent to the proposed algorithm. And also if $\mathcal{G}_{Z|Y}(\boldsymbol{\theta}_t)$ and $G_{Z|Y}(\boldsymbol{\theta}_t)$ are close to $\boldsymbol{O}$, both of them are almost equivalent. This is the case where almost all information of z is observed through y. For example, if the mean of each Gaussian distribution is far from each other compared to the variance of each distribution, it is easy to have the information of the latent variable z. In these cases, the EM algorithm and the Fisher's scoring are almost equivalent, and the proposed algorithm gives almost the same result as the EM algorithm.

Let us consider the case where $L(Y^N;\boldsymbol{\theta}_{new})$ is larger than $L(Y^N;\boldsymbol{\theta}_{t+2})$. When $G_X(\boldsymbol{\theta}_t)$ and $G_X(\boldsymbol{\theta}_{t+1})$ are almost the same, it depends on the matrices $\mathcal{G}_{Z|Y}(\boldsymbol{\theta}_t)$ and $G_{Z|Y}(\boldsymbol{\theta}_t)$. This is the case where the given data $\{y_s\}$ includes more information of z than $\{\bar{y}_s\} \sim p(y;\boldsymbol{\theta}_{t+1})$. This is not clear generally. This is also true for higher order approximations.

In the continuous distributions, we used a sort of Monte Carlo method and it gives fluctuation. We want to estimate the variance of the fluctuation using the result of Appendix B. When we generate N' samples by Monte Carlo methods, and N' is sufficiently large, asymptotically $\boldsymbol{\theta}_{new}$ follows a normal distribution. Let the mean of $\boldsymbol{\theta}_{new}$ be $\boldsymbol{\theta}^*_{new}$, and its variance is approximated as ${G_X}^{-1}G_Y(\boldsymbol{\theta}_{t+1}){G_X}^{-1}/N'$. Let $L_{new} = L(Y^N;\boldsymbol{\theta}_{new})$ and $L^*_{new} = L(Y^N;\boldsymbol{\theta}^*_{new})$, neglecting the higher orders, expand L_{new} around $\boldsymbol{\theta}^*_{new}$. If N' is sufficiently large, $\boldsymbol{\theta}_{new}-\boldsymbol{\theta}^*_{new}$ will be sufficiently small, and we can have the following approximation

$$L_{new} \simeq L^*_{new} + \partial {L^*_{new}}^T(\boldsymbol{\theta}_{new}-\boldsymbol{\theta}^*_{new}). \tag{14}$$

Expectation of L_{new} is almost L^*_{new}. Its variance is $\partial {L^*_{new}}^T({G_X}^{-1}G_Y(\boldsymbol{\theta}_{t+1}){G_X}^{-1})\partial L^*_{new}/N'$. This is inversely proportional to N'. We should take care of N' depending on the problem.

We proposed a method of switching using a parameter η. If the variance of $L(Y^N;\theta_t)$ is the same and t is large, finally the expectation of the variance of $\lambda(t)$ converges to $(1-\eta)^2/(1-\eta^2)$ times of the variance of L_{new}. Therefore the method switches at the point where $(1-\eta)^2/(1-\eta^2)$ times of the variance is roughly equivalent to (12). When $\eta = 0.7$, $(1-\eta)^2/(1-\eta^2) = 9/51$.

6 Conclusion

A lot of acceleration algorithms have been proposed for the EM algorithm. Most of them is based on the same expansion of the Fisher's scoring as the proposed algorithm. Usually they define the Jacobian J of the function $\boldsymbol{\theta}_{t+1} = EM(\boldsymbol{\theta}_t)$, and J and $(\boldsymbol{\theta}_{t+1}-\boldsymbol{\theta}_t)$ are used for approximating the Fisher's scoring. J corresponds to ${G_X}^{-1}G_{Z|Y}$ in our formulation. The Aitken acceleration is one of the methods to calculate J. This method approximate J directly from the function $\boldsymbol{\theta}_{t+1} = EM(\boldsymbol{\theta}_t)$[9], and the cost of calculation can be roughly the same as our proposed method but the eigen values of J does not always stay between 0 and 1.

Another popular acceleration algorithm is Louis turbo[8]. Louis turbo itself does not give any practical method to obtain J. Meng and Rubin proposed a method to obtain J by using the EM algorithm[11]. In their method, they need to apply the EM algorithm as many times as the number of the parameters. Once you obtain J, you can approximate the Fisher's scoring up to any order, but in order to have the second order approximation, you need to apply the EM algorithm more than twice. On the other hand, our method only needs two EM steps. For higher order approximations, if the order is less or equal to the number of the parameters, we need to apply the EM steps as much as the order. Therefore, the proposed algorithm needs equal or less calculation than Meng and Rubin.

We are planning to apply proposed algorithm to Neural Networks, HMM and on-line learning.

Acknowledgment

The author thanks Shun-ichi Amari and Noboru Murata in BSI, RIKEN for very useful discussions on this work.

References

[1] S. Amari. "Dualistic geometry of the manifold of higher-order neurons." *Neural Networks*, Vol. 4, No. 4, pp. 443–451, 1991.

[2] S. Amari, K. Kurata, and H. Nagaoka. "Information geometry of Boltzmann machines." *IEEE Trans. Neural Networks*, Vol. 3, No. 2, pp. 260–271, March, 1992.

[3] S. Amari. "Information Geometry of the EM and em Algorithms for Neural Networks." *Neural Networks*, Vol. 8, No. 9, pp. 1379–1408, 1995.

[4] A. P. Dempster, N. M. Laird, and D. B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm." *J. R. Statistical Society, Series B*, Vol. 39, pp. 1–38, 1977.

[5] R. A. Jacobs and M. I. Jordan. "Adaptive mixtures of local experts." *Neural Computation*, Vol. 3, No. 1, pp. 79–87, Spring 1991.

[6] M. I. Jordan and R. A. Jacobs. "Hierarchical mixtures of experts and the EM algorithm." *Neural Computation*, Vol. 6, No. 2, pp. 181–214, March 1994.

[7] M. I. Jordan and L. Xu. "Convergence results for the EM approach to mixture of experts architectures." *Neural Networks*, Vol. 8, No. 9, pp. 1409–1431, 1995.

[8] T. A. Louis. "Finding the observed information matrix when using the EM algorithm." *J. R. Statistical Society, Series B*, Vol. 44, No. 2, pp. 226–233, 1982.

[9] G. J. McLachlan and T. Krishnan. "*The EM Algorithm and Extensions*." Wiley series in probability and statistics. John Wiley & Sons, Inc., 1997.

[10] L. R. Rabiner, S. E. Levinson, and M. M. Sondhi. "On the application of vector quantization and hidden Markov models to speaker-independent, isolated word recognition." *The Bell System Technical Journal*, Vol. 62, No. 4, pp. 1075–1105, April 1983.

[11] M. A. Tanner. "*Tools for Statistical Inference – Observed Data and Data Augmentation Methods*," Vol. 67 of *Lecture Notes in Statistics*. Springer-Verlag, 1991.

[12] D. M. Titterington. "*Recursive Parameter Estimation using Incomplete Data*," *J. R. Statist. Soc. B*, Vol. 46, No. 2, pp. 257–267, 1984.

[13] L. Xu and M. I. Jordan. "On convergence properties of the EM algorithm for Gaussian mixture." A.I.Memo No.1520, C.B.C.L. Paper No.111, 1995.

A Curved Exponential Family

For curved exponential family, generally (3) does not hold. For the proof of (3), we used the fact that second derivative of $Q(\boldsymbol{\theta}, \boldsymbol{\theta}_t)$ can be written with the Fisher's information matrix as,

$$\begin{aligned}\partial\partial Q(\boldsymbol{\theta}, \boldsymbol{\theta}_t)\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_t} &= E_{\hat{q}(y)p(z|y;\boldsymbol{\theta}_t)}\left[\partial\partial l(y, z; \boldsymbol{\theta}_t)\right] \\ &= E_{\hat{q}(y)p(z|y;\boldsymbol{\theta}_t)}\left[-\partial\partial \psi(\boldsymbol{\theta}_t)\right] \\ &= -\partial\partial\psi(\boldsymbol{\theta}_t) \\ &= -G_X(\boldsymbol{\theta}_t). \end{aligned} \quad (15)$$

This is true for exponential family but not always for curved exponential family. Suppose an exponential family with an n-dimensional parameter $\boldsymbol{\theta}$, and $\boldsymbol{\theta}$ is a function of $\boldsymbol{u} = (u^1, \cdots, u^m)$, $(\boldsymbol{\theta} = \boldsymbol{\theta}(\boldsymbol{u}))$, and $m < n$. In this case, second derivative of $Q(\boldsymbol{u}, \boldsymbol{u}_t)$ respect to $\boldsymbol{u}$, is,

$$\begin{aligned}&\frac{\partial^2 Q(\boldsymbol{u}, \boldsymbol{u}_t)}{\partial u^k \partial u^l}\Big|_{\boldsymbol{u}=\boldsymbol{u}_t} \\ &= E_{\hat{q}(y)p(z|y;\boldsymbol{u}_t)}\left[\frac{\partial^2 l(y, z; \boldsymbol{u})}{\partial u^k \partial u^l}\Big|_{\boldsymbol{u}=\boldsymbol{u}_t}\right] \\ &= \sum_i \frac{\partial^2 \theta^i(\boldsymbol{u})}{\partial u^k \partial u^l} E_{\hat{q}(y)p(z|y;\boldsymbol{u}_t)}\left[r_i(x) - \partial_i\psi(\boldsymbol{\theta}(\boldsymbol{u_t}))\right] \\ &\qquad - \frac{\partial^2 \psi(\boldsymbol{\theta}(\boldsymbol{u}))}{\partial u^k \partial u^l}\Big|_{\boldsymbol{u}=\boldsymbol{u}_t}. \end{aligned} \quad (16)$$

The first term of (16) is generally not equal to 0, therefore this is not equal to the Fisher's Information matrix as (15). The exceptional case is when $\boldsymbol{\theta}$ is a linear function of $\boldsymbol{u}$. In this case, first term of (16) is 0 and the approximation (16) can work.

B Proof of theorem 2

From (3), we have the following approximation,

$$\begin{aligned}\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t &\simeq G_X{}^{-1}\partial L(Y^N; \boldsymbol{\theta}_t) \\ &= G_X{}^{-1}\partial\left(E_{\hat{q}(y)}\left[l(y; \boldsymbol{\theta})\right]\right)\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_t}.\end{aligned}$$

We can derive the following equation by replacing $\hat{q}(y)$ with $p(y; \boldsymbol{\theta}_{t+1})$ in (3),

$$\begin{aligned}&\bar{\boldsymbol{\theta}}_{t+1} - \boldsymbol{\theta}_t \simeq G_X{}^{-1}\partial\left(E_{p(y;\boldsymbol{\theta}_{t+1})}\left[l(y; \boldsymbol{\theta})\right]\right)\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_t} \\ &= G_X{}^{-1}\int p(y; \boldsymbol{\theta}_{t+1})\partial l(y; \boldsymbol{\theta})\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_t} d\mu(y). \end{aligned} \quad (17)$$

Here, we use the following approximation of $p(y; \boldsymbol{\theta}_{t+1})$ as,

$$\begin{aligned}p(y; \boldsymbol{\theta}_{t+1}) &\simeq p(y; \boldsymbol{\theta}_t) \\ &\quad + p(y; \boldsymbol{\theta}_t)\left(\partial l(y; \boldsymbol{\theta}_t)\right)^T(\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t).\end{aligned}$$

Using this formulation, (17) is approximated as,

$$\begin{aligned}&\bar{\boldsymbol{\theta}}_{t+1} - \boldsymbol{\theta}_t \\ &\simeq G_X{}^{-1}\int\Big(p(y; \boldsymbol{\theta}_t)\partial l(y; \boldsymbol{\theta}_t) \\ &\quad + p(y; \boldsymbol{\theta}_t)\partial l(y; \boldsymbol{\theta}_t)\partial l(y; \boldsymbol{\theta}_t)^T(\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t)\Big)d\mu(y) \\ &= G_X{}^{-1}\left(\int p(y; \boldsymbol{\theta}_t)\partial l(y; \boldsymbol{\theta}_t)\partial l(y; \boldsymbol{\theta}_t)^T d\mu(y)\right) \\ &\qquad \cdot(\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t) \\ &= G_X{}^{-1}G_Y(\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t) \\ &\simeq G_X{}^{-1}G_Y G_X{}^{-1}\partial L(Y^N; \boldsymbol{\theta}_t).\end{aligned}$$

And this gives the proof of theorem 2. Here, we used the following fact,

$$\int p(y;\boldsymbol{\theta}_t)\partial l(y;\boldsymbol{\theta}_t)d\mu(y) = 0.$$

When we use the proposed algorithm for continuous distribution, we have to use a Monte Carlo sampling method and $\bar{\boldsymbol{\theta}}_{t+1}$ does not converge to a point but fluctuate according to some distribution. We give the form of the asymptotic distribution of $\bar{\boldsymbol{\theta}}_{t+1}$. When N' samples are drawn according to $p(y;\boldsymbol{\theta}_{t+1})$ as $\{\bar{y}_1, \cdots, \bar{y}_{N'}\}$ and N' is sufficiently large, we define $\hat{p}(y;\boldsymbol{\theta}_{t+1})$ as,

$$\hat{p}(y;\boldsymbol{\theta}_{t+1}) = \frac{1}{N'}\sum_{s=1}^{N'}\delta(y-\bar{y}_s).$$

And we also denote the MLE as $\boldsymbol{\theta}^*_{t+1}$, when the target distribution is $\hat{p}(y;\boldsymbol{\theta}_{t+1})$. Take the 2nd expansion of (17), and we get,

$$\begin{aligned}
&\int \hat{p}(y;\boldsymbol{\theta}_{t+1})\partial l(y;\boldsymbol{\theta})\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_t} d\mu(y). \\
\simeq\ & E_{\hat{p}(y;\boldsymbol{\theta}_{t+1})}\left[\partial l(y;\boldsymbol{\theta}^*_{t+1})\right] && (18)\\
& -E_{\hat{p}(y;\boldsymbol{\theta}_{t+1})}\left[\partial^2 l(y;\boldsymbol{\theta}^*_{t+1})\right](\boldsymbol{\theta}^*_{t+1}-\boldsymbol{\theta}_t). && (19)
\end{aligned}$$

The first term which is shown as (18) is 0, and asymptotically $E_{\hat{p}(y;\boldsymbol{\theta}_{t+1})}\left[\partial^2 l(y;\boldsymbol{\theta}^*_{t+1})\right]$ is equivalent to $-G_Y(\boldsymbol{\theta}_{t+1})$, and $\boldsymbol{\theta}^*_{t+1}$ will normally distribute with $\boldsymbol{\theta}_{t+1}$ as its mean, and $G_Y(\boldsymbol{\theta}_{t+1})^{-1}/N'$ as its variance. From these results, we can see that the covariance matrix of the distribution of $\bar{\boldsymbol{\theta}}_{t+1}$ is ${G_X}^{-1}G_Y(\boldsymbol{\theta}_{t+1}){G_X}^{-1}/N'$.