The EM Algorithm: A Guided Tour

Christophe Couvreur

Service de Physique Générale, Faculté Polytechnique de Mons
Rue de Houdain 9, B-7000 Mons, Belgium
couvreur@thor.fpms.ac.be

Abstract

The Expectation-Maximization (EM) algorithm has become one of the methods of choice for maximum-likelihood (ML) estimation. In this tutorial paper, the basic principles of the algorithm are described in an informal fashion and illustrated on a notional example. Various applications to real-world problems are briefly presented. We also provide selected entry points to the vast literature on the EM algorithm for the reader interested in a rigorous mathematical treatment and further details on the applications. We discuss the convergence properties of the algorithm and review some variants and improvements that have been proposed. We conclude by some practical advice for the practicing engineer interested in the implementation of the EM algorithm.

1. Introduction

Because of its asymptotic optimal properties, maximum-likelihood (ML) has become one of the preferred methods of estimation in many areas of application of statistics, including system identification, speech and image processing, communication, computer tomography, pattern recognition, and many others. Often, however, no theoretical solution of the likelihood equations is available and it is necessary to resort to numerical optimization techniques. Direct maximization of the likelihood function by standard numerical optimization methods such as Newton-Raphson or gradient (scoring) methods is possible, but generally requires heavy analytical preparatory work to obtain the gradient (and, possibly, the Hessian) of the likelihood function. Moreover, the implementation of these methods may present numerical difficulties (memory requirements, convergence, instabilities, ...), particularly when the number of parameters to be estimated is high (the dreaded "curse of dimensionality"). For a certain class of statistical problems, an alternative to the direct numerical maximization of the likelihood was introduced in 1977 by Dempster, Laird, and Rubin: the *Expectation-Maximization* or EM algorithm [1]. The EM algorithm is a general method for maximum-likelihood estimation for so-called "incomplete data" problems. Since its inception it has been used successfully in a wide variety of applications ranging from mixture density estimation to system identification and from speech processing to computer tomography.

The remainder of the paper is organized as follows. In Section 2, incomplete data problems are defined and the EM algorithm for their solution is presented. A notional example illustrates how the algorithm can be put to use. In Section 3, arguments motivating the choice of the EM algorithm for a ML problem are discussed and examples of practical applications of the EM algorithm are briefly presented. The convergence properties of the algorithm are the subject of Section 4. Some variants of the EM algorithm are reviewed in Section 5. We conclude by a summary of the advantages and disadvantages of the EM algorithms when compared to other likelihood maximization methods.

2. The EM Algorithm

2.1. Incomplete Data Problems

Let $\mathcal{X}$ and $\mathcal{Y}$ be two sample spaces, and let H be a many-to-one transformation from $\mathcal{X}$ to $\mathcal{Y}$. Let us assume that the observed random variable $\mathbf{y}$ in $\mathcal{Y}$ is related to an unobserved random variable $\mathbf{x}$ by $\mathbf{y} = H(\mathbf{x})$. That is, there is some "complete" data $\mathbf{x}$ which is only partially observed in the form of the "incomplete data" $\mathbf{y}$. Let $p(\mathbf{x}|\theta)$ be the parametric distribution of $\mathbf{x}$, where θ is a vector of parameters taking its values in Θ. The distribution of $\mathbf{y}$, denoted by $q(\mathbf{y}|\theta)$, is also parameterized by θ since

$$q(\mathbf{y}|\theta) = \int_{H(\mathbf{x})=\mathbf{y}} p(\mathbf{x}|\theta)\, d\mathbf{x}. \qquad (1)$$

Estimation of θ from $\mathbf{y}$ is an *incomplete data problem*. For example, an incomplete data problem arises in signal processing when parameters have to be estimated from a coarsely quantized signal: the complete data are the analog values of the signal (non-measured), the incomplete data are the values of the signal quantized on a few bits. Other typical examples of incomplete data problems can

To appear in the Proceedings of the 2d IEEE European Workshop on Computationaly Intensive Methods in Control and Signal Processing, Prague, Czech Rep., August 28-30, 1996.

be found, e.g., in [1].

2.2. The EM Algorithm

The maximum-likelihood estimator $\hat{\theta}$ is the maximizer of the log-likelihood

$$L(\theta) = \ln q(\mathbf{y}|\theta) \tag{2}$$

over θ, i.e.,

$$\hat{\theta} = \arg\max_{\theta \in \Theta} L(\theta). \tag{3}$$

The main idea behind the EM algorithm is that, in some problems, the estimation of θ would be easy if the complete data $\mathbf{x}$ was available while it is difficult based on the incomplete data $\mathbf{y}$ only (i.e., the maximization of $\ln p(\mathbf{x}|\theta)$ over θ is easily performed while the maximization of $\ln q(\mathbf{y}|\theta)$ is complex). Since only the incomplete data $\mathbf{y}$ is available in practice, it is not possible to perform directly the optimization of the complete data likelihood $\ln p(\mathbf{x}|\theta)$. Instead, it seems intuitively reasonable to "estimate" $\log p(\mathbf{x}|\theta)$ from $\mathbf{y}$ and use this "estimated" likelihood function to obtain the maximizer $\hat{\theta}$. Since estimating the complete data likelihood $\ln p(\mathbf{x}|\theta)$ requires θ, it is necessary to use an iterative approach: first estimate the complete data likelihood given the current value of θ, then maximize this likelihood function over θ, and iterate, hoping for convergence. The "best estimate" of $\log p(\mathbf{x}|\theta)$ given a current value θ' of the parameters and $\mathbf{y}$ is the conditional expectation

$$Q(\theta, \theta') = E[\log p(\mathbf{x}|\theta)|\mathbf{y}, \theta']. \tag{4}$$

Following this heuristic argument, the E and M steps of the iterative EM algorithm (also known as the Generalized EM algorithm or GEM) can be formally expressed as:

E-step: compute

$$Q(\theta, \theta^{(p)}) = E[\log p(x|\theta)|\mathbf{y}, \theta^{(p)}], \tag{5}$$

M-step: choose

$$\theta^{(p+1)} \in \arg\max_{\theta \in \Theta} Q(\theta, \theta^{(p)}). \tag{6}$$

where $\theta^{(p)}$ denotes the value of the parameter obtained at the p-th iteration. Note that, if the complete data distribution belong to the exponential (Koopmans-Darmois) family, the algorithm takes a slightly simpler form [1]. The EM algorithm will be now illustrated on a notional example.

2.3. A Notional Example

Let $\mathbf{y} = (y_1, y_2, \ldots, y_N)$ be a sequence of i.i.d. observations drawn from a mixture of two univariate Gaussians with means μ_1 and μ_2, variances σ_1^2 and σ_2^2, and mixing proportions π_1 and π_2. That is, $y_k \sim q(y)$ where

$$q(y) = \pi_1 q_1(y) + \pi_2 q_2(y), \qquad y \in \mathbb{R} \tag{7}$$

with $\pi_1 + \pi_2 = 1$ and

$$q_j(y) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp -\frac{1}{2}\left(\frac{y - \mu_j}{\sigma}\right)^2, \qquad j = 1, 2.$$

For simplicity, assume that the variances and mixing proportions are known. The unknown parameters that have to be estimated from $\underline{y}$ are the means, i.e., $\theta = \{\mu_1, \mu_2\}$. The log-likelihood of θ is given by

$$\ln q(\mathbf{y}|\theta) = \sum_{k=1}^{N} \ln q(y_k|\theta). \tag{8}$$

The maximization of (8) can be easily performed by casting the mixture problem as an incomplete data problem and by using the EM algorithm. Drawing a sample y of a random variable with mixture pdf (7) can be interpreted as a two step process. First, a Bernoulli random variable i taking value 1 with probability π_1 or value 2 with probability $\pi_2 = 1 - \pi_1$ is drawn. According to the value of i, y is then drawn from one of the two populations with pdf $q_1(y)$ and $q_2(y)$. Of course, the "selector" variable i is not directly observed. The complete data is thus $\mathbf{x} = (x_1, x_2, \ldots, x_N)$ with $x_k = (y_k, i_k)$, and the associated complete data log-likelihood is

$$\ln p(\mathbf{x}|\theta) = \sum_{k=1}^{N} \ln p((y_k, i_k)|\theta)$$

with

$$\begin{aligned} p(x_k|\theta) &= \pi_{i_k} q_{i_k}(y_k) \\ &= \pi_1 q_1(y_k) 1_{\{i_k=1\}} + \pi_2 q_2(y_k) 1_{\{i_k=2\}}, \end{aligned}$$

where 1_A is the indicator function for the event A. The auxiliary function is then easily seen to be equal to

$$\begin{aligned} Q(\theta, \theta') &= E[\ln p(\mathbf{x}|\theta)|\mathbf{y}, \theta'] \\ &= \sum_{k=1}^{N} \sum_{j=1}^{2} [\ln \pi_j + \ln q_j(y_k)]\, P[i_k = j|\mathbf{y}, \theta']. \end{aligned}$$

From (9) it is straightforward to show that the EM algorithm (5)–(6) reduces to a pair of re-estimation formulae for the means of the mixture of two Gaussians:

$$\mu_1^{(p+1)} = \frac{1}{N} \sum_{k=1}^{N} y_k P[i_k = 1|y_k, \theta^{(p)}] \tag{9}$$

$$\mu_2^{(p+1)} = \frac{1}{N} \sum_{k=1}^{N} y_k P[i_k = 2|y_k, \theta^{(p)}] \tag{10}$$

where the *a posteriori* probabilities $P[i_k = j|y_k, \theta^{(p)}]$, $j = 1, 2$, can be obtained by the Bayes rule

$$P[i_k = j|y_k, \theta^{(p)}] = \frac{\pi_j q_j(y_k|\theta^{(p)})}{\sum_{j=1}^{2} \pi_j q_j(y_k|\theta^{(p)})}. \qquad (11)$$

These re-estimation formulae have a satisfying intuitive interpretation. If the complete data was observable, the ML estimators for the means of the mixture components would be

$$\hat{\mu}_j = \frac{1}{N} \sum_{k=1}^{N} y_k 1_{\{i_k=j\}}, \qquad j = 1, 2. \qquad (12)$$

That is, each of the observations y_k is classified as coming from the first or the second component distributions and the means are computed by averaging the classified observations. With only the incomplete data, the observations are still "classified" in some sense: at each iteration, they are assigned to both the first and the second component distributions with weights depending on the posterior probabilities given the current estimate of the means. The new estimates of the means are then computed by a weighted average.

3. Practical Applications

3.1. Motivation

The EM algorithm is mainly used in incomplete data problems when the direct maximization of the incomplete data likelihood is either not desirable or not possible. This can happen for various reasons. First, the incomplete data distribution $q(\mathbf{y}|\theta)$ may not be easily available while the form of the complete data distribution $p(\mathbf{x}|\theta)$ is known. Of course, relation (1) could be used, but the integral may not necessarily exists in closed form and its numerical computation may not be possible at a reasonable cost, especially in high dimension. Next, even if a closed form expression for $q(\mathbf{y}|\theta)$ is available, the implementation of a Gauss-Newton, scoring, or other direct maximization algorithm might be difficult because it requires a heavy preliminary analytical work in order to obtain the required derivatives (gradient or Hessian) of $q(\mathbf{y}|\theta)$ or because it requires too much programming work. The EM algorithm, on the other hand, can often be reduced to a very simple re-estimation procedure without much analytical work (like in the notional example of the previous section). Finally, in some problems, the high dimensionality of θ can lead to memory requirements for direct optimization algorithms exceeding the possibilities of the current generation of computers. The PET tomography application below is an example of how the EM algorithm can sometimes provide a solution requiring little storage in this case. There are other arguments in favor of the utilization of the EM algorithm; there are also some drawbacks. They will be discussed in the last sections.

To give the reader a flavor of the kind of ML problems in which the EM algorithm is currently used, we now briefly review some applications. It will be seen that the EM algorithm leads to an elegant and heuristically appealing formulation in many cases. The applications will be simply outlined and the interested reader will be referred to the literature for further details. As much as possible, we tried to provide references to the key papers in each field rather than attempting to give an exhaustive bibliographic review (which would have been outside of the scope of this paper anyway). We also tried to provide examples that are of interest for the control and signal processing community.

3.2. Examples of Applications

3.2.1 Mixture Densities: A family of finite mixture densities is of the form

$$q(y|\theta) = \sum_{j=1}^{K} \pi_j q_j(y|\phi_j), \qquad y \in \mathbb{R}^d \qquad (13)$$

where $\pi_j \geq 0$, $\sum_{j=1}^{K} \pi_j = 1$, $q_j(y|\phi_j)$ is itself a density parameterized by ϕ_j, and $\theta = \{\pi_1, \ldots, \pi_K, \phi_1, \ldots, \phi_K\}$. The complete data is naturally formulated as the combination of the observations y with multinomial random variables i acting as "selectors" for the component densities $q_j(y|\phi_i)$, like in the notional example. Let $\mathbf{y} = (y_1, y_2, \ldots, y_N)$ be a sample of i.i.d. observation, $y_k \sim q(y_k|\theta)$. It can be shown [2] that the EM algorithm for the ML estimation of θ reduces to the set of re-estimation formulae

$$\pi_j^{(p+1)} = \frac{1}{N} \sum_{k=1}^{N} \frac{\pi_j^{(p+1)} q_j(y_k|\phi_j^{(p)})}{q(y_k|\theta^{(p)})},$$

$$\phi_j^{(p+1)} \in \arg\max_{\phi_j} \sum_{k=1}^{N} \left[\ln q_j(y_k|\phi_j) \times \frac{\pi_j^{(p+1)} q_j(y_k|\phi_j^{(p)})}{q(y_k|\theta^{(p)})} \right],$$

for $j = 1, \ldots, K$. Again, the solution has a heuristically appealing interpretation as a weighted ML solution. The weight associated with y_k is the posterior probabilities that the sample originated from the jth distribution, i.e., the posterior probability that the selector variable i_k is equal to j. Furthermore, in most applications of interest $\phi^{(p+1)}$ is uniquely and easily determined from (14), like in the mixture of two Gaussians presented in the notional example of Section 2.3.

The EM algorithm for mixture densities is widely

used in statistics and signal processing, for example, for clustering or for vector quantization with a mixture of multivariate Gaussians. Moreover, the well-known Baum-Welsh algorithm used for the training of hidden Markov models in speech recognition [3] is also an instance of the EM algorithm for mixtures with a particular Markov distribution for the "selectors" i_k [4].

3.2.2 PET Tomography: The EM algorithm has been used for over a decade to compute ML estimate of radionucleide distributions from tomographic data, such as that measured by positron emission tomography (PET) [5][6]. It relies on the following statistical model. Assume that the radionucleide distribution discretized into d pixels with emission rates $\theta = (\lambda_1, \ldots, \lambda_d)$. Assume that there are N detectors and let x_{nk} denote the number of emissions from the kth pixel that are detected by the nth detector. The variates x_{nk} are assumed to have independent Poisson distribution:

$$x_{nk} \sim \text{Poisson with rate } a_{nk}\lambda_k,$$

where the a_{nk} are non-negative (known) constants that characterize the measurement system. Neglecting background emissions, random coincidences, and scatter contamination, the total number of detection at the nth sensor is $y_n = \sum_{k=1}^{d} x_{nk}$. The ML estimate of θ can be obtained by applying the EM algorithm to the complete data $\mathbf{x} = (x_{nk})$, $1 \le n \le N$, $1 \le k \le d$, with the incomplete data $\mathbf{y} = (y_1, \ldots, y_N)$. It can be shown that the EM algorithm reduces to re-estimation formulae which are extremely simple and easy to implement [6]. Many variations of the EM algorithm have been proposed for PET image reconstruction, e.g., [7][8].

3.2.3 System Identification: Consider the discrete-time linear stochastic system with state and observation equations

$$\begin{aligned} x_{t+1} &= \mathbf{F}x_t + u_t \\ y_t &= \mathbf{H}x_t + v_t \end{aligned}$$

where u_t and v_t are Gaussian zero-mean vector random processes with covariance matrices Σ_u and Σ_v, respectively, and $\mathbf{F}$ and $\mathbf{H}$ are matrices of appropriate dimensions. Let $\mathbf{y} = (y_1, y_2, \ldots, y_N)$ be a length N sample of the output of the system, and let $\theta = \{\mathbf{F}, \mathbf{H}, \Sigma_u, \Sigma_v\}$ be the parameter set of interest. The estimation of θ from $\mathbf{y}$ lends itself naturally to a formulation as an incomplete data problem and the EM algorithm can be used to compute $\hat{\theta}$. In this case, the complete data is simply the combination of the state and observation vectors, i.e., $\mathbf{x} = ((x_1, y_1), \ldots, (x_N, y_N))$. The E-step can be handled by a Kalman smoother, and the M-step reduces to a linear system of equations with a closed form solution [9] (see also [10] and [11]). This EM approach to ML system identification can be straightforwardly extended to deal with missing observations [9][12] or coarsely quantized observations [13].

4. Convergence Properties

It is possible to prove some general convergence properties of EM algorithms. Since the EM algorithm is a "meta-algorithm," a method for implementing ML algorithms, the results are universal in the sense that they apply to the maximization of a wide class of incomplete data likelihood functions.

4.1. Monotonous Increase of the Likelihood

The sequence $\{\theta^{(p)}\}$ generated by the EM algorithm increases monotonously the likelihood $L(\theta)$; that is,

$$L(\theta^{(p+1)}) \ge L(\theta^{(p)}).$$

This property is a direct corollary of the next theorem.

Theorem: If

$$Q(\theta, \theta') \ge Q(\theta', \theta')$$

then

$$L(\theta) \ge L(\theta').$$

Proof: Let $r(\mathbf{x}|\mathbf{y}, \theta)$ denote the conditional distribution of $\mathbf{x}$ given $\mathbf{y}$, $r(\mathbf{x}|\mathbf{y}, \theta) = p(\mathbf{x}|\theta)/q(\mathbf{y}|\theta)$, and let

$$V(\theta, \theta') = E[\ln r(\mathbf{x}|\underline{\mathrm{y}}, \theta)|\mathbf{y}, \theta'].$$

From (2), (4), and this definition, we have

$$L(\theta) = Q(\theta, \theta') - V(\theta, \theta').$$

Invoking Jensen's inequality, we get

$$V(\theta, \theta') \le V(\theta', \theta'),$$

and the theorem follows. □

4.2. Convergence to a Local Maxima

The global maximization of the auxiliary function performed during the M-step can be misleading. With the exception of a few specific cases, the EM algorithm is *not* guaranteed to converge to a global maximizer of the likelihood. Under some regularity conditions on the likelihood $L(\theta)$ and on the parameters set Θ, it is possible, however, to show that the sequence $\{\theta^{(p)}\}$ obtained by EM algorithm converges to a local maximizer of $L(\theta)$, or, at least, to a stationary point of $L(\theta)$. Necessary conditions for the convergence of the EM algorithm and related theorems can be found in [14]. Note that the original proof of convergence of the EM algorithm given in [1] was incorrect (see the counter-example

of Boyles [15]). Convergence results are also available for various particular applications of the EM algorithm, e.g., in [2] for mixtures of densities.

Remark: The reader should not confuse the *algorithmic* convergence of the EM algorithm towards a local maximizer of the likelihood function for given data with the *stochastic* convergence of the likelihood estimate towards the true parameters when the amount of observed data increases (i.e., the consistency of the maximum likelihood estimator).

4.3. Speed of Convergence

It can be shown that, near the solution, the EM algorithm converges linearly. The rate of convergence corresponds to the fraction of the variance of the complete data score function unexplained by the incomplete data [1][16] (see also [17]). That is, if the complete data model is much more informative about θ than the incomplete data model, then the EM algorithm will converge slowly.

5. Variants of the EM Algorithm

5.1. Acceleration of the Algorithm

In practice, the convergence of the EM algorithm can be desperately slow in some case. Roughly speaking, the EM algorithm is the equivalent of a gradient method whose linear convergence is well known. Variants of the EM algorithms with improved convergence speed have been proposed. They are usually based on the application to the EM algorithm of optimization theory techniques such as conjugate gradient [18], Aitkin's acceleration [19], or coordinate ascent [7][10]. Many acceleration schemes have also been proposed for specific EM applications.

5.2. Approximation of the E or M Step

Another cause of slowness of the EM algorithm arises when the E or M step do not admit an analytical solution. It becomes then necessary to use iterative methods for the computation of the expectation or for the maximization, which can be computationally expensive. Variants of the EM algorithm preserving its convergence properties have been proposed that can alleviate this problem, e.g., [20][21][22][23]. They are based on approximations of the E or M steps that preserve the convergence properties of the algorithm. For example, it is shown in [20] and [24] that the algorithm still converges if a Monte-Carlo approximation of the E step is used. Furthermore, this approximation can even decrease the probability of getting stuck in a local maxima.

5.3. Penalized Likelihood Estimation

The EM algorithm can be straightforwardly modified to compute *penalized likelihood* estimates [1], that is, estimates of the form

$$\tilde{\theta} = \arg\max_{\theta \in \Theta} [L(\theta) + G(\theta)] .$$

The penalty term $G(\theta)$ could represent, for example, the logarithm of a prior on θ if a Bayesian approach is used and the maximum *a posteriori* (MAP) estimate of θ is desired instead of the ML estimate. The EM algorithm for penalized-likelihood estimation can be obtained by replacing the M-step with [1]

$$\theta^{(p+1)} = \arg\max_{\theta \in \Theta} \left[Q(\theta, \theta^{(p)}) + G(\theta) \right] .$$

It is straightforward to see that the monotonicity property of Section 4.1 is preserved, i.e., $L(\theta^{(p+1)}) + G(\theta^{(p+1)}) \geq L(\theta^{(p)}) + G(\theta^{(p)})$. Some extension of the EM algorithm for dealing specifically with penalized likelihood problems have been proposed, e.g., in [25] and [26]. It is also noted in [25] that the inclusion of a penalty term can speed up the convergence of the EM algorithm.

6. Concluding Remarks

As with all numerical methods, the EM algorithm should not be used with uncritical faith. In fact, given an identification/estimation problem the engineer should first ask whether maximum likelihood is a good method for the specific application, and only then if the EM algorithm is a good method for the maximization of the likelihood. The alternatives to the EM algorithm include the scoring and Newton-Raphson methods that are commonly used in statistics and any other numerical maximization method that can be applied to the likelihood function. When is the EM algorithm a reasonable approach to a maximum-likelihood problem? Compared to its rivals, the EM algorithm possesses a series of advantages and disadvantages. The decision to use the EM algorithm should be based on an analysis of the trade-offs between those.

The main advantages of the EM algorithm are its simplicity and ease of implementation. Unlike, say, the Newton-Raphson method, implementing the EM algorithm does not usually require heavy preparatory analytical work. It is easy to program: either it reduces to very simple re-estimation formulae or it is possible to use standard code to perform the E step (like the Kalman smoother in the examples of Section 3.2.3). Because of its simplicity, it can often be easily parallelized and its memory requirements tend to be modest compared to other methods. Also, the EM algorithm is numerically very stable. In addition, it can often provide fitted

values for the complete data without the need of further computation (they are obtained during the E step).

The main disadvantage of the EM algorithm is its hopelessly slow linear convergence is some case. Of course, the acceleration schemes of Section 5.1 can be used, but they generally require some preparatory analytical work and they increase the complexity of the implementation. Thus, the simplicity advantages over other alternative methods may be lost. Furthermore, unlike other methods based on the computation of derivatives of the incomplete data log-likelihood, the EM algorithm does not provide an estimate of the information matrix of $\hat{\theta}$ as a by-product, which can be a drawback when these estimates are desired. Extensions of the EM algorithm have been proposed for that purpose though ([27] and references therein, or [16][19]), but, again, they increase the complexity of the implementation.

Finally, a word of advice for the practicing engineer interested in implementing the EM algorithm. The EM algorithm requires an initial estimate of θ. Since multiple local maxima of the likelihood function are frequent in practice and the algorithm converges only to a local maxima, the quality of the initial estimate can greatly influence the final result. The initial estimate should be carefully chosen. As with all numerical optimization methods, it is often sound to try various initial starting points. Also, because of the slowness of convergence of the EM algorithm, the stopping criterion should be selected with care.

In conclusion, the EM algorithm is a simple and versatile procedure for likelihood maximization in incomplete data problems. It is elegant, easy to implement, numerically very stable, and its memory requirements are generally reasonable, even in very large problems. However, it also suffers from several drawbacks, the main one being its hopelessly slow convergence in some cases. Nevertheless, we believe that the EM algorithm should be part of the "numerical toolbox" of any engineer dealing with maximum likelihood estimation problems.

References

[1] A. P. Dempster, N. M. Laird, and D. B. Rubin. *J. Roy. Stat. Soc. B* **39**, 1–38 (1977).

[2] R. A. Redner and H. F. Walker. *SIAM Rev.* **26**(2), 192–239 (Apr. 1984).

[3] L. R. Rabiner. *Proc. IEEE* **77**(2), 257–286 (Feb. 1989).

[4] D. M. Titterington. *Statistics* **21**(4), 619–641 (1990).

[5] L. A. Shepp and Y. Vardi. *IEEE Trans. Med. Imag.* **1**(2), 113–122 (Oct. 1982).

[6] Y. Vardi, L. A. Shepp, and L. Kaufman. *J. Am. Stat. Assoc.* **80**, 8–37 (1985).

[7] J. A. Fessler and A. O. Hero. *IEEE Trans. Sig. Proc.* **42**(10), 2664–2677 (Oct. 1994).

[8] B. W. Silverman, M. C. Jones, J. D. Wilson, and D. W. Nychka. *J. R. Stat. Soc. B* **52**(2), 271–324 (May 1990).

[9] R. H. Shumway and D. S. Stoffer. *J. Time Series Anal.* **3**(4), 253–264 (1982).

[10] M. Segal and E. Weinstein. *Proc. IEEE* **76**(10), 1388–1390 (Oct. 1988).

[11] M. Segal and E. Weinstein. *IEEE Trans. Inf. Theory* **35**(3), 682–687 (May 1989).

[12] V. Digalakis, J. R. Rohlicek, and M. Ostendorf. *IEEE Trans. Sp. Audio Proc.* **1**(4), 431–442 (Oct. 1993).

[13] I. Ziskand and D. Hertz. *IEEE Trans. Sig. Proc.* **41**(11), 3202–3206 (Nov. 1993).

[14] C. F. J. Wu. *Ann. Stat.* **11**(1), 95–103 (1983).

[15] R. A. Boyles. *J. Roy. Stat. Soc. B* **45**(1), 47–50 (1983).

[16] T. A. Louis. *J. Roy. Stat. Soc. B* **44**(2), 226–233 (1982).

[17] X. L. Meng and D. B. Rubin. *Lin. Alg. and Appl.* **199**, 413–425 (Mar. 1994).

[18] M. Jamshidian and R. I. Jennrich. *J. Am. Stat. Assoc.* **88**(421), 221–228 (Mar. 1993).

[19] I. Meilijson. *J. Roy. Stat. Soc. B* **51**(1), 127–138 (1989).

[20] G. Celeux and J. Diebolt. *C. R. Acad. Sc. Paris I* **310**, 119–124 (1990). in French.

[21] J.-F. Cardoso, M. Lavielle, and E. Moulines. *C. R. Acad. Sc. Paris I* **320**, 363–368 (1995). in French.

[22] X. L. Meng and D. B. Rubin. *Biometrika* **80**, 267–278 (1993).

[23] K. Lange. *J. Roy. Stat. Soc. B* **57**(2), 425–437 (1995).

[24] G. Celeux and J. Diebolt. *Rev. Stat. Appl.* **24**(2), 35–52 (1986). in French.

[25] P. J. Green. *J. Roy. Stat. Soc. B* **52**(3), 443–452 (1990).

[26] M. R. Segal, P. Bacchetti, and N. P. Jewell. *J. Roy. Stat. Soc. B* **56**(2), 345–352 (1994).

[27] X.-L. Meng and D. B. Rubin. *J. Am. Stat. Assoc.* **86**(416), 899–909 (Dec. 1991).