The Regularized EM Algorithm

Haifeng Li
Department of Computer Science
University of California
Riverside, CA 92521
hli@cs.ucr.edu

Keshu Zhang
Human Interaction Research Lab
Motorola, Inc.
Tempe, AZ 85282
keshu.zhang@motorola.com

Tao Jiang
Department of Computer Science
University of California
Riverside, CA 92521
jiang@cs.ucr.edu

Abstract

The EM algorithm heavily relies on the interpretation of observations as incomplete data but it does not have any control on the uncertainty of missing data. To effectively reduce the uncertainty of missing data, we present a regularized EM algorithm that penalizes the likelihood with the mutual information between the missing data and the incomplete data (or the conditional entropy of the missing data given the observations). The proposed method maintains the advantage of the conventional EM algorithm, such as reliable global convergence, low cost per iteration, economy of storage, and ease of programming. We also apply the regularized EM algorithm to fit the finite mixture model. Our theoretical analysis and experiments show that the new method can efficiently fit the models and effectively simplify over-complicated models.

Introduction

In statistics and many related fields, the method of maximum likelihood is widely used to estimate an unobservable population parameter that maximizes the log-likelihood function

$$L(\Theta;\mathcal{X}) = \sum_{i=1}^{n} \log p(x_i|\Theta) \tag{1}$$

where the observations $\mathcal{X} = \{x_i | i = 1,\ldots,n\}$ are independently drawn from the distribution $p(x)$ parameterized by Θ. The Expectation-Maximization (EM) algorithm is a general approach to iteratively compute the maximum-likelihood estimates when the observations can be viewed as *incomplete data* (Dempster, Laird, & Rubin 1977). It has been found in most instances that the EM algorithm has the advantage of reliable global convergence, low cost per iteration, economy of storage, and ease of programming (Redner & Walker 1984). The EM algorithm has been employed to solve a wide variety of parameter estimation problems, especially when the likelihood function can be simplified by assuming the existence of additional but *missing* data $\mathcal{Y} = \{y_i | i = 1,\ldots,n\}$ corresponding to $\mathcal{X}$. The observations together with the missing data are called *complete data*. The EM algorithm maximizes the log-likelihood of the incomplete data by exploiting the relationship between the complete data and the incomplete data. In each iteration, two steps, called E-step and M-step, are involved. In the E-step, the EM algorithm determines the expectation of log-likelihood of the complete data based on the incomplete data and the current parameter

$$Q(\Theta|\Theta^{(t)}) = E\left(\log p(\mathcal{X},\mathcal{Y}|\Theta)\big|\mathcal{X},\Theta^{(t)}\right) \tag{2}$$

In the M-step, the algorithm determines a new parameter maximizing Q

$$\Theta^{(t+1)} = \arg\max_{\Theta} Q(\Theta|\Theta^{(t)}) \tag{3}$$

Each iteration is guaranteed to increase the likelihood, and finally the algorithm converges to a local maximum of the likelihood function.

Clearly, the missing data $\mathcal{Y}$ has strong affects on the performance of the EM algorithm since the optimal parameter Θ^* is obtained by maximizing $E(\log p(\mathcal{X},\mathcal{Y}|\Theta))$. For example, the EM algorithm finds a *local* maximum of the likelihood function, which depends on the choice of $\mathcal{Y}$. Since the missing data $\mathcal{Y}$ is totally unknown and is "guessed" from the incomplete data, how can we choose a suitable $\mathcal{Y}$ to make the solution more reasonable? This question is not addressed in the EM algorithm because the likelihood function does not reflect any influence of the missing data. In order to address the issue, a simple and direct method is to regularize the likelihood function with a suitable functional of the distribution of the complete data.

In this paper, we introduce a regularized EM (REM) algorithm to address the above issue. The basic idea is to regularize the likelihood function with the mutual information between the observations and the missing data or the conditional entropy of the missing data given the observations. The intuition behind is that we hope that the missing data have little uncertainty given the incomplete data because the EM algorithm implicitly assumes a strong relationship between the missing data and the incomplete data. When we apply the regularized EM algorithm to fit the finite mixture model, the new method can efficiently fit the models and effectively simplify over-complicated models.

The Regularized EM Algorithm

Simply put, the regularized EM algorithm tries to optimize the penalized likelihood

$$\widetilde{L}(\Theta;\mathcal{X}) = L(\Theta;\mathcal{X}) + \gamma P(\mathcal{X},\mathcal{Y}|\Theta) \tag{4}$$

where the regularizer P is a functional of the distribution of the complete data given Θ and the positive value γ is the so-called regularization parameter that controls the compromise between the degree of regularization of the solution and the likelihood function.

As mentioned before, the EM algorithm assumes the existence of missing data. Intuitively, we would like to choose the missing data that has a strong (probabilistic) relation with the observations, which implies that the missing data has little uncertainty given the observations. In other words, the observations contain a lot of *information* about the missing data and we can infer the missing data from the observations with a small error rate. In general, the information about one object contained in another object can be measured by either Kolmogorov (or algorithmic) mutual information based on the theory of Kolmogorov complexity or Shannon mutual information based on Shannon information theory.

Both the theory of Kolmogorov complexity (Li & Vitányi 1997) and Shannon information theory (Shannon 1948) aim at providing a means for measuring the quantity of *information* in terms of *bit*. In the theory of Kolmogorov complexity, the *Kolmogorov complexity* (or *algorithmic entropy*) $K(x)$ of a finite binary string x[1] is defined as the length of a shortest binary program p to compute x on an appropriate universal computer, such as a universal Turing machine. The conditional Kolmogorov complexity $K(x|y)$ of x relative to y is defined similarly as the length of a shortest program to compute x if y is furnished as an auxiliary input to the computation. The *Kolmogorov* (or *algorithmic*) *mutual information* is defined as $I(x:y) = K(y) - K(y|x, K(x))$ that is the information about x contained in y. Up to an additive constant term, $I(x:y) = I(y:x)$. Although $K(x)$ is the ultimate lower bound of any other complexity measures, $K(x)$ and related quantities are not Turing computable. Therefore, we can only try to approximate these quantities in practice.

In Shannon information theory, the quantity entropy plays a central role as measures of information, choice and uncertainty. Mathematically, Shannon's entropy of a discrete random variable X with a probability mass function $p(x)$ is defined as (Shannon 1948)

$$H(X) = -\sum_x p(x)\log p(x) \tag{5}$$

Entropy is the number of bits on the average required to describe a random variable. In fact, entropy is the minimum descriptive complexity of a random variable (Kolmogorov 1965). Consider two random variables X and Y with a joint distribution $p(x,y)$ and marginal distributions $p(x)$ and $p(y)$, respectively. The *conditional entropy* $H(X|Y)$ is defined as

$$\begin{aligned} H(X|Y) &= \sum_y p(y)H(X|Y=y) \\ &= -\sum_x\sum_y p(x,y)\log p(x|y) \end{aligned} \tag{6}$$

which measures how uncertain we are of X on the average when we know Y. The *mutual information* $I(X;Y)$ between X and Y is the relative entropy (or Kullback-Leibler distance) between the joint distribution $p(x,y)$ and the product distribution $p(x)p(y)$

$$I(X;Y) = \sum_x\sum_y p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \tag{7}$$

which is symmetric. Note that when X and Y are independent, Y can tell us nothing about X and it is easy to show $I(X;Y) = 0$ in this case. Besides, the relationship between entropy and mutual information $I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$ demonstrates that the mutual information measures the amount of information that one random variable contains about another one. For continuous random variables, the summation operation is replaced with integration in the definitions of entropy and related notions.

Clearly, the theory of Kolmogorov complexity and Shannon information theory are fundamentally different although they share the same purpose. Shannon information theory considers the uncertainty of the population but ignores each individual. On the other hand, the theory of Kolmogorov complexity considers the complexity of a single object in the ultimate compressed version irrespective of the manner in which the object arose. Besides, Kolmogorov thinks that information theory must precede probability theory, and not be based on it (Kolmogorov 1983b). To regularize the likelihood function, we prefer Shannon mutual information to Kolmogorov mutual information because we cannot precede probability theory since the goal is just to estimate the parameters of distributions. Besides, we do consider the characteristics of the population of missing data rather than a single object in this case. Moreover, Kolmogorov complexity is not computable and we have to approximate it in applications. In fact, entropy is a popular approximation to Kolmogorov complexity in practice because it is a computable upper bound of Kolmogorov complexity (Kolmogorov 1983a).

With Shannon mutual information as the regularizer, we have the regularized likelihood

$$\widetilde{L}(\Theta;\mathcal{X}) = L(\Theta;\mathcal{X}) + \gamma I(X;Y|\Theta) \tag{8}$$

where X is the random variable of observations and Y is the random variable of missing data. Because we usually do not know much about the missing data, we may naturally assume that Y follows a uniform distribution and thus $H(Y)$ is a constant value given the range of Y. Since $I(X;Y) = H(Y) - H(Y|X)$, we may also use the following regularized likelihood

$$\widetilde{L}(\Theta;\mathcal{X}) = L(\Theta;\mathcal{X}) - \gamma H(Y|X;\Theta) \tag{9}$$

[1]Other finite objects can be encoded as finite binary strings in a natural way.

Fano's inequality (Cover & Thomas 1991) provides us another evidence that the conditional entropy $H(Y|X)$ could be a good regularizer here. Suppose we know a random variable X and we wish to guess the value of the correlated random variable Y that takes values in $\mathcal{Y}$. Fano's inequality relates the probability of error in guessing the random variable Y to its conditional entropy $H(Y|X)$. Suppose we employ a function $\hat{Y} = f(X)$ to estimate Y. Define the probability of error $P_e = \Pr\{\hat{Y} \neq Y\}$. Fano's inequality is

$$H(P_e) + P_e \log(|\mathcal{Y}| - 1) \geq H(Y|X) \tag{10}$$

This inequality can be weakened to

$$1 + P_e \log |\mathcal{Y}| \geq H(Y|X) \tag{11}$$

Note that $P_e = 0$ implies that $H(Y|X) = 0$. In fact, $H(Y|X) = 0$ if and only if Y is a function of X (Cover & Thomas 1991). Fano's inequality indicates that we can estimate Y with a low probability of error only if the conditional entropy $H(Y|X)$ is small. Thus, the conditional entropy of missing variable given the observed variable(s) is clearly a good regularizer for our purpose.

To optimize (8) or (9), we only need slightly modify the M-step of the EM algorithm. Instead of (3), we use

$$\Theta^{(t+1)} = \arg\max_{\Theta} \widetilde{Q}(\Theta|\Theta^{(t)}) \tag{12}$$

where

$$\widetilde{Q}(\Theta|\Theta^{(t)}) = Q(\Theta|\Theta^{(t)}) + \gamma I(X; Y|\Theta) \tag{13}$$

or

$$\widetilde{Q}(\Theta|\Theta^{(t)}) = Q(\Theta|\Theta^{(t)}) - \gamma H(Y|X; \Theta) \tag{14}$$

The modified algorithm is called the regularized EM (REM) algorithm. We can easily prove the convergence of the REM algorithm in the framework of proximal point algorithm (Bertsekas 1999). For the objective function $f(\Theta)$, a generalized proximal point algorithm is defined by the iteration

$$\Theta^{(t+1)} = \arg\max_{\Theta}\{f(\Theta) - \beta_t d(\Theta, \Theta^{(t)})\} \tag{15}$$

where $d(\Theta, \Theta^{(t)})$ is a distance-like penalty function (i.e. $d(\Theta, \Theta^{(t)}) \geq 0$ and $d(\Theta, \Theta^{(t)}) = 0$ if and only if $\Theta = \Theta^{(t)}$), and β_t is a sequence of positive numbers. It is easy to show that the objective function $f(\Theta)$ increases with the iteration (15). In (Chretien & Hero 2000), it was shown that EM is a special case of proximal point algorithm implemented with $\beta_t = 1$ and a Kullback-type proximal penalty. In fact, the M-step of the EM algorithm can be represented as

$$\Theta^{(t+1)} = \arg\max_{\Theta}\left\{L(\Theta; \mathcal{X}) - E\left[\log \frac{p(\mathcal{Y}|\mathcal{X}; \Theta^{(t)})}{p(\mathcal{Y}|\mathcal{X}; \Theta)}\middle| \mathcal{X}, \Theta^{(t)}\right]\right\} \tag{16}$$

Thus, we can immediately prove the convergence of the REM algorithm by replacing $L(\Theta; \mathcal{X})$ with $\widetilde{L}(\Theta; \mathcal{X})$ in (16). Because the regularization term in (8) (or (9)) biases the searching space to some extent, we expect that the REM algorithm also converges faster than the plain EM algorithm, which will be confirmed in the experiments.

Finite Mixture Model

In this section, we apply the regularized EM algorithm to fit the finite mixture model. The finite mixture model arises as the fundamental model naturally in the areas of statistical machine learning. With the finite mixture model, we assume that the density associated with a population is a finite mixture of densities. Finite mixture densities can naturally be interpreted as that we have m component densities mixed together with mixing coefficients $\alpha_k, k = 1, \ldots, m$, which can be thought of as the *a priori* probabilities of each mixture component c_k, i.e. $\alpha_k = p(c_k)$. The mixture probability density functions have the form

$$p(x|\Theta) = \sum_{k=1}^{m} \alpha_k p(x|\theta_k) \tag{17}$$

where the parameters are $\Theta = (\alpha_1, \ldots, \alpha_m, \theta_1, \ldots, \theta_m)$ such that $\sum_{k=1}^{m} \alpha_k = 1$ and $\alpha_k \geq 0, k = 1, \ldots, m$; and each p is the density function of the component c_k that is parameterized by θ_k. [2]

For the finite mixture model, we usually employ the category information $\mathcal{C}$ associated with the observations $\mathcal{X}$ as the missing data, which indicates which component in the mixture produces the observation. In this section, we use the conditional entropy as the regularizer in particular. The reason will be clear later. Let C be a random variable taking values in $\{c_1, c_2, \ldots, c_m\}$ with probabilities $\alpha_1, \alpha_2, \ldots, \alpha_m$. Thus, we have

$$\begin{aligned}
&\widetilde{L}(\Theta; \mathcal{X}) = L(\Theta; \mathcal{X}) - \gamma H(C|X; \Theta) \\
&= \sum_{i=1}^{n} \log \sum_{k=1}^{m} \alpha_k p(x_i|\theta_k) \\
&\quad + \gamma \int \sum_{k=1}^{m} \frac{\alpha_k p(x|\theta_k)}{p(x|\Theta)} \log\left(\frac{\alpha_k p(x|\theta_k)}{p(x|\Theta)}\right) p(x|\Theta) dx
\end{aligned}$$

The corresponding $\widetilde{Q}$ is

$$\begin{aligned}
\widetilde{Q}(\Theta|\Theta^{(t)}) &= \sum_{k=1}^{m}\sum_{i=1}^{n} \log(\alpha_k) p(c_k|x_i; \Theta^{(t)}) \\
&\quad + \sum_{k=1}^{m}\sum_{i=1}^{n} \log(p(x_i|\theta_k)) p(c_k|x_i; \Theta^{(t)}) \\
&\quad + \gamma \int \sum_{k=1}^{m} \frac{\alpha_k p(x|\theta_k)}{p(x|\Theta)} \log \frac{\alpha_k p(x|\theta_k)}{p(x|\Theta)} p(x|\Theta) dx
\end{aligned}$$

In order to find $\alpha_k, k = 1, \ldots, m$, we introduce a Lagrangian

$$\mathcal{L} = \widetilde{Q}(\Theta|\Theta^{(t)}) - \lambda\left(\sum_{k=1}^{c} \alpha_k - 1\right) \tag{18}$$

[2] Here, we assume that all components have the same form of density for simplicity. More generally, the densities do not necessarily need belong to the same parametric family.

with multiplier λ for the constraint $\sum_{k=1}^{m} \alpha_k = 1$. Solving the Lagrangian $\mathcal{L}$, we obtain

$$\alpha_k^{(t+1)} = \frac{\sum_{i=1}^{n} p(c_k|x_i;\Theta^{(t)})(1+\gamma \log p(c_k|x_i;\Theta^{(t)}))}{\sum_{i=1}^{n}\sum_{k=1}^{m} p(c_k|x_i;\Theta^{(t)})(1+\gamma \log p(c_k|x_i;\Theta^{(t)}))} \tag{19}$$

To find $\theta_k, k = 1, \ldots, m$, we take the derivatives of $\widetilde{Q}$ with respect to θ_k

$$\frac{\partial \widetilde{Q}(\Theta|\Theta^{(t)})}{\partial \theta_k} = 0 \qquad k = 1, \ldots, m$$

For exponential families, it is possible to get an analytical expression for θ_k, as a function of everything else. Suppose that $p(x|\theta_k)$ has the regular exponential-family form (Barndorff-Nielsen 1978):

$$p(x|\theta_k) = \varphi^{-1}(\theta_k)\psi(x)e^{\theta_k^T t(x)} \tag{20}$$

where θ_k denotes an $r \times 1$ vector parameter, $t(x)$ denotes an $r \times 1$ vector of sufficient statistics, the superscript T denotes matrix transpose, and $\varphi(\theta_k)$ is given by

$$\varphi(\theta_k) = \int \psi(x)e^{\theta_k^T t(x)}dx \tag{21}$$

The term "regular" means that θ_k is restricted only to a convex set Ω such that equation (20) defines a density for all θ_k in Ω. Such parameters are often called natural parameters. The parameter θ_k is also unique up to an arbitrary nonsingular $r \times r$ linear transformation, as is the corresponding choice of $t(x)$. For example, *expectation parametrization* employs $\phi_k = E(t(x)|\theta_k)$, which is a both-way continuously differentiable mapping (Barndorff-Nielsen 1978).

For exponential families, we have

$$\phi_k^{(t+1)} = \frac{\sum_{i=1}^{n} t(x_i)p(c_k|x_i;\Theta^{(t)})(1+\gamma \log p(c_k|x_i;\Theta^{(t)}))}{\sum_{i=1}^{n} p(c_k|x_i;\Theta^{(t)})(1+\gamma \log p(c_k|x_i;\Theta^{(t)}))} \tag{22}$$

For a Gaussian mixture $p(x) = \sum_{k=1}^{m} \alpha_k N(\mu_k, \Sigma_k)$, we have

$$\mu_k^{(t+1)} = \frac{\sum_{i=1}^{n} x_i p(c_k|x_i;\Theta^{(t)})(1+\gamma \log p(c_k|x_i;\Theta^{(t)}))}{\sum_{i=1}^{n} p(c_k|x_i;\Theta^{(t)})(1+\gamma \log p(c_k|x_i;\Theta^{(t)}))} \tag{23}$$

$$\Sigma_k^{(t+1)} = \frac{\sum_{i=1}^{n} d_{ik} p(c_k|x_i;\Theta^{(t)})(1+\gamma \log p(c_k|x_i;\Theta^{(t)}))}{\sum_{i=1}^{n} p(c_k|x_i;\Theta^{(t)})(1+\gamma \log p(c_k|x_i;\Theta^{(t)}))} \tag{24}$$

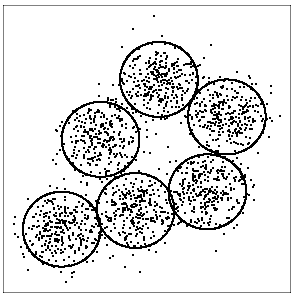


Figure 1: The simulated two-dimensional Gaussian mixture of six components, each of which contains 300 points.

where $d_{ik} = (x_i - \mu_k)(x_i - \mu_k)^T$.

When we apply the EM algorithm to fit the finite mixture model, we have to determine the number of components, which is usually referred as to model selection. Because the maximized likelihood is a non-decreasing function of the number of components (Figueiredo & Jain 2002), the plain EM algorithm cannot reduce a specified over-complicated model to a simpler model by itself. That is, if a larger number of components is specified, the plain EM algorithm cannot reduce it to the true but smaller number of components (i.e. a simpler model). Because over-complicated models introduce more uncertainty, [3] we expect that the REM algorithm in contrast will be able to automatically simplify over-complicated models to simpler ones through reducing the uncertainty of missing data. [4] Besides, note the conditional entropy of category information C given X

$$H(C|X) = -\int \sum_{k=1}^{m} p(c_k|x) \log(p(c_k|x))p(x)dx \tag{25}$$

is a non-decreasing function of the number of components because a larger m implies more choices and a larger entropy (Shannon 1948). In fact, $H(C|X)$ is minimized to 0 if $m = 1$, i.e. all data are from the same component. Thus, the term $-\gamma H(C|X)$ in $\widetilde{L}(\Theta; \mathcal{X})$ would support the merge of the components to reduce the entropy in the iterations of REM. On the other hand, the term $L(\Theta; \mathcal{X})$ supports keeping the number of components as large as possible to achieve a high likelihood. Finally, the REM algorithm reaches a balance between the likelihood and the conditional entropy and it reduces the number of components to some extent.

The model selection problem is an old problem and many criteria/methods have been proposed, such as Akaike's information criterion (AIC) (Akaike 1973), Bayesian inference criterion (BIC) (Schwarz 1978), Cheeseman-Stutz criterion (Cheeseman & Stutz 1995), minimum message length (MML) (Wallace & Boulton 1968), and minimum description length (MDL) (Rissanen 1985). However, we do not attempt to compare our method with the aforementioned methods because the goal of our method is to reduce the un-

[3] The more choices, the more entropy (Shannon 1948).

[4] In fact, the purged components still exist in the mixture model. But their probabilities are close to zero.

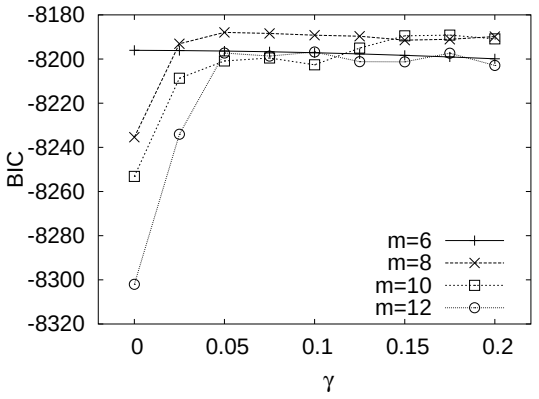


Figure 2: The BIC scores of the learned models. Here, m is the specified number of components and γ is the regularization factor.

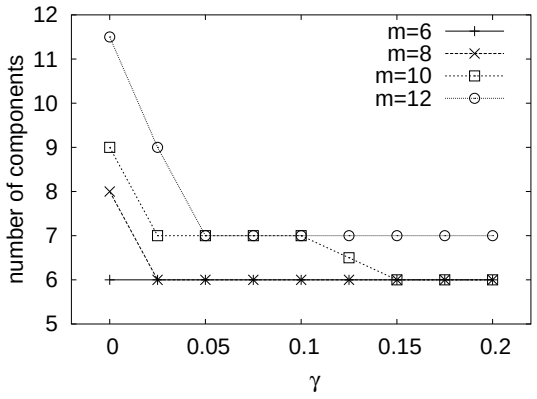


Figure 3: The number of components in the learned models.

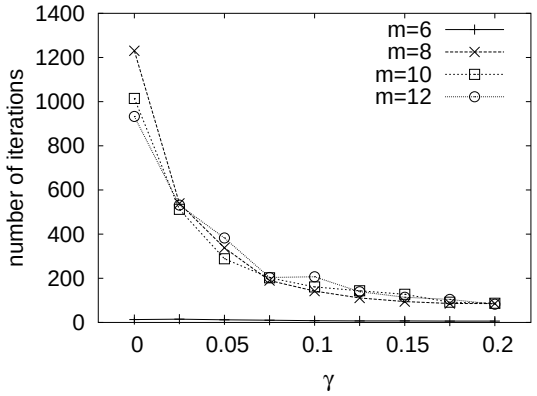


Figure 4: The number of iterations.

certainty of missing data rather than to determine the number of components. In fact, simplifying an over-complicated model is only a byproduct of our method obtained through reducing the uncertainty of missing data. Besides, our method is not a comprehensive method to determine the number of components since it cannot extend an over-simple model to the true model.

Demonstration

In this section, we present an example to illustrate the performance of the REM algorithm on a two-dimensional Gaussian mixture. The mixture contains six components, each of which has 300 samples. The data is shown in Figure 1. In the experiments, we use k-means to give the initial partition. The stop criterion in iterations is that the increase in the regularized log-likelihood (9) is less than 10^{-7}. In the experiments, we test the REM algorithm with different numbers of components and regularization factor γ. Note that the REM algorithm reduces to the plain EM algorithm when γ is set to 0. With each setting, we run the algorithm 30 times. The medians of the results are reported here.

To measure the quality of learned models, we employ BIC/MDL [5] (Schwarz 1978; Rissanen 1985) here for simplicity. Let v be the number of independent parameters to be estimated in the model. [6] BIC can be approximated by

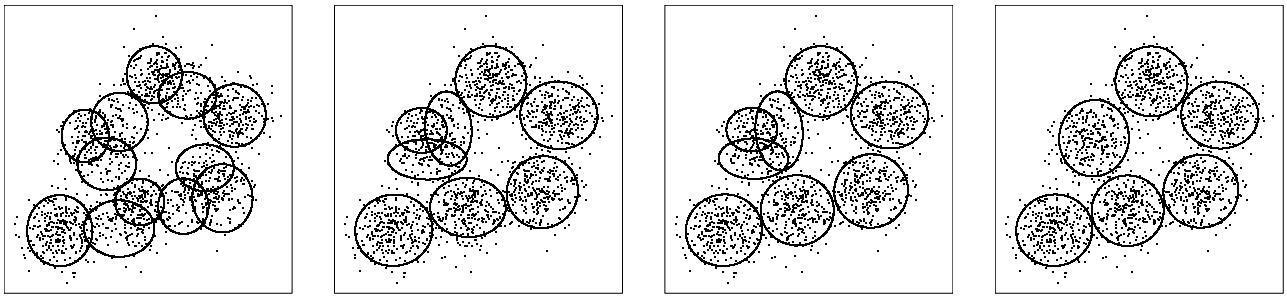
$$BIC \approx L(\hat{\Theta}) - \frac{1}{2} v \log n \qquad (26)$$

A large BIC score indicates that the model has a large posteriori and thus is most likely close to the true model. As shown in Figure 2, the REM algorithm achieves much larger BIC scores than the plain EM algorithm (i.e. the $\gamma = 0$ case) when the number of components is incorrectly specified. When the specified number of components is correct (i.e. $m = 6$), the plain EM and REM obtain similar BIC scores. We also observe that, if a suitable γ is employed, the REM algorithm may achieve a higher BIC score than the EM algorithm even when the number of components is correctly set for the EM algorithm. For example, the REM algorithm achieves a higher BIC score with $\gamma = 0.05$ and $m = 8$ than that of the EM algorithm with $m = 6$. This study also suggests that we may choose γ by BIC/MDL. Further research on determining optimal γ is in progress.

Besides BIC/MDL scores, we also investigate the number of components in the learned models. In this study, we regard a component as purged out of the model if its priori probability is less than 0.01. The (median of) learned numbers of components are shown in Figure 3. As shown in the figure, the REM algorithm can usually reduce an incorrectly specified number of components to the correct one (i.e. 6). We also observe that the REM algorithm does not reduce the models to over-simplified ones (e.g. the learned number of components is less than 6) in all cases. It is well-known that the plain EM algorithm may also return empty clusters (corresponding to components with zero probability), which is confirmed in our experiments. For $m = 10$ and $m = 12$, we observe that the EM algorithm may return fewer (say 9 or 11) components. Compared with the true model, however, it is still far from perfection.

[5] BIC coincides with the two-stage form of MDL (Hansen & Yu 2001).

[6] We consider only the parameters of the components with non-zero probabilities.



(a) Initialization (b) Iteration 50 (c) Iteration 100 (d) Final Result ($t = 252$)

Figure 5: Trace of the REM algorithm with $\gamma = 0.1$ and $m = 12$.

It is known that the EM algorithm may converge very slowly in practice. In the experiments, we find that the REM algorithm converges much faster than the EM algorithm as shown in Figure 4. The reason may be that the regularization biases the search space toward more likely regions so that it improves the efficiency of iterations. Interestingly, the number of iterations seems to decrease with the increase of γ.

Finally, we give a graphical representation of iterations of the REM algorithm in Figure 5. Here, we set $\gamma = 0.1$ and $m = 12$. After 50 iterations, the estimated model can already describe the shape of the data well. Finally, the REM algorithm converges at iteration 252 with six components that are very close to the true model. The extra components (not represented in the figure) are successively purged from the model due to their zero *a priori* probabilities.

Conclusion

We have proposed a regularized EM algorithm to control the uncertainty of missing data. The REM algorithm tries to maximize the likelihood and the information about the missing data contained in the observations. Besides reducing the uncertainty of missing data, the proposed method maintains the advantage of the conventional EM algorithm. When we apply the regularized EM algorithm to fit the finite mixture model, it can efficiently fit the models and effectively simplify over-complicated models. The convergence properties of the REM algorithm would be an interesting future research topic.

References

Akaike, H. 1973. Information theory and an extension of the maximum likelihood principle. In Petrov, B. N., and Csaki, F., eds., *Second International Symposium on Information Theory*, 267–281.

Barndorff-Nielsen, O. E. 1978. *Information and Exponential Families in Statistical Theory*. New York: Wiley.

Bertsekas, D. P. 1999. *Nonlinear Programming*. Belmont, MA: Athena Scientific, 2nd edition.

Cheeseman, P., and Stutz, J. 1995. Bayesian classification (AutoClass): Theory and results. In *Advances in Knowledge Discovery and Data Mining*, 153–180. Menlo Park, CA: AAAI Press.

Chretien, S., and Hero, A. O. 2000. Kullback proximal algorithms for maximum likelihood estimation. *IEEE Transactions on Information Theory* 46(5):1800–1810.

Cover, T. M., and Thomas, J. A. 1991. *Elements of Information Theory*. New York: John Wiley & Sons.

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

Figueiredo, M. A. T., and Jain, A. K. 2002. Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24(3):381–396.

Hansen, M. H., and Yu, B. 2001. Model selection and the principle of minimum description length. *Journal of the American Statistical Association* 96(454):746–774.

Kolmogorov, A. N. 1965. Three approaches for defining the concept of information quantity. *Information Transmission* 1:3–11.

Kolmogorov, A. N. 1983a. Combinatorial foundations of information theory and the calculus of probabilities. *Russian Mathematical Surveys* 38:29–40.

Kolmogorov, A. N. 1983b. *On logical foundations of probability theory*, volume 1021 of *Lecture Notes in Mathematics*. New York: Springer. 1–5.

Li, M., and Vitányi, P. 1997. *An Introduction to Kolmogorov Complexity and its Applications*. New York: Springer-Verlag, 2nd edition.

Redner, R. A., and Walker, H. F. 1984. Mixture densities, maximum likelihood and the EM algorithm. *SIAM Review* 26(2):195–239.

Rissanen, J. 1985. Minimum description length principle. In Kotz, S., and Johnson, N. L., eds., *Encyclopedia of Statistical Sciences*, volume 5. New York: Wiley. 523–527.

Schwarz, G. 1978. Estimating the dimension of a model. *The Annals of Statistics* 6(2):461–464.

Shannon, C. E. 1948. A mathematical theory of communication. *Bell System Techical Journal* 27:379–423 and 623–656.

Wallace, C. S., and Boulton, D. M. 1968. An information measure for classification. *Computer Journal* 11:185–194.