A New View of the EM Algorithm that Justifies Incremental and Other Variants

Radford M. Neal and Geoffrey E. Hinton

Department of Computer Science
University of Toronto
10 King's College Road
Toronto, Canada M5S 1A4

12 February 1993

Abstract

We present a new view of the EM algorithm for maximum likelihood estimation in situations with unobserved variables. In this view, both the E and the M steps of the algorithm are seen as maximizing a joint function of the model parameters and of the distribution over unobserved variables. From this perspective, it is easy to justify an incremental variant of the algorithm in which the distribution for only one of the unobserved variables is recalculated in each E step. This variant is shown empirically to give faster convergence in a mixture estimation problem. A wide range of other variant algorithms are also seen to be possible.

Some key words: EM algorithm, incremental algorithms, free energy, mixtures

Submitted to *Biometrika*

1 Introduction

The Expectation-Maximization (EM) algorithm finds maximum likelihood parameter estimates in problems where some variables were unobserved. Its widespread applicability was first discussed by Dempster, Laird, and Rubin (1977).

The EM algorithm estimates the parameters iteratively, starting from some initial guess. Each iteration consists of an Expectation (E) step, which finds the distribution for the unobserved variables, given the known values for the observed variables and the current estimate of the parameters, and a Maximization (M) step, which re-estimates the parameters to be those with maximum likelihood, under the assumption that the distribution found in the E step is correct. It can be shown that each such iteration improves the true likelihood, or leaves it unchanged, if a local maximum has already been reached.

The M step of the algorithm may be only partially implemented, with the new estimate for the parameters improving the likelihood given the distribution found in the E step, but not necessarily maximizing it. Such a partial M step always results in the true likelihood improving as well. Dempster, *et al* refer to such variants as "generalized EM (GEM)" algorithms. A sub-class of GEM algorithms of wide applicability, the "Expectation-Conditional Maximization (ECM)" algorithms, have been developed by Meng and Rubin (1992).

In many cases, partial implementation of the E step is also natural. The unobserved variables are commonly independent, and influence the likelihood of the parameters only through simple sufficient statistics. If these statistics can be updated incrementally when the distribution for one of the variables is re-calculated, it makes sense to immediately re-estimate the parameters before performing the E step for the next unobserved variable, as the new information from this partial E step is then utilized immediately, speeding convergence. An incremental algorithm along these general lines was investigated by Nowlan (1991). Up until now, however, such incremental variants of the EM algorithm have not been formally justified.

We present here a new view of the EM algorithm, in which it is seen as maximizing a joint function of the parameters and of the distribution over the unobserved variables that is analogous to the "free energy" function used in statistical physics. The E step maximizes this function with respect to the distribution over unobserved variables; the M step with respect to the parameters. From this viewpoint, it is clear that other approaches to jointly maximizing this function will also lead to a maximum of the true likelihood. This justifies a wide range of variants of the EM algorithm, including incremental versions, "sparse" versions, in which most iterations update only that part of the distribution for an unobserved variable pertaining to its most likely values, and "winner-take-all" versions, in which, for early iterations, the distributions over unobserved variables are restricted to those in which a single value has probability one.

We include a brief demonstration showing that use of an incremental algorithm speeds convergence for a simple mixture estimation problem.

2 General theory

Suppose that we have observed the value of some random variable, Z, but not the value of another variable, Y, and that based on this data, we wish to find the maximum likelihood estimate for the parameters of a model for Y and Z. We assume that this problem is not easily solved directly, but that the corresponding problem in which Y is also known would be more tractable. For simplicity, we assume here that Y has a finite range, as is often the case, but the results can be generalized.

Assume that the joint probability for Y and Z is parameterized using θ, as $P(y, z \mid \theta)$. The marginal probability for Z is then $P(z \mid \theta) = \sum_y P(y, z \mid \theta)$. Given observed data, z, we wish to find the value of θ that maximizes the log likelihood, $L(\theta) = \log P(z \mid \theta)$.

The EM algorithm starts with some initial guess at the maximum likelihood parameters, $\theta^{(0)}$, and then proceeds to iteratively generate successive estimates, $\theta^{(1)}, \theta^{(2)}, \ldots$ by repeatedly applying the following two steps, for $t = 1, 2, \ldots$

$$\left.\begin{array}{ll} \textbf{E Step:} & \text{Compute a distribution } \tilde{P}^{(t)} \text{ over the range of } Y \text{ such that} \\ & \tilde{P}^{(t)}(y) = P(y \mid z, \theta^{(t-1)}). \\[1ex] \textbf{M Step:} & \text{Set } \theta^{(t)} \text{ to that } \theta \text{ which maximizes } E_{\tilde{P}^{(t)}}[\log P(y, z \mid \theta)]. \end{array}\right\} \quad (1)$$

Here, $E_{\tilde{P}}[\,\cdot\,]$ denotes expectation with respect to the distribution over the range of Y given by $\tilde{P}$. Note that in preparation for the later generalization, the standard algorithm has here been expressed in a slightly non-standard fashion.

The E step of the algorithm can be seen as representing the unknown value for Y by a distribution of values, and the M step as then performing maximum likelihood estimation for the joint data obtained by combining this with the known value of Z, an operation that is assumed to be feasible. As shown by Dempster, *et al*, each EM iteration increases the true log likelihood, $L(\theta)$, unless a local maximum has already been reached. This is true also for any GEM algorithm, in which only a partial maximization is performed in the M step, with $\theta^{(t)}$ simply set to some value such that $E_{\tilde{P}^{(t)}}[\log P(z,\, y \mid \theta^{(t)})]$ is greater than $E_{\tilde{P}^{(t)}}[\log P(y,\, z \mid \theta^{(t-1)})]$ (or is equal if convergence has been reached).

In order to make sense of the corresponding idea of partially performing the E step, we take a new view of the EM algorithm and its variants, in which both the E and the M steps are seen as maximizing, or at least increasing, the same function, $F(\tilde{P}, \theta)$. We will show that if a local maximum of F occurs at θ^* and $\tilde{P}^*$, then a local maximum of L occurs at θ^* as well. We can therefore contemplate a wide variety of algorithms for maximizing L by means of maximizing F, among which are incremental algorithms, in which each E step updates only one factor of $\tilde{P}$, corresponding to one data item.

The function $F(\tilde{P}, \theta)$ is defined as follows:

$$F(\tilde{P}, \theta) \;=\; E_{\tilde{P}}[\log P(y,\, z \mid \theta)] \;+\; H(\tilde{P}) \tag{2}$$

where $H(\tilde{P}) = -E_{\tilde{P}}[\log \tilde{P}(y)]$ is the entropy of the distribution $\tilde{P}$. Note that F is defined with respect to a particular value for the observed data, z, which is fixed throughout. For simplicity, we will assume here that $P(y,\, z \mid \theta)$ is never zero, so that F is always finite, but this restriction is not essential. We also need to assume that $P(y,\, z \mid \theta)$ is a continuous function of θ, from which we can conclude that F is a continuous function of both θ and $\tilde{P}$.

Apart from a change of sign, the function F is analogous to the "free energy" that plays a central role in statistical physics, provided that the physical states are taken to be values of Y, and the "energy" of a state is $-\log P(y,\, z \mid \theta)$. The following two lemmas state properties of F corresponding to well-known facts from statistical physics — that the "Boltzmann" distribution over states minimizes the free energy, and that the free energy is related to the log of the "partition function".

Lemma 1 *For a fixed value of θ, there is a unique distribution, $\tilde{P}_\theta$, that maximizes $F(\tilde{P}, \theta)$, given by $\tilde{P}_\theta(y) = P(y \mid z,\, \theta)$. Furthermore, this $\tilde{P}_\theta$ varies continuously with θ.*

PROOF. In maximizing F with respect to $\tilde{P}$, we are constrained by the requirement that $\tilde{P}(y) \geq 0$ for all y. Solutions with $\tilde{P}(y) = 0$ for some y are not possible, however — one can easily show that the slope of the entropy is infinite at such points, so that moving slightly away from the boundary will increase F. Any maximum of F must therefore occur at a critical point subject to the constraint that $\sum_y \tilde{P}(y) = 1$, and can be found using a Lagrange multiplier. At such a maximum, $\tilde{P}_\theta$, the gradient of F with respect to the components of $\tilde{P}$ will be normal to the constraint surface, i.e. for some λ and for all y,

$$\lambda \;=\; \frac{\partial F}{\partial \tilde{P}(y)}(\tilde{P}_\theta) \;=\; \log P(y,\, z \mid \theta) \;-\; \log \tilde{P}_\theta(y) \;-\; 1 \tag{3}$$

From this, it follows that $\tilde{P}_\theta(y)$ must be proportional to $P(y,\, z \mid \theta)$. Normalizing so that $\sum_y \tilde{P}_\theta(y) = 1$, we have $\tilde{P}_\theta(y) = P(y \mid z,\, \theta)$ as the unique solution. That $\tilde{P}_\theta$ varies continuously with θ follows immediately from our assumption that $P(y,\, z \mid \theta)$ does.

Lemma 2 *If $\tilde{P}(y) = P(y \mid z,\, \theta)$ then $F(\tilde{P}, \theta) = \log P(z \mid \theta)$.*

Proof. If $\tilde{P}(y) = P(y \mid z,\, \theta)$, then

$$\begin{aligned} F(\tilde{P}, \theta) &= E_{\tilde{P}}[\log P(y,\, z \mid \theta)] \;+\; H(\tilde{P}) \\ &= E_{\tilde{P}}[\log P(y,\, z \mid \theta)] \;-\; E_{\tilde{P}}[\log P(y \mid z,\, \theta)] \\ &= E_{\tilde{P}}[\log P(y,\, z \mid \theta) - \log P(y \mid z,\, \theta)] \\ &= E_{\tilde{P}}[\log P(z \mid \theta)] \\ &= \log P(z \mid \theta) \end{aligned}$$

An iteration of the the standard EM algorithm can be expressed in terms of the function F as follows:

$$\left.\begin{aligned} &\textbf{E Step:} \quad \text{Set } \tilde{P}^{(t)} \text{ to that } \tilde{P} \text{ which maximizes } F(\tilde{P}, \theta^{(t-1)}). \\ &\textbf{M Step:} \quad \text{Set } \theta^{(t)} \text{ to that } \theta \text{ which maximizes } F(\tilde{P}^{(t)}, \theta). \end{aligned}\right\} \tag{4}$$

Theorem 1 *The iterations given by (1) and by (4) are equivalent.*

Proof. That the E steps of the iterations are equivalent follows directly from Lemma 1. That the M steps are equivalent follows from the fact that the entropy term in the definition of F in equation (2) does not depend on θ.

Once the EM iterations have been expressed in the form (4), it is clear that the algorithm converges to values $\tilde{P}^*$ and θ^* that locally maximize $F(\tilde{P}, \theta)$ (ignoring the possibility of convergence to a saddle point). The following theorem shows that, in general, finding a local maximum for $F(\tilde{P}, \theta)$ will also yield a local maximum for $L(\theta)$, justifying not only the standard algorithm of (4), but variants of it in which the E and M steps are performed partially, as well as algorithms in which the maximization is done with respect to $\tilde{P}$ and θ simultaneously.

Theorem 2 *If $F(\tilde{P}, \theta)$ has a local maximum at $\tilde{P}^*$ and θ^*, then $L(\theta)$ has a local* *maximum at θ^* as well. Similarly, if F has a global maximum at $\tilde{P}^*$ and θ^*, then L has a global maximum at θ^*.*

PROOF. By combining Lemmas 1 and 2, we see that $L(\theta) = \log P(z \mid \theta) = F(\tilde{P}_\theta, \theta)$, for any θ, and that, in particular, $L(\theta^*) = F(\tilde{P}_{\theta^*}, \theta^*) = F(\tilde{P}^*, \theta^*)$. To show that θ^* is a local maximum of L, we need to show that there is no $\theta^\dagger$ near to θ^* for which $L(\theta^\dagger) > L(\theta^*)$. To see this, note that if such a $\theta^\dagger$ existed, then we would also have $F(\tilde{P}^\dagger, \theta^\dagger) > F(\tilde{P}^*, \theta^*)$, where $\tilde{P}^\dagger = \tilde{P}_{\theta^\dagger}$. But since $\tilde{P}_\theta$ varies continuously with θ, $\tilde{P}^\dagger$ must be near to $\tilde{P}^*$, contradicting the assumption that F has a local maximum at $\tilde{P}^*$ and θ^*. The proof for global maxima is analogous, but without the restriction to nearby values of θ. The assumptions of continuity are unnecessary for this latter result.

3 Incremental algorithms

In typical applications, we wish to find the maximum likelihood parameter estimate given a number of independent data items. The observed variable, Z, can then be decomposed as $(Z_1, \ldots, Z_n)$, and the unobserved variable, Y, as $(Y_1, \ldots, Y_n)$. The joint probability for Y and Z can be factored as $P(y,\, z \mid \theta) = \prod_i P(y_i,\, z_i \mid \theta)$. Often, the data items are also identically distributed, but we will not need to assume that here.

An incremental variant of the EM algorithm that exploits this structure can be justified on the basis of Theorem 2. Note that since the Y_i are independent, we can restrict the search for a maximum of F to distributions $\tilde{P}$ that factor as $\tilde{P}(y) = \prod_i \tilde{P}_i(y_i)$, since $\tilde{P}$ will have this form at the maximum. We can then write F in the form $F(\tilde{P}, \theta) = \sum_i F_i(\tilde{P}_i, \theta)$, where

$$F_i(\tilde{P}_i, \theta) \;=\; E_{\tilde{P}_i}[\log P(y_i,\, z_i \mid \theta)] \;+\; H(\tilde{P}_i) \tag{5}$$

An incremental algorithm using the following iteration can then be used to maximize F, and hence L, starting from a guess at the parameters, $\theta^{(0)}$, and a guess at the distribution, $\tilde{P}_i^{(0)}$, which may or may not be consistent with $\theta^{(0)}$:

$$\left.\begin{array}{ll}
\textbf{E Step:} & \text{Choose some data item, } i \text{, to be updated.} \\
& \text{Set } P_j^{(t)} = P_j^{(t-1)} \text{ for } j \neq i. \text{ (This needn't take any time.)} \\
& \text{Set } P_i^{(t)} \text{ to that } \tilde{P}_i \text{ which maximizes } F_i(\tilde{P}_i, \theta^{(t-1)}), \\
& \text{given by } \tilde{P}_i^{(t)}(y_i) = P(y_i \mid z_i,\, \theta^{(t-1)}). \\
\textbf{M Step:} & \text{Set } \theta^{(t)} \text{ to that } \theta \text{ which maximizes } F(\tilde{P}^{(t)}, \theta) \text{, or,} \\
& \text{equivalently, which maximizes } E_{\tilde{P}^{(t)}}[\log P(y,\, z \mid \theta)].
\end{array}\right\} \tag{6}$$

Data items might be selected for updating in the E step cyclicly, or by some scheme that gives preference to data items for which $\tilde{P}_i$ has not yet stabilized.

Each E step of the above algorithm requires looking at only a single data item, but, as written, it appears that the M step requires looking at all components of $\tilde{P}$. This can be avoided in the common case where the inferential import of the complete data can be summarized by a vector of sufficient statistics that can be incrementally updated, as is the case with models in the exponential family.

Letting this vector of sufficient statistics be $s(y,z) = \sum_i s_i(y_i, z_i)$, the standard EM iteration of (1) can be implemented as follows:

$$\left.\begin{array}{ll}\textbf{E Step:} & \text{Set } \tilde{s}^{(t)} \;=\; E_{\tilde{P}}[s(y,z)], \text{ where } \tilde{P}(y) = P(y \mid z,\, \theta^{(t-1)}). \\ & (\text{In detail, set } \tilde{s}^{(t)} = \sum_i \tilde{s}_i^{(t)}, \text{ with } \tilde{s}_i^{(t)} = E_{\tilde{P}_i}[s_i(y_i, z_i)], \\ & \text{where } \tilde{P}_i(y_i) = P(y_i \mid z_i,\, \theta^{(t-1)}).) \\ \textbf{M Step:} & \text{Set } \theta^{(t)} \text{ to that } \theta \text{ with maximum likelihood given } \tilde{s}^{(t)}.\end{array}\right\} \quad (7)$$

Similarly, the iteration of (6) can be implemented using sufficient statistics that are maintained incrementally, starting with an initial guess, $\tilde{s}_i^{(0)}$, which may or may not be consistent with $\theta^{(0)}$. Subsequent iterations proceed as follows:

$$\left.\begin{array}{lll}\textbf{E Step:} & \text{Choose some data item, } i, \text{ to be updated.} & \\ & \text{Set } \tilde{s}_j^{(t)} \;=\; \tilde{s}_j^{(t-1)} \text{ for } j \neq i. \text{ (This needn't take any time.)} & \\ & \text{Set } \tilde{s}_i^{(t)} \;=\; E_{\tilde{P}_i}[s_i(y_i, z_i)], \text{ where } \tilde{P}_i(y_i) = P(y_i \mid z_i,\, \theta^{(t-1)}). & \\ & \text{Set } \tilde{s}^{(t)} \;=\; \tilde{s}^{(t-1)} - \tilde{s}_i^{(t-1)} + \tilde{s}_i^{(t)}. & \\ \textbf{M Step:} & \text{Set } \theta^{(t)} \text{ to that } \theta \text{ with maximum likelihood given } s^{(t)}. &\end{array}\right\} \quad (8)$$

In iteration (8), both the E and the M steps take constant time, independent of the number of data items. A cycle of n such iterations, visiting each data item once, will sometimes take only slightly more time than one iteration of the standard algorithm, and should make more progress, since the distributions for each variable found in the partial E steps are utilized immediately, instead of being held until the distributions for all the unobserved variables have been found. Nearly as fast convergence may be obtained with an intermediate variant of the algorithm, in which each E step recomputes the distributions for several data items (but many fewer than n). Use of this intermediate variant rather than the pure incremental algorithm reduces the amount of time spent in performing the M steps.

Note that an algorithm based on iteration (8) must save the last value computed for each $\tilde{s}_i$, so that its contribution to $\tilde{s}$ may be removed when a new value for $\tilde{s}_i$ is computed. This requirement will generally not be onerous. The incremental update of $\tilde{s}$ could potentially lead to problems with cumulative round-off error. If necessary, this accumulation can be avoided in several ways — one could use a fixed-point representation of $\tilde{s}$, in which addition and subtraction is exact, for example, or recompute $\tilde{s}$ non-incrementally at infrequent intervals.

An incremental variant of the EM algorithm somewhat similar to that of (8) was investigated by Nowlan (1991). His variant does not maintain strictly accurate sufficient statistics, however. Rather, it uses statistics computed as an exponentially decaying average of recently-visited data points, with iterations of the following form:

$$\left.\begin{array}{ll}
\textbf{E Step:} & \text{Select the next data item, } i \text{, for updating.} \\
& \text{Set } \tilde{s}_i^{(t)} \; = \; E_{\tilde{P}_i}[s_i(y_i, z_i)], \text{ where } \tilde{P}_i(y_i) = P(y_i \mid z_i,\, \theta^{(t-1)}). \\
& \text{Set } \tilde{s}^{(t)} \; = \; \gamma \tilde{s}^{(t-1)} + \tilde{s}_i^{(t)}. \\
\textbf{M Step:} & \text{Set } \theta^{(t)} \text{ to that } \theta \text{ with maximum likelihood given } s^{(t)}.
\end{array}\right\} \quad (9)$$

where $0 < \gamma < 1$ is a decay constant.

The above algorithm will not converge to the exact answer, at least not if γ is kept at some fixed value. It is found empirically, however, that it can converge to the vicinity of the correct answer more rapidly than the standard EM algorithm. When the data set is large and redundant, one might expect that, with an appropriate value for γ, this algorithm could be faster than the incremental algorithm of (8), since it can forget out-of-date statistics more rapidly.

4 Demonstration for a mixture model

In order to demonstrate that the incremental algorithm of (8) can speed convergence, we have applied it to a simple mixture of Gaussians problem. The algorithm using iteration (9) was also tested.

In the Gaussian mixture model, the observed variables, Z_i, are real-valued, and the unobserved variables, Y_i, are binary, indicating from which of two Gaussian distributions the corresponding observed variable was generated. The joint probability (density) for Y_i and Z_i given parameters $\theta = (\alpha, \mu_0, \sigma_0, \mu_1, \sigma_1)$ is as follows:

$$
P(y_i,\, z_i \mid \alpha,\, \mu_0,\, \sigma_0,\, \mu_1,\, \sigma_1)
= \begin{cases} (1-\alpha)\,(2\pi\sigma_0^2)^{-\frac{1}{2}}\,\exp\Big(-\tfrac{1}{2}(z_i-\mu_0)^2/\sigma_0^2\Big) & \text{if } y_i = 0 \\ \alpha\,(2\pi\sigma_1^2)^{-\frac{1}{2}}\,\exp\Big(-\tfrac{1}{2}(z_i-\mu_1)^2/\sigma_1^2\Big) & \text{if } y_i = 1 \end{cases} \tag{10}
$$

For this problem, the vector of sufficient statistics for data item i is

$$
s_i(y_i, z_i) \;=\; \langle\,(1-y_i),\; (1-y_i)z_i,\; (1-y_i)z_i^2,\; y_i,\; y_i z_i,\; y_i z_i^2\,\rangle \tag{11}
$$

Given $s(y,z) = \sum_i s_i(y_i, z_i) = (n_0,\, m_0,\, q_0,\, n_1,\, m_1,\, q_1)$, the maximum likelihood parameter estimates are given by $\alpha = n_1/(n_0+n_1)$, $\mu_0 = m_0/n_0$, $\sigma_0^2 = q_0/n_0 - (m_0/n_0)^2$, $\mu_1 = m_1/n_1$, and $\sigma_1^2 = q_1/n_1 - (m_1/n_1)^2$.

We synthetically generated a sample of 1000 points, z_i, from this distribution with $\alpha = 0.3$, $\mu_0 = 0$, $\sigma_0 = 1$, $\mu_1 = -0.2$, and $\sigma_1 = 0.1$. We then applied the standard algorithm of (7) and the incremental algorithm of (8) to this data. As initial parameter values, we used $\alpha^{(0)} = 0.5$, $\mu_0^{(0)} = +1.0$, $\sigma_0^{(0)} = 1$, $\mu_1^{(0)} = -1$, and $\sigma_1^{(0)} = 1$. For the incremental algorithm, a single iteration of the standard algorithm was then performed to initialize the distributions for the unobserved variables. This is not necessarily the best procedure, but was done to avoid any arbitrary selection for the starting distributions, which would affect the comparison with the standard algorithm. The incremental algorithm visited data points cyclicly.

Both algorithms converged to identical maxima of L, at which $\alpha^* = 0.269$, $\mu_0^* = -0.016$, $\sigma_0^* = 0.959$, $\mu_1^* = -0.193$, and $\sigma_1^* = 0.095$. Special measures to control round-off error in the incremental algorithm were found to be unnecessary in this case (using 64-bit floating-point numbers). The rates of convergence of the two algorithms are shown in Figure 1, in which the log likelihood, L, is plotted as a function of the number of "passes" — a pass being one iteration for the standard algorithm, and n iterations for the incremental algorithm. (In both case, a pass visits each data point once.) As can be seen, the incremental algorithm reached any given level of L in about half as many passes as the standard algorithm.

Unfortunately, each pass of the incremental algorithm required about twice as much computation time as did a pass of the standard algorithm, due primarily to the computation required to perform an M step after visiting every data point. This cost can be greatly reduced by using an intermediate algorithm in which each E step recomputes the distributions for the next ten data points. The rate of convergence with this algorithm is virtually indistinguishable from that of the

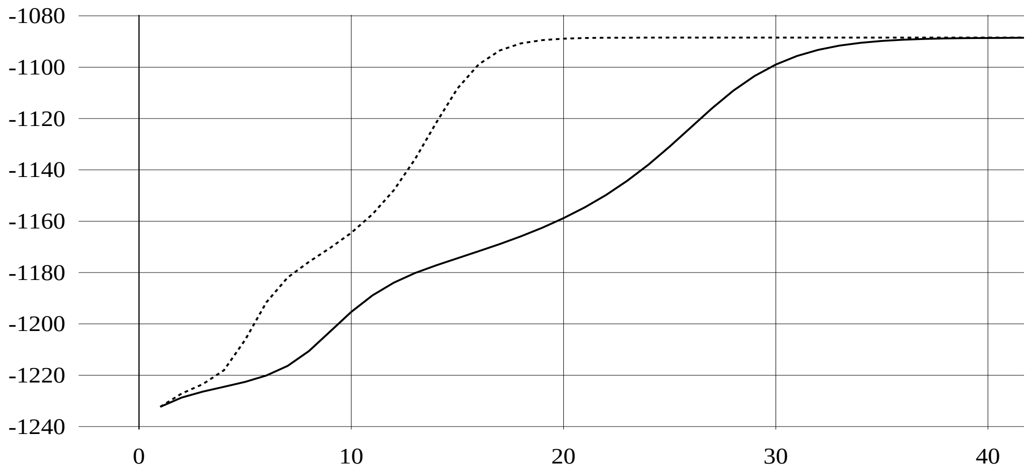

Figure 1: Comparison of convergence rates for the standard EM algorithm (solid line) and the incremental algorithm (dotted line). The log likelihood is shown on the vertical axis, the number of passes of the algorithm on the horizontal axis.

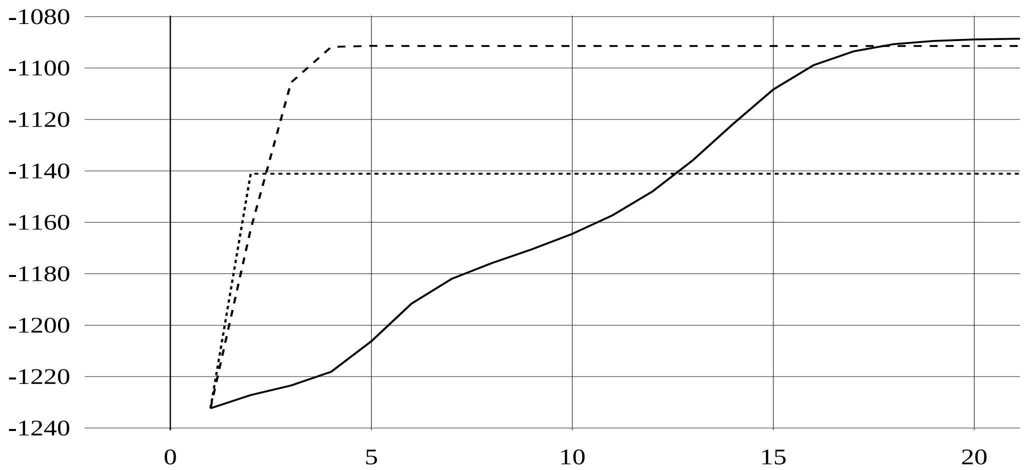

Figure 2: Convergence rates of the algorithm using exponentially decayed statistics with $\gamma = 0.99$ (dashed line) and $\gamma = 0.95$ (dotted line). For comparison, the performance of the incremental algorithm (solid line) is reproduced as well (as in Figure 1).

pure incremental algorithm, while the time required for each pass is only about 10% greater than for the standard algorithm, producing a significant net gain in speed.

The algorithm of iteration (9) was also tested. The same initialization procedure was used, with the elaboration that the decayed statistics were computed, but not used, during the initial standard iteration, in order to initialize them for use in later iterations.

Two runs of this algorithm are shown in Figure 2, done with $\gamma = 0.99$ and with $\gamma = 0.95$. Also shown is the run of the incremental algorithm (as in Figure 1). The run with $\gamma = 0.99$ converged to a good (but not optimal) point more rapidly than the incremental algorithm, but the run with $\gamma = 0.95$ converged to a rather poor point. These results indicate that there may be scope for improved algorithms that combine such fast convergence with the guarantees of stability and convergence to a true maximum that the incremental algorithm provides.

5 Other variants

The incremental algorithm is not the only variant of EM that can be justified by viewing it in terms of maximizing F. One could, for example, employ any of a wide variety of standard optimization methods to find the maximum of $F(\tilde{P}, \theta)$ with respect to $\tilde{P}$ and θ jointly. It is perhaps more interesting to look for further variants that, like the incremental algorithm, exploit the special nature of the problem.

A "sparse" variant of the algorithm may be advantageous when the unobserved variable, Y, can take on many possible values, but only a small set of "plausible" values have significant probability (given the observed data and the current parameter estimate). Substantial computation may sometimes be saved in this case by "freezing" the probabilities of the implausible values for many iterations, re-computing only the relative probabilities of the plausible values. At infrequent intervals, the probabilities for all values are recomputed, and a new set of plausible values selected (which may differ from the old set due to the intervening change in the parameter estimate). This procedure can be designed so that F is guaranteed to increase with every iteration, ensuring stability, even though some iterations may decrease L.

In detail, the sparse algorithm represents $\tilde{P}^{(t)}$ as follows:

$$\tilde{P}^{(t)}(y) \;=\; \begin{cases} q_y^{(t)} & \text{if } y \notin S^{(t)} \\ Q^{(t)} r_y^{(t)} & \text{if } y \in S^{(t)} \end{cases} \tag{12}$$

Here, $S^{(t)}$ is the set of plausible values for Y, the $q_y^{(t)}$ are the frozen probabilities for implausible values, $Q^{(t)}$ is the frozen total probability for plausible values, and the $r_y^{(t)}$ are the relative probabilities for the plausible values, which are updated every iteration.

Most iterations of the sparse algorithm go as follows:

$$\left.\begin{array}{ll} \textbf{E Step:} & \text{Set } S^{(t)} = S^{(t-1)},\ Q^{(t)} = Q^{(t-1)}, \text{ and } q_y^{(t)} = q_y^{(t-1)} \text{ for } y \notin S^{(t)}. \\ & \text{(This needn't take any time.)} \\ & \text{For all } y \in S^{(t)}, \text{ set } r_y^{(t)} = P(y \,|\, z, \theta^{(t-1)}) \,/\, P(y \in S^{(t)} \,|\, z, \theta^{(t-1)}). \\ \textbf{M Step:} & \text{Set } \theta^{(t)} \text{ to that } \theta \text{ which maximizes } F(\tilde{P}^{(t)}, \theta). \end{array}\right\} \tag{13}$$

It can easily be shown that the above E step selects those $r_y^{(t)}$ which maximize $F(\tilde{P}^{(t)}, \theta^{(t-1)})$. For suitable models, this restricted E step will take time proportional only to the size of $S^{(t)}$, independent of how many values are in the full range for Y. For the method to be useful, the model must also be such that the M step above can be done efficiently, as is discussed below.

On occasion, the sparse algorithm performs a full iteration, as follows:

$$\left.\begin{array}{ll} \textbf{E Step:} & \text{Set } S^{(t)} \text{ to those } y \text{ for which } P(y \mid z,\ \theta^{(t-1)}) \text{ is significant.} \\ & \text{For all } y \notin S^{(t)}, \text{ set } q_y^{(t)} = P(y \,|\, z,\ \theta^{(t-1)}). \\ & \text{Set } Q^{(t)} = P(y \in S^{(t)} \mid z,\ \theta^{(t-1)}). \\ & \text{For all } y \in S^{(t)}, \text{ set } r_y^{(t)} = P(y \mid z,\ \theta^{(t-1)}) \,/\, Q^{(t)}. \\ \textbf{M Step:} & \text{Set } \theta^{(t)} \text{ to that } \theta \text{ which maximizes } F(\tilde{P}^{(t)}, \theta). \end{array}\right\} \tag{14}$$

The decisions as to which values have "significant" probability can be made using various heuristics. One could take the N most probable values, for some predetermined N, or one could take as many values as are needed to account for some predetermined fraction of the total probability. The choice made will affect only the speed of convergence, not the stability of the algorithm — even with a bad choice for $S^{(t)}$, subsequent iterations cannot decrease F.

For problems with independent observations, where $Y = (Y_1, \ldots, Y_n)$, each data item, i, can be treated independently, with a separate set of "plausible" values, $S_i^{(t)}$, and with distributions $P_i^{(t)}$ expressed in terms of quantities $q_{i,y}^{(t)}$,

$Q_i^{(t)}$, and $r_{i,y}^{(t)}$. For efficient implementation of the M step in (13), it is probably necessary for the model to have simple sufficient statistics. The contribution to these of values with frozen probabilities can then be computed once when the full iteration of (14) is performed, and saved for use in the M step of (13), in combination with the statistics for the plausible values in $S_i^{(t)}$.

The Gaussian mixture problem provides an example of the potential usefulness of the sparse algorithm. If there are many components in the mixture, each data point will typically have a significant probability of having come from only a few components whose means are nearby. Freezing the small probabilities for the distant components avoids the continual re-computation of quantities that have negligible effect on the course of the algorithm.

Note that the sparse and incremental variants of EM can easily be applied in combination.

A "winner-take-all" variant of the EM algorithm may be obtained by constraining the distribution $\tilde{P}$ to assign zero probability to all but one value. Such a distribution can, of course, be represented by the single value that is assigned probability one. Obviously, such a variant of the algorithm cannot, in general, converge to the unconstrained maximum of $F(\tilde{P}, \theta)$, and hence need not find a value of θ that maximizes L. There might, however, be computational advantages to using this variant in the early stages of maximizing F, switching to a variant capable of finding the true maximum only when the winner-take-all variant has converged.

The well-known "K-means" clustering algorithm can be seen in this light as an incremental, winner-take-all version of the EM algorithm as applied to the Gaussian mixture problem (with variances and mixing proportions fixed). The winner-take-all method is also often used in estimating Hidden Markov Models for speech recognition. In neither instance is L guaranteed to increase with each iteration, which might lead one to regard these methods as completely *ad hoc*, but they appear more sensible when seen in terms of maximizing F, even though they don't find the unconstrained maximum.

Acknowledgements

This work was supported by the Natural Sciences and Engineering Research Council of Canada and by the Ontario Information Technology Research Centre. Geoffrey Hinton is the Noranda fellow of the Canadian Institute for Advanced Research.

References

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977) Maximum likelihood from incomplete data via the EM algorithm (with discussion), *Journal of the Royal Statistical Society B*, vol. 39, pp. 1-38.

Meng, X. L. and Rubin, D. B. (1992) "Recent extensions of the EM algorithm (with discussion)", in J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith (editors), *Bayesian Statistics 4*, Oxford: Clarendon Press.

Nowlan, S. J. (1991) *Soft Competitive Adaptation: Neural Network Learning Algorithms based on Fitting Statistical Mixtures*, Ph. D. thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh.