Expectation propagation for infinite mixtures

(Extended abstract)

Thomas Minka and Zoubin Ghahramani

December 17, 2003

Abstract

This note describes a method for approximate inference in infinite models that uses deterministic Expectation Propagation instead of Monte Carlo. For infinite Gaussian mixtures, the algorithm provides cluster parameter estimates, cluster memberships, and model evidence. Model parameters, such as the expected size of the mixture, can be efficiently tuned via EM with EP as the E-step. The same approach can apply other infinite models such as infinite HMMs.

1 Introduction

Consider a mixture model with an infinite number of components. From data we can obtain a posterior on the parameters of the components, but since this is infinite-dimensional we need a sensible way to summarize it. Define θ_i to be the parameters of the mixture component which generated point x_i. Points which came from the same component will have the same θ_i. The likelihood for $\boldsymbol{\theta}$ is thus

$$p(D|\boldsymbol{\theta}) = \prod_i p(x_i|\theta_i) \tag{1}$$

If the mixture weights come from a Dirichlet process with parameter α, then the prior probability for the θ_i is given by the recursion

$$p(\theta_i|\theta_{<i}) = \frac{\alpha}{i-1+\alpha}p(\theta_i) + \frac{1}{i-1+\alpha}\sum_{j<i}\delta(\theta_i - \theta_j) \tag{2}$$

These equations summarize the infinite mixture model, in a form which is amenable to Expectation Propagation. The strategy will be to approximate the posterior over $\boldsymbol{\theta}$ by a simple factorized distribution:

$$q(\boldsymbol{\theta}) = \prod_i q(\theta_i) \tag{3}$$

In this note, the factors will be Gaussian:

$$q(\theta_i) \sim \mathcal{N}(\mathbf{m}_i, \mathbf{V}_i) \tag{4}$$

2 ADF equations

We want the probability of a particular dataset $D = \{x_1, ..., x_n\}$, given by an integral over $\boldsymbol{\theta}$:

$$p(D) = \int_{\boldsymbol{\theta}} p(D|\boldsymbol{\theta}) \prod_i p(\theta_i|\theta_{<i}) d\boldsymbol{\theta} \tag{5}$$

There are two types of terms in the integrand: likelihood terms and prior terms. EP will iterate across the terms, approximating them one by one. Let's start with the prior terms, denoted $f_i(\boldsymbol{\theta}) = p(\theta_i|\theta_{j<i})$. Each of these, as a function of $\boldsymbol{\theta}$, will be approximated by a factorized function:

$$\tilde{f}_i(\boldsymbol{\theta}) = \prod_{j \le i} \tilde{f}_{ij}(\theta_j) \tag{6}$$

The factors are interpreted as 'messages', and f_i only sends messages to $j \le i$. Because the approximate posterior is disconnected and normal, we must have:

$$\tilde{f}_{ij}(\theta_j) = s_{ij} \exp(-\frac{1}{2}(\theta_j - \mathbf{m}_{ij})^{\mathrm{T}} \mathbf{V}_{ij}^{-1} (\theta_j - \mathbf{m}_{ij})) \tag{7}$$

$$\mathbf{V}_i = (\sum_{j \le i} \mathbf{V}_{ji}^{-1})^{-1} \tag{8}$$

$$\mathbf{m}_i = \mathbf{V}_i \sum_{j \le i} \mathbf{V}_{ji}^{-1} \mathbf{m}_{ji} \tag{9}$$

$$q^{\backslash i}(\theta_j) = q(\theta_j)/\tilde{f}_{ij}(\theta_j) \sim s_j^{\backslash i} \mathcal{N}(\mathbf{m}_j^{\backslash i}, \mathbf{V}_j^{\backslash i}) \tag{10}$$

$$\mathbf{V}_j^{\backslash i} = (\mathbf{V}_j^{-1} - \mathbf{V}_{ij}^{-1})^{-1} \tag{11}$$

$$\mathbf{m}_j^{\backslash i} = \mathbf{V}_j^{\backslash i} (\mathbf{V}_j^{-1} \mathbf{m}_j - \mathbf{V}_{ij}^{-1} \mathbf{m}_{ij}) \tag{12}$$

$$s_j^{\backslash i} = s_{ij}^{-1} \frac{\left|\mathbf{V}_j^{\backslash i}\right|}{|\mathbf{V}_j|} \exp(-\frac{1}{2}(\mathbf{m}_j - \mathbf{m}_j^{\backslash i})^{\mathrm{T}} (\mathbf{V}_j - \mathbf{V}_j^{\backslash i})^{-1} (\mathbf{m}_j - \mathbf{m}_j^{\backslash i})) \tag{13}$$

For each term, we form the product $f_i(\boldsymbol{\theta}) q^{\backslash i}(\boldsymbol{\theta})$ and find its moments. First the integral:

$$Z_i = \int_{\boldsymbol{\theta}} f_i(\boldsymbol{\theta}) q^{\backslash i}(\boldsymbol{\theta}) d\boldsymbol{\theta} = \frac{\alpha}{i - 1 + \alpha} Z_{ii} + \frac{1}{i - 1 + \alpha} \sum_{j<i} Z_{ji} \tag{14}$$

$$\text{where } Z_{ii} = \int_{\theta} p(\theta) q^{\backslash i}(\theta_i = \theta) d\theta \tag{15}$$

$$Z_{ji} = \int_{\theta} q^{\backslash i}(\theta_j = \theta) q^{\backslash i}(\theta_i = \theta) d\theta \tag{16}$$

It is assumed the latter integrals can be solved analytically. Now the first moments:

$$\mathbf{m}_i = \frac{1}{Z_i}\int_{\boldsymbol{\theta}} \theta_i f_i(\boldsymbol{\theta}) q^{\backslash i}(\boldsymbol{\theta}) d\boldsymbol{\theta} = \sum_{j \le i} r_{ji}\hat{\theta}_{ji} \tag{17}$$

$$\text{where } \hat{\theta}_{ii} = \frac{1}{Z_{ii}}\int_{\theta} \theta p(\theta) q^{\backslash i}(\theta_i = \theta) d\theta \tag{18}$$

$$\hat{\theta}_{ji} = \frac{1}{Z_{ji}}\int_{\theta} \theta q^{\backslash i}(\theta_j = \theta) q^{\backslash i}(\theta_i = \theta) d\theta \tag{19}$$

$$r_{ii} = \frac{\alpha}{i-1+\alpha}\frac{Z_{ii}}{Z_i} \tag{20}$$

$$r_{ji} = \frac{1}{i-1+\alpha}\frac{Z_{ji}}{Z_i} \tag{21}$$

$$r_{ii} + \sum_{j<i} r_{ji} = 1 \tag{22}$$

$$\mathbf{m}_j = \frac{1}{Z_i}\int_{\boldsymbol{\theta}} \theta_j f_i(\boldsymbol{\theta}) q^{\backslash i}(\boldsymbol{\theta}) d\boldsymbol{\theta} = (1 - r_{ji})\mathbf{m}_j^{\backslash i} + r_{ji}\hat{\theta}_{ji} \tag{23}$$

The number r_{ji} can be interpreted as a soft assignment of point i to the cluster of point j, and the expectation is a sum over possible assignments. Now the second moments:

$$\mathbf{V}_i + \mathbf{m}_i^2 = \frac{1}{Z_i}\int_{\boldsymbol{\theta}} \theta_i^2 f_i(\boldsymbol{\theta}) q^{\backslash i}(\boldsymbol{\theta}) d\boldsymbol{\theta} = \sum_{j \le i} r_{ji}\breve{\theta}_{ji} \tag{24}$$

$$\text{where } \breve{\theta}_{ii} = \frac{1}{Z_{ii}}\int_{\theta} \theta^2 p(\theta) q^{\backslash i}(\theta_i = \theta) d\theta \tag{25}$$

$$\breve{\theta}_{ji} = \frac{1}{Z_{ji}}\int_{\theta} \theta^2 q^{\backslash i}(\theta_j = \theta) q^{\backslash i}(\theta_i = \theta) d\theta \tag{26}$$

$$\mathbf{V}_j + \mathbf{m}_j^2 = \frac{1}{Z_i}\int_{\boldsymbol{\theta}} \theta_j^2 f_i(\boldsymbol{\theta}) q^{\backslash i}(\boldsymbol{\theta}) d\boldsymbol{\theta} \tag{27}$$

$$= (1 - r_{ji})(\mathbf{V}_j^{\backslash i} + (\mathbf{m}_j^{\backslash i})^2) + r_{ji}\breve{\theta}_{ji} \tag{28}$$

$$\mathbf{V}_j = (1 - r_{ji})\mathbf{V}_j^{\backslash i} + r_{ji}(\breve{\theta}_{ji} - \hat{\theta}_{ji}^2) + r_{ji}(1 - r_{ji})(\hat{\theta}_{ji} - \mathbf{m}_j^{\backslash i})^2 \tag{29}$$

To make this concrete, consider the case of Gaussian components with fixed variance:

$$p(x_i|\theta_i) \sim \mathcal{N}(\theta_i, \Sigma) \tag{30}$$

$$p(\theta) \sim \mathcal{N}(\mathbf{m}_0, \mathbf{V}_0) \tag{31}$$

$$p(\theta) q^{\backslash i}(\theta_i = \theta) = \mathcal{N}(\theta; \hat{\theta}_{ii}, (\mathbf{V}_0^{-1} + (\mathbf{V}_i^{\backslash i})^{-1})^{-1}) \mathcal{N}(\mathbf{m}_0; \mathbf{m}_i^{\backslash i}, \mathbf{V}_0 + \mathbf{V}_i^{\backslash i}) \tag{32}$$

$$\hat{\theta}_{ii} = (\mathbf{V}_0^{-1} + (\mathbf{V}_i^{\backslash i})^{-1})^{-1}(\mathbf{V}_0^{-1}\mathbf{m}_0 + (\mathbf{V}_i^{\backslash i})^{-1}\mathbf{m}_i^{\backslash i}) \tag{33}$$

$$q^{\backslash i}(\theta_j = \theta) q^{\backslash i}(\theta_i = \theta) = \mathcal{N}(\theta; \hat{\theta}_{ji}, ((\mathbf{V}_j^{\backslash i})^{-1} + (\mathbf{V}_i^{\backslash i})^{-1})^{-1}) \mathcal{N}(\mathbf{m}_j^{\backslash i}; \mathbf{m}_i^{\backslash i}, \mathbf{V}_j^{\backslash i} + \mathbf{V}_i^{\backslash i}) \tag{34}$$

$$\hat{\theta}_{ji} = ((\mathbf{V}_j^{\backslash i})^{-1} + (\mathbf{V}_i^{\backslash i})^{-1})^{-1}((\mathbf{V}_j^{\backslash i})^{-1}\mathbf{m}_j^{\backslash i} + (\mathbf{V}_i^{\backslash i})^{-1}\mathbf{m}_i^{\backslash i}) \tag{35}$$

From this you can read off Z, $\hat{\theta}$, and $\breve{\theta}$.

3 EP

The EP algorithm is:

1. Initialize each $q(\theta_i)$ with the likelihood term $p(x_i|\theta_i)$, which does not need to be approximated.
2. Until all $\tilde{f}_i$ converge, loop $i = 1, ..., n$:
   (a) *Deletion.* Remove $\tilde{f}_i$ from q to get the 'old' approximation $q^{\setminus i}$ (11,12).
   (b) *Incorporate evidence.* Compute the moments of $f_i(\boldsymbol{\theta})q^{\setminus i}(\boldsymbol{\theta})$ to get a new $q(\boldsymbol{\theta})$.
   (c) *Update.* Re-estimate $\tilde{f}_i$ by division (apply (11,12) in reverse).

The input to the algorithm is $(\alpha, \Sigma, \mathbf{m}_0, \mathbf{V}_0)$ and the data. The output is $(\mathbf{m}_i, \mathbf{V}_i)$ and a soft assignment matrix r_{ji}. From the assignment matrix we can estimate the probability that two points are from the same component, and the expected number of components. For the former, apply dynamic programming. For the latter, just take $\hat{k} = \sum_i r_{ii}$. The prior expected number of components is $\alpha(\Psi(\alpha + n) - \Psi(\alpha))$, so setting this equal to $\hat{k}$ gives an update rule for α. This is equivalent to the M-step in an EM algorithm for α, where the E-step is handled by EP.

During the deletion step, the covariance $\mathbf{V}_j^{\setminus i}$ may turn out not positive definite. In this case, it is sufficient to skip term i for that iteration of EP.

The cost of this algorithm is $O(d^3n^2)$ per iteration.

4 Example

Figure 1(a) plots 8 points in two dimensions, generated by sampling from two Gaussians, $\mathcal{N}([-1.5\ -1.5], \mathbf{I})$ and $\mathcal{N}([1.5\ 1.5], \mathbf{I})$. Thus the true θ_i take on two distinct values. The model parameters were set to $\mathbf{m}_0 = [0\ 0]$, $\mathbf{V}_0 = 25\mathbf{I}$, and $\Sigma = \mathbf{I}$. Figure 1(b) plots $E[\theta_i|D]$, for each i, computed in two ways: approximately by EP and exactly by expanding the prior into $n!$ terms. Relative to the exact means, the EP means are pulled inward, implying a bias toward fewer clusters. α was estimated to be 0.38, so that the expected number of components was 1.82. The exact $\log p(D) = -33.25$, and the estimate from EP was -33.15.

The probabilities in figure 1(c) strongly suggest that points 1–4 came from one component and points 5–8 came from another, which is indeed how the data were generated.

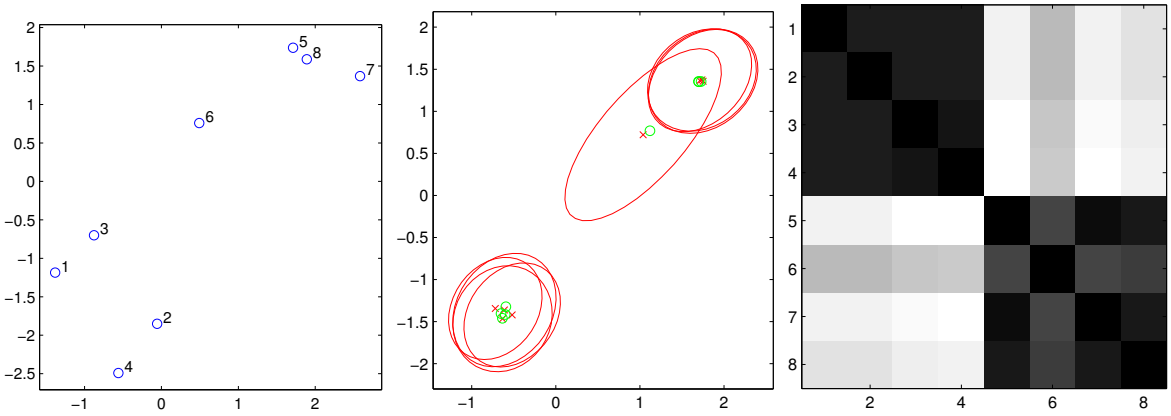

Figure 1: (a) A sample from two Gaussians. (b) The EP approximate posterior for each θ_i is plotted as a mean 'x' and 1-standard deviation ellipse. Compare to the exact posterior means 'o'. (c) The estimated probabilities that two points came from the same Gaussian (darker means higher probability).

5 Ordering

The model defined by (2) is independent of the order of points. However, the approximation computed by EP, because of the factorization assumption, does depend on the order. Thus the ordering of points is a parameter of the approximation algorithm.

To demonstrate the dependence on order, 100 random orderings were tried on the above dataset. Figure 2 plots the root-mean-squared error in the means $E[\theta_i|D]$ versus (a) the estimated evidence and (b) the total flops, for each ordering. There appears to be no connection with evidence, but some connection with flops. The best orderings tend to be ones where EP converges quickly.

Inspection of the best orderings shows that they are anti-correlated—they put nearby points far apart in the ordering. For example, in the above plot the best ordering was (3,4,7,5,6,1,2,8), which has error

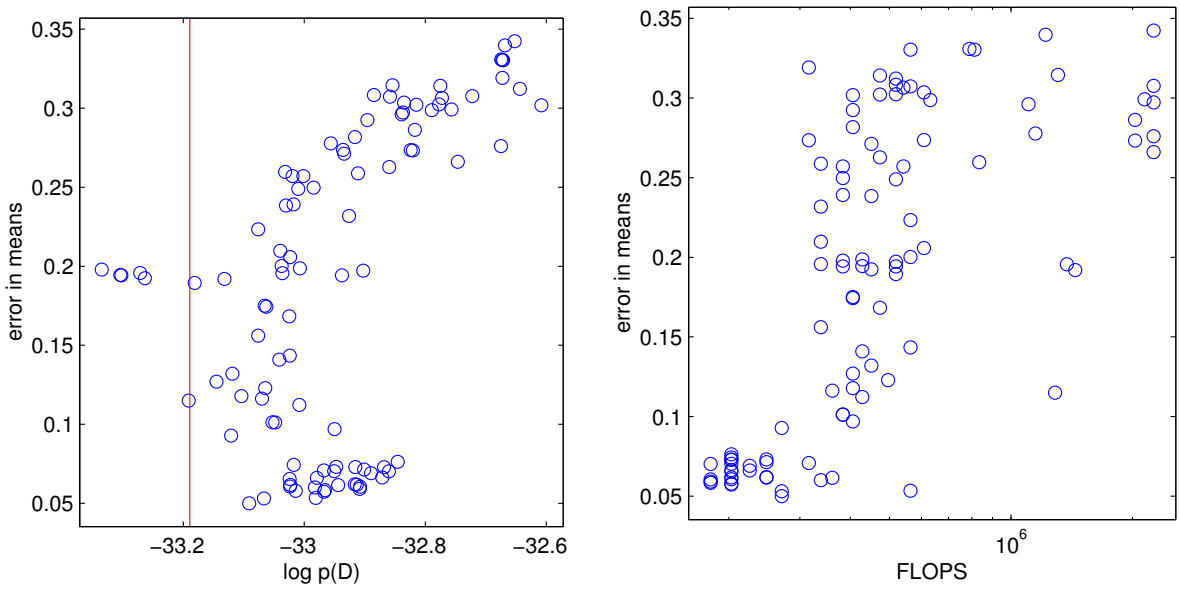

Figure 2: Accuracy of EP over 100 random permutations of the data. The red line in (a) marks the exact value of $\log p(D)$.

0.05. This makes sense because of the factorization assumption—points which are distant will have nearly independent θ_i's. Another way to say it is that we want r_{ii} to be as big as possible for the first few points.

A heuristic algorithm that works well is to pick points one at a time, always picking the point which is furthest away from previously picked points, but close to an unpicked point. This can be viewed as a crude version of Mettu & Plaxton (2002). This rule produces the ordering (5,4,3,7,6,2,1,8), which is the order used to produce the plots in section 4, and has error 0.04.

Another approach is to throw out (2) and reformulate the prior in an order-independent way. To date, we have not found a way which retains the $O(n^2)$ complexity of the algorithm. It is tempting to simply redefine (2) to depend on all other θ_j, not just $j < i$, but that does not define a proper probability model.

References

Mettu, R. R., & Plaxton, G. C. (2002). Optimal time bounds for approximate clustering. *UAI*.