Algorithms in Computational Biology

Extensions of the EM Algorithm

Ela Fallik

The **Expectation Maximization (EM) algorithm** is an intuitive, widely used algorithm. It was outlined and given its name by Dempster, Laird and Rubin in a paper published in 1977 [1]. Since then, an entire field of expansions and extensions to the classical EM algorithm has evolved, following some challenges in implementation for complex problems. Here, we present some of the challenges, show some examples of extensions (focusing on the Expectation step of the algorithm), and explore how they help address these challenges.

Remark 1. This scibe relies heavlly on the introduction of "A New Class of Stochastic EM Algorithms. Escaping Local Maxima and Handling Intractable Sampling" by Stéphanie Allassonnière and Juliette Chevallier [2].

The EM Algorithm

The EM algorithm is an iterative algorithm for Maximum (Log)-Likelihood estimation, in situations with incomplete data or unobserved latent variables. In those type of problems, it is intractable to maximize the log-likelihood

$$l(\theta; X) = \log p(X|\theta)$$

directly, but given some additional data Z, it's possible (and usually considered easy) to maximize

$$l(\theta; X, Z) = \log p(X, Z|\theta)$$

Algorithm: Given a starting point θ^0, the algorithm include two steps, which are applied iteratively until convergence of the incomplete-data (marginal) log-likelihood:

1. Expectation (E)-step: Calculate the expected complete-data log-likelihood with the current parameters

$$Q\left(\theta|\theta^t\right) = \mathbb{E}_{Z|X,\theta^t}\left[l(\theta; X, Z)\right]$$

2. Maximization (M)-step: Find the parameters that maximize the expectation

$$\theta^{t+1} = \arg\max_{\theta} Q\left(\theta|\theta^t\right)$$

Properties

First, it is important to notice how intuitive is this algorithm: We don't have all the data, and so we can't maximize according to the complete-data log-likelihood to get the MLE. Instead, we choose some starting point for the parameters, calculate what complete-data log-likelihood we expect if the guess was correct, and maximize according to that. Iterating over those steps, we expect to get closer to the actual complete-data log-likelihood, and therefore to the parameters we're after.

This intuition is supported by the following facts:

Proposition 1. If $\theta^0, \theta^1, ..., \theta^t, ...$ is the output of the EM algorithm, then

$$\forall_t, l\left(\theta^t; X\right) \geq l\left(\theta^{t-1}; X\right)$$

Additionally, as $\forall_t, l\left(\theta^t; X\right) \leq l\left(\theta^{ML}; X\right)$ by definition (assuming it exists), we get that $\{l\left(\theta^t; X\right)\}_t$ is a bounded non-decreasing sequence, and therefore converge to some l^*. Assuming some regularity conditions (which we will not specify here, but generally hold for most real life problems, and we'll assume hold for ours) we get

Proposition 2. $\lim_{t\to\infty} l\left(\theta^t; X\right) = l^*$ is a local maximum of the marginal log-likelihood $l\left(\theta; X\right)$.

For some cases it's enough, and we get a global convergence to the MLE (either directly or by using multiple random initializations\ a smart choice of starting point).

Another advantage of the EM algorithm is the simplicity of each step: Usually, each step is easy to implement, analytically and computationally.

Challenges

Although the EM algorithm is very popular, it introduces some challenges in complex problems:

Convergence to local maxima: As much as the convergence results we outlined above is encouraging, it's still not exactly what we want: For most problems, we only have an assurance of convergance to a local maximum, and not to the global maximum, i.e. the MLE. This is the situation in many real-life non-esoteric problems: We'll get a convergence to different local maxima, depending on our starting point.
This problem can sometimes be solved by multiple random initializations. However, in some cases, it may require many such initializations, and each run can be computationally heavy.

Computational challenges: Another setback can be the calculation in the E-step. In our setup, we assumed that the complete-data log-likelihood $l\left(\theta; X, Z\right)$ is easy to maximize, making the M-step straight forward. But what about the calculation of the expectation $Q\left(\theta|\theta^t\right)$? It may require a complicated calculation (e.g. dynamic programming), which introduce a computational challenge as it need to be done for each iteration of the EM.

Over-fit: The last challenge we would review is not so much the result of using the EM algorithm, but a general product of choosing the MLE as our estimator, when aiming at predictions. This estimator rely on maximizing the log-likelihood on a training set, meaning we're looking for

$$\hat{\theta}^{ML} = \arg\max_{\theta} \hat{p}_{\theta}$$

for the empirical distribution $\hat{p} = p(X|\cdot)$, whereas what we actually want is θ^*, the "true" parameters of the distribution p^* from which the data set was sampled. Therefore, it's likely we'll see an over-fit, meaning low generalization capabilities.

Addressing the Challenges

In the next section, we will explore some classic and widely used examples of EM extensions that address the presented challenges. All of them will introduce elements of randomness, incorporating a well-research method as a replacement to the E-step.
Following some assumptions, those replacements should decrease the computational cost of the E-step significantly. The randomness aspect will help, heuristically, escape local maxima and avoid over-fit. Although not a proven product, it is a common practice and considered a good practical tool to overcome these issues. Additionally, increased computational cost will allow multiple runs in order to find the best initialization.
As we go over the proposed extensions, we will include the logic in the basis of the methods behind them.

Extensions

Monte-Carlo Integration

Generally, in cases where an expectation calculation is intractable, a good approximation of it is the **sample mean**: Say we can draw independent samples $y_1, ..., y_N$ from a distribution p. Then we can approximate the expectation (w.r.t p) using the sample mean:

$$\mathbb{E}[f(Y)] \approx \frac{1}{N} \sum_{i=1}^{N} f(y_i)$$

This estimator is intuitive, efficient and unbiased, and thus is a very popular one. It is sometimes known as **Monte-Carlo Integration**.
In our case, p will be the posterior distribution of the unobserved\ missing data, $p = p(Z|X, \theta^t)$.

Stochastic EM (SEM)

This algorithm was first introduced by Celeux and Diebolt in 1985 [3]. In this version, we get a very narrow usage of the principle above - we only use one sample. The E-step thus becomes a simulation step:

1. Simulation (S)-step: Sample from the posterior of the unobserved\ missing data

$$Q\left(\theta|\theta^t\right) = l\left(\theta; X, z^t\right), z^t \sim p\left(Z|X;\theta^t\right)$$

Monte-Carlo EM (MCEM)

Similarly, we can use several i.i.d samples from the posterior (as many as we can afford computationally), and average the complete-data log-likelihood over them. This idea was formulated by Wei and Tanner in 1990 [4]:

1. Simulation (S)-step: Sample N_t samples from the posterior of the unobserved\ missing data, $z_1^t, ..., z_{N_t}^t \overset{\text{i.i.d}}{\sim} p\left(Z|X;\theta^t\right)$.

2. Monte-Carlo integration (MC)-step:

$$Q\left(\theta|\theta^t\right) = \frac{1}{N_t}\sum_{i=1}^{N_t} l\left(\theta; X, z_i^t\right)$$

Properties

There are obvious advantages to the MCEM algorithm. It's straightforward, computationally light, and simple to implement, assuming we can easily sample from the posterior. It also allows us to estimate the uncertainty of the calculation.

However, as an additional error is introduced by the MC integration, we can't count on proposition 1 anymore, and the monotonicity of the marginal log-likelihood isn't guaranteed. Nevertheless, in certain situations it's possible to prove some statements about the convergence of the MCEM (e.g. [5, 6, 7]). We will not dive into details here (can be found in this review [8]), but one interesting and practical remark is that **the sample size of the MC integration must increase over time**.

As explained in the review, it makes sense intuitively, as we often see the classical EM algorithm make a significant jump (in terms of the marginal log-likelihood) at the start of the run, when the parameters are far from the MLE. At this time, precision isn't extra important and a small sample size should suffice. However, as the run continues, the EM update becomes smaller and the approximation of it has to be more accurate. Therefore, it requires a much larger sample size.

Stochastic Approximation EM (SAEM)

Next, we can notice that in each iteration of the Stochastic and Monte Carlo versions of EM, we discard all the information we obtained in the previous iterations. This information can be valuable, especially if we can't afford to sample many samples from the posterior in each iteration. In the next version (proposed by Lavielle and Moulines in 1995 [11]), instead of calculating the expected complete-data log-likelihood, we sample (one or more samples) from the posterior, and average the log-likelihood (weighted by some step-size) with all the approximations from previous iterations:

1. Simulation (S)-step: Sample from the posterior of the unobserved\ missing data, $z^t \overset{\text{i.i.d}}{\sim} p\left(Z|X;\theta^t\right)$.
2. Stochastic approximation (SA)-step:

$$\begin{aligned} Q\left(\theta|\theta^t\right) &= Q\left(\theta|\theta^{t-1}\right) + \gamma^t \cdot \left[l\left(\theta;X,z^t\right) - Q\left(\theta|\theta^{t-1}\right)\right] \\ &= \left(1-\gamma^t\right) \cdot Q\left(\theta|\theta^{t-1}\right) + \gamma^t \cdot l\left(\theta;X,z^t\right) \end{aligned}$$

Properties

While this procedure enjoys the basic advantages of the MCEM algorithm, it also addresses the requirement for an increased sample size. It gives an increasingly accurate approximation of the expectations, relying on all the sampled unobserved\ missing data from the previous iterations, and not only the current one. Moreover, if we choose to take a decreasing step size, we get an approximation that relies more on the directions established in many previous iterations and less on the noisy Monte-Carlo approximation. This algorithm is guaranteed to converge to a local maximum under certain conditions [10].
However, with the reliance on previous iterations, the SAEM is heavily dependent on initialization, which increases the chance of convergence to local maxima instead of the MLE. A few proposed solutions exists to this problem, one of them being **Simulated Annealing SAEM** (Lavielle and Moulines, 1997 [12]). The idea is to present a "false" model in each iteration by adding noise to the incomplete-data log-likelihood. This flatten the function, which should preserve the global maximum while "hiding" the local ones. The noise levels are decreasing at each iteration untill reaching 0, in a precedure inspired by the annealing process of metals.

Markov Chain Monte-Carlo (MCMC) EM

In all the previous versions, we assumed we're able to sample from the posterior (once, or many i.i.d samples). In some situations, however, this is as hard as calculating the expected complete-data log-likelihood, and therefore irrelevant. One possible solution is to use the Markov Chain Monte-Carlo method, presented by Hastings in 1970 [13] (and in this useful review [14]):
Say our goal is to estimate the expectation $\mathbb{E}\left[f\left(Y\right)\right]$ for some r.v. Y. If we have a Markov chain with a stationary distribution $\pi = p$, then after a burn-in period of T_0 iterations we can expect to get $y^{T_0+1}, ..., y^{T_0+N}$ dependent samples sampled from $\approx p$ with the "correct" proportions. Therefore we can approximate the expectations using the **ergodic average**:

$$\mathbb{E}\left[f\left(Y\right)\right] \approx \frac{1}{N}\sum_{i=1}^{N} f\left(y^{T_0+i}\right)$$

Therefore, all we need is a Markov chain with the stationary distribution $\pi = p$. Before we specify an algorithm that achieves this goal, let's see how to use that process in our case:

For each iteration t, we wish to estimate the expectation

$$\mathbb{E}_{Z|X,\theta^t}\left[l\left(\theta; X, Z\right)\right]$$

and thus we're looking for a Markov chain with the posterior $p = p\left(Z|X, \theta^t\right)$ of the unobserved\ missing data as a stationary distribution.

Metropolis Hastings Algorithm:

We wish to design a Markov chain that converge to a unique stationary distribution $\pi = p$. We would assume that the ratios $\frac{p\left(y'\right)}{p\left(y\right)}$ can be calculated quickly.

Markov chains: A Markov chain is defined by its states S and transition probabilities $\left\{p\left(y'|y\right)\right\}_{y,y'\in S}$.
For it to have a unique stationary distribution, it's sufficient to require two conditions:

1. **Detailed balance**:

$$\pi\left(y\right)p\left(y'|y\right) = \pi\left(y'\right)p\left(y|y'\right)$$

2. **Ergodicity:** p is a positive distribution.

As we want this stationary distribution to be $\pi = p$, the detailed balance conditions becomes:

$$\frac{p\left(y'\right)}{p\left(y\right)} = \frac{p\left(y'|y\right)}{p\left(y|y'\right)} \tag{1}$$

Transition probabilities: The idea is to divide each step of the chain into two parts:

1. Proposal: $g\left(y'|y\right)$ = proposal distribution, the probability to propose y' given we're now at y.
2. Acceptance: $A\left(y',y\right)$ = acceptance distribution, the probability to accept the proposed y' given y.

So we get

$$p\left(y'|y\right)=g\left(y'|y\right)\cdot A\left(y',y\right)$$

and condition (1) becomes

$$\frac{p\left(y'\right)}{p\left(y\right)}=\frac{p\left(y'|y\right)}{p\left(y|y'\right)}=\frac{g\left(y'|y\right)\cdot A\left(y',y\right)}{g\left(y|y'\right)\cdot A\left(y,y'\right)}\iff\frac{A\left(y',y\right)}{A\left(y,y'\right)}=\frac{p\left(y'\right)\cdot g\left(y|y'\right)}{p\left(y\right)\cdot g\left(y'|y\right)}$$

The choice of g is very much problem dependent. Given such g, we can choose A which will depend on g, and only need to check that it uphold the condition above. As we're only looking at the ratio here, we can take A to be only proportional to a distribution. We'll take

$$A\left(y',y\right)=\min\left\{1,\frac{p\left(y'\right)\cdot g\left(y|y'\right)}{p\left(y\right)\cdot g\left(y'|y\right)}\right\}$$

If we set $r=\frac{p\left(y'\right)\cdot g\left(y|y'\right)}{p\left(y\right)\cdot g\left(y'|y\right)}$, we can notice that

$$\text{if } r\geq 1\Rightarrow A\left(y',y\right)=1,A\left(y,y'\right)=r$$
$$\text{if } r<1\Rightarrow A\left(y',y\right)=r,A\left(y,y'\right)=1$$

and thus we always get $\frac{A\left(y',y\right)}{A\left(y,y'\right)}=r$, as require.

Algorithm: Overall we get the following algorithm:

1. Init: y^0 initial state.
2. Iteration $1\leq i\leq T_0+N$:
   (a) Proposal: sample $y'\sim g\left(\cdot|y^{i-1}\right)$.
   (b) Acceptance: calculate $a=A\left(y',y^{i-1}\right)$.
      i. With prob. a, accept: $y^i=y'$.
      ii. With prob. $1-a$, reject: $y^i=y^{i-1}$.
3. Return: $y^{T_0+1},...,y^{T_0+N}$.

Properties

This algorithm differs from the MCEM in its sampling and integration methods.

As the convergence of the posterior probability stems from the Markov chain construction, we theoretically (in some sense of limit) get an integration method equivalent to the Monte-Carlo integration, meaning converge to the desired expectation. This theoretical guarantee is a result of the **ergodic theorem**. However, we cannot actually run the chain infinitely, and thus this sampling method bound to introduce some level of error to the approximation of the expectation. There are multiple suggestions for estimation of convergence and error rates [14].

There are many additional issues to address concerning the sampling method itself: the choice of starting point, burn-in period length, conversion (stop time), etc. All of those are highly researched questions and have practical and theoretical solutions [14].

Unlike the algorithms which we mentioned above, for this algorithm, I couldn't find any general guarantees for convergence. However, as MCMC is considered a useful method for estimation of expectation, it is commonly used as a substitute for the E-step in the EM algorithm and provides good results in many cases.

Conclusion

In this scribe, we reviewed some of the challenges that arise from the use of the EM algorithm, namely local maxima, computational challenges, and over-fit. We discussed how stochastic replacements of the E-step help overcome those challenges and saw some popular examples: SEM, MCEM, SAEM, and MCMC-EM. Those are obviously only the tip of the iceberg, and many more extensions exist.

References

[1] Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society: Series B (Methodological), 39(1), 1-22.

[2] Stéphanie Allassonnière, Juliette Chevallier. A New Class of Stochastic EM Algorithms. Escaping Local Maxima and Handling Intractable Sampling. Computational Statistics and Data Analysis, Elsevier, 2021, 159, pp.107159.

[3] Celeux, G. (1985). The SEM algorithm: a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. Computational statistics quarterly, 2, 73-82.

[4] Wei, G. C., & Tanner, M. A. (1990). A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. Journal of the American statistical Association, 85(411), 699-704.

[5] Chan, K. S., & Ledolter, J. (1995). Monte Carlo EM estimation for time series models involving counts. Journal of the American Statistical Association, 90(429), 242-252.

[6] Fort, G., & Moulines, E. (2003). Convergence of the Monte Carlo expectation maximization for curved exponential families. Annals of Statistics, 31(4), 1220-1259.

[7] Booth, J. G., & Hobert, J. P. (1999). Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 61(1), 265-285.

[8] Neath, R. C. (2013). On convergence properties of the Monte Carlo EM algorithm. In Advances in modern statistical theory and applications: a Festschrift in Honor of Morris L. Eaton (pp. 43-62). Institute of Mathematical Statistics.

[9] Lavielle, M. and Moulines, E. (1995). On a stochastic approximation version of the EM algorithm. Technical Report Publication Universite Paris-Sud.

[10] Delyon, B., Lavielle, M., & Moulines, E. (1999). Convergence of a stochastic approximation version of the EM algorithm. The Annals of Statistics, 27(1), 94-128.

[11] Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. science, 220(4598), 671-680.

[12] Lavielle, M., & Moulines, E. (1997). A simulated annealing version of the EM algorithm for non-Gaussian deconvolution. Statistics and Computing, 7(4), 229-236.

[13] Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications.

[14] Gilks, W. R., Richardson, S., & Spiegelhalter, D. J. (1995). Introducing Markov chain Monte. Markov chain Monte Carlo in practice, 1.