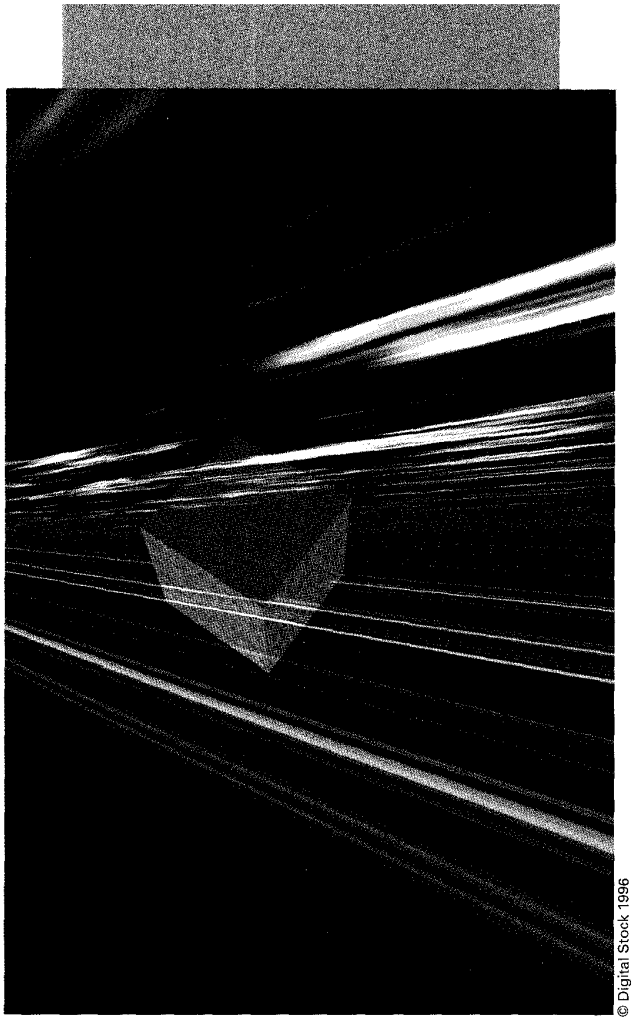


The Expectation-Maximization Algorithm

TODD K. MOON

A common task in signal processing is the estimation of the parameters of a probability distribution function. Perhaps the most frequently encountered estimation problem is the estimation of the mean of a signal in noise. In many parameter estimation problems the situation is more complicated because direct access to the data necessary to estimate the parameters is impossible, or some of the data are missing. Such difficulties arise when an outcome is a result of an accumulation of simpler outcomes, or when outcomes are clumped together, for example, in a binning or histogram operation. There may also be data dropouts or clustering in such a way that the number of underlying data points is unknown (censoring and/or truncation). The EM (expectation-maximization) algorithm is ideally suited to problems of this sort, in that it produces maximum-likelihood (ML) estimates of parameters when there is a many-to-one mapping from an underlying distribution to the distribution governing the observation. In this article, the EM algorithm is presented at a level suitable for signal processing practitioners who have had some exposure to estimation theory. (A brief summary of ML estimation is provided in Box 1 for review.)

The EM algorithm consists of two major steps: an expectation step, followed by a maximization step. The expectation is with respect to the unknown underlying variables, using the current estimate of the parameters and conditioned upon the observations. The maximization step then provides a new esti-

1053-5888/96/$5.00©1996IEEE

mate of the parameters. These two steps are iterated until convergence. The concept is illustrated in Fig. 1.

The EM algorithm was discovered and employed independently by several different researchers until Dempster [1] brought their ideas together, proved convergence, and coined the term "EM algorithm." Since that seminal work, hundreds of papers employing the EM algorithm in many areas have been published. A large list of references is found at [2]. A typical application area of the EM algorithm is in genetics, where the observed data (the phenotype) is a function of the underlying, unobserved gene pattern (the genotype), e.g. [3]. Another area is estimating parameters of mixture distributions, e.g. [4]. The EM algorithm has also been widely used in econometric, clinical, and sociological studies that have unknown factors affecting the outcomes [5]. Some applications to the theory of statistical methods are found in [6].

In the area of signal processing applications, the largest area of interest in the EM algorithm is in maximum likelihood tomographic image reconstruction, e.g. [7, 8] Another commonly cited application is training of hidden Markov models, especially for speech recognition, e.g. [9]. The books [10, 11] have chapters with extensive development on hidden Markov models (HMMs).

Other signal processing and engineering applications began appearing in about 1985. These include: parameter estimation [12, 13]; ARMA modeling [14, 15]; image modeling, reconstruction, and processing [16, 17]; simultaneous detection and estimation [18, 19, 20]; pattern recognition and neural network training [21, 22, 23]; direction finding [24]; noise suppression [25]; spectroscopy [27]; signal and sequence detection [28]; time-delay estimation [29]; and specialized developments of the EM algorithm itself [30]. The EM algorithm has been the subject for multiprocessing algorithm development [31]. The EM algorithm is also related to algorithms used in information theory to compute channel capacity and rate distortion functions [32, 33], since the expectation step in the EM algorithm produces a result similar to entropy. The EM algorithm is philosophically similar to ML detection in the presence of unknown phase (incoherent detection) or other unknown parameters: the likelihood function is averaged with respect to the unknown quantity (i.e., the expected value of the likelihood function is computed) before detection, which is a maximization step (see, e.g., [34, Chap. 5].

Ector's Problem: An Introductory Example

The image-processing example introduced by Ector and Hatter (see the "Tale of Two Distributions" sidebar), although somewhat contrived, illustrates most of the principles of the EM algorithm as well as the notational conventions of this article. In many aspects it is similar to a problem that is of practical interest — the emission tomography (ET) problem discussed later in this article.

Suppose that in an image pattern-recognition problem, there are two general classes to be distinguished: a class of dark objects and a class of light objects. The class of dark objects may be further subdivided into two shapes: round and square. Using a pattern recognizer, it is desired to determine the probability of a dark object. For the sake of the example, assume that the objects are known to be trinomially distributed. Let the random variable X_1 represent the number of round dark objects, X_2 represent the number of square dark objects, and X_3 represent the number of light objects, and let $[x_1, x_2, x_3]^T = \mathbf{x}$ be the vector of values the random variables take for some image. (In this article the convention is that vectors are printed in bold font, and scalars are printed in math italic. All vectors by convention are taken as column vectors. Uppercase letters are random variables.) Assume further that enough is known about the probabilities of the different classes so that the probability may be written as

$$P(X_1 = x_1, X_2 = x_2, X_3 = x_3 | p) \tag{1}$$

$$= \left(\frac{n!}{x_1! x_2! x_3!} \right) \left(\frac{1}{4} \right)^{x_1} \left(\frac{1}{4} + \frac{p}{4} \right)^{x_2} \left(\frac{1}{2} - \frac{p}{4} \right)^{x_3},$$

$$= f(x_1, x_2, x_3 | p) \tag{2}$$

where p is an unknown parameter of the distribution and $n = x_1+x_2+x_3$. The notation $f(x_1,x_2,x_3|p)$ is typical throughout the article; it is used to indicate the probability function which

A Tale of Two Distributions

D.T. Ector and S. Tim Hatter, the distinguished signal processors, build a line of image-processing hardware that computes the maximum likelihood estimate of parameters governing the distribution of light and dark regions of an image. Ector and Hatter learn from their contact, C.B. Daily, about the need for a detector that can estimate model parameters for a model that distinguishes between two different types of dark objects, round and square. Ever on the prowl for new markets, they decide to adapt their old detector to the new problem. Ector and Hatter hand the project off to their trusted assistant, Matt A. Titian, who discovers how to solve the problem using the old detector hardware combined with a new algorithm.

Matt solves the problem by knowing about the distribution of the three different classes of objects: round dark, square dark, and light. For the particular problem, he learns there is only a single parameter governing the distribution of all three classes. By choosing (guessing) an initial value for this parameter, he is able to estimate, using only the number of light and dark regions, the number of the two different types of dark regions. Then, using these estimates as the true values, he comes up with another estimate for the underlying parameter. He iterates the process until it converges. It works!

Ector and Hatter are so pleased with Matt's work, especially when they learn that it is a special case of a whole class of problems, that they give him the rest of the afternoon off, with pay, to play with his calculator.

may be either a probability density function (pdf) or a probability mass function (pmf).

A feature extractor is employed that can distinguish which objects are light and which are dark, but cannot distinguish shape. Let $[y_1,y_2]^T = \mathbf{y}$ be the number of dark objects and number of light objects detected, respectively, so that $y_1 = x_1 + x_2$ and $y_2 = x_3$, and let the corresponding random variables be Y_1 and Y_2. There is a many-to-one mapping between $\{x_1,x_2\}$ and y_1. For example, if $y_1 = 3$, there is no way to tell from the measurements whether $x_1 = 1$ and $x_2 = 2$ or $x_1 = 2$ and $x_2 = 1$. The EM algorithm is specifically designed for problems with such many-to-one mappings. Then (see Box 2),

$$P(Y_1 = y_1|p) = \binom{n}{y_1}\left(\frac{1}{2}+\frac{p}{4}\right)^{y_1}\left(\frac{1}{2}-\frac{p}{4}\right)^{n-y_1} = g(y_1|p)$$

(The symbol g is used to indicate the probability function for the observed data.) From the observation of y_1 and y_2, compute the ML estimate of p,

$$p_{ML} = \arg\max_p g(Y_1 = y_1|p), \tag{3}$$

where "argmax" means "the value that maximizes the function." In this example, it would be a simple matter to determine an ML estimate of p. In more interesting problems, however, such straightforward estimation is not possible. In the interest of introducing the EM algorithm, we will not take the direct approach to the ML estimate. Taking the logarithm of the likelihood often simplifies the maximization and yields equivalent results since log is an increasing function, so Eq. (3) may be written as

$$p_{ML} = \arg\max_p \log\binom{n}{y_1}\left(\frac{1}{2}+\frac{p}{4}\right)^{y_1}\left(\frac{1}{2}-\frac{p}{4}\right)^{n-y_1}. \tag{4}$$

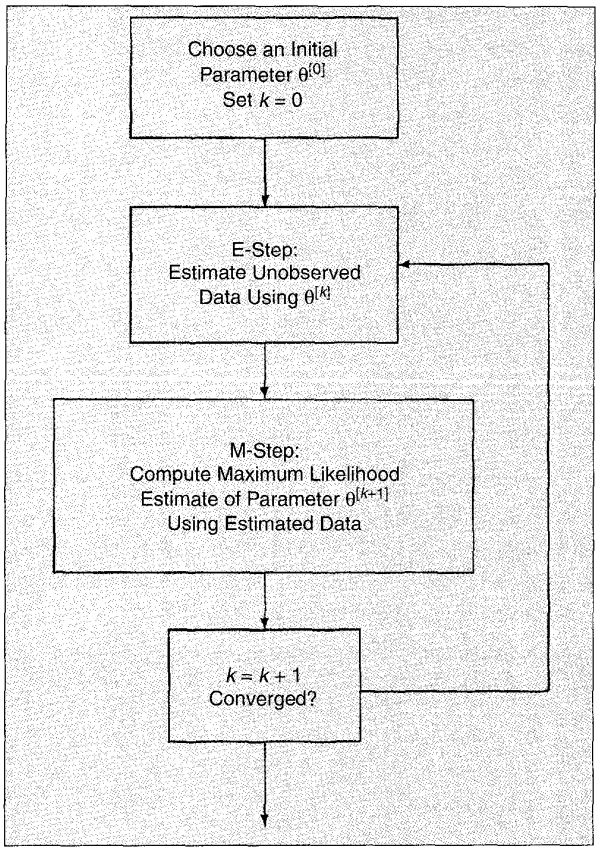
The idea behind the EM algorithm is that, even though we do not know x_1 and x_2, knowledge of the underlying distribution $f(x_1, x_2, x_3|p)$ can be used to determine an estimate for p. This is done by first estimating the underlying data, then using these data to update our estimate of the parameter. This is repeated until convergence. Let $p^{[k]}$ indicate the estimate of p after the kth iteration, $k = 1,2,....$ An initial parameter value $p^{[0]}$ is assumed. The algorithm consists of two major steps:

Expectation Step (E-step). Compute the expected value of the **x** data using the current estimate of the parameter and the observed data.

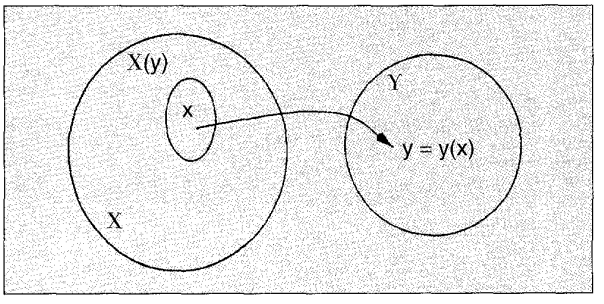
The expected value of x_1, given the measurement y_1 and based upon the current estimate of the parameter, may be computed as

$$x_1^{[k+1]} = E\left[x_1|y_1,p^{[k]}\right].$$

Using the results of Box 2,



1. An overview of the EM algorithm. After initialization, the E-step and the M-step are alternated until the parameter estimate has converged (no more change in the estimate).



2. Illustration of many-to-one mapping from X *to* Y. *The point* **y** *is the image of* **x**, *and the set* X(**y**) *is the inverse map of* **y**.

$$x_1^{[k+1]} = y_1\frac{\frac{1}{4}}{\frac{1}{2}+\frac{p^{[k]}}{2}}. \tag{5}$$

Similarly,

$$x_2^{[k+1]} = E[x_1|y_1,p^{[k]}] = y_1\frac{\frac{1}{4}+\frac{p^{[k]}}{2}}{\frac{1}{2}+\frac{p^{[k]}}{2}}. \tag{6}$$

In the current example, x_3 is known and does not need to be computed.

$$l(p) = \log\binom{n}{y_1}\left(\frac{1}{2}+\frac{p}{4}\right)^{y_1}\left(\frac{1}{2}-\frac{p}{4}\right)^{n-y_1}$$

$$= \log\binom{n}{y_1}\left(\frac{2+p}{4}\right)^{y_1}\left(\frac{2-p}{4}\right)^{n-y_1}$$

$$= \log\binom{n}{y_1} + y_1\log(2+p) - y_1\log 4 + (n-y_1)\log(2-p) - (n-y_1)\log 4$$

$$\frac{\partial l(p)}{\partial p} = \frac{y_1}{p+2} - \frac{n-y_1}{2-p} = 0$$

$$2y_1 - py_1 - (np - py_1 + 2n - 2y_1) = 0$$

$$np = 4y_1 - 2n$$

$$p = \frac{4y_1 - 2n}{n}$$

$\frac{1-\theta}{4}$ $\frac{2+\theta}{4}$

$$l(p) = \log(\quad)\cdot\left(\frac{1}{4}\right)^{x_1}\left(\frac{1}{4}+\frac{1}{4}p\right)^{x_2}\left(\frac{1}{2}-\frac{1}{4}p\right)^{x_3}$$

$$= x_2\log\left(\frac{p+1}{4}\right) + x_3\log\left(\frac{2-p}{4}\right)$$

Maximization Step (M-step). Use the data from the expectation step as if it were actually measured data to determine an ML estimate of the parameter. This estimated data is sometimes called "imputed" data.

In this example, with $x_1^{[k+1]}$ and $x_2^{[k+1]}$ imputed and x_3 available, the ML estimate of the parameter is obtained by taking the derivative of $\log f(x_1^{[k+1]}, x_2^{[k+1]}, x_3|p)$ with respect to p, equating it to zero, and solving for p,

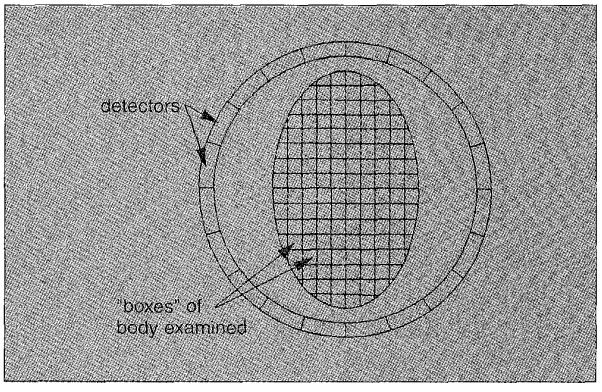
$$0 = \frac{d}{dp}\log f(x_1^{[k+1]}, x_2^{[k+1]}, x_3|p) \tag{7}$$

$$\Rightarrow p^{[k+1]} = \frac{2x_2^{[k+1]} - x_3}{x_2^{[k+1]} + x_3}$$

The estimate $x_1^{[k+1]}$ is not used in Eq. (7) and so, for this example, need not be computed. The EM algorithm consists of iterating Eqs. (6) and (7) until convergence. Intermediate computation and storage may be eliminated by substituting Eq. (6) into Eq. (7) to obtain a one-step update:

Table 1. Results of the EM algorithm for an example using trinomial data

k	$x_1^{[k]}$	$x_2^{[k]}$	$p^{[k]}$
1	31.500000	31.500000	0.379562
2	26.475460	36.524540	0.490300
3	25.298157	37.701843	0.514093
4	25.058740	37.941260	0.518840
5	25.011514	37.988486	0.519773
6	25.002255	37.997745	0.519956
7	25.000441	37.999559	0.519991
8	25.000086	37.999914	0.519998
9	25.000017	37.999983	0.520000
10	25.000003	37.999997	0.520000



3. Representation of ET. There are B boxes in the body and D detectors surrounding the body.

$$p^{[k+1]} = \frac{p^{[k]}(4y_1 - 2x_3) + 2y_1 - 2x_3}{p^{[k]}(2y_1 + 2x_3) + y_1 + 2x_3}. \tag{8}$$

As a numerical example, suppose that the true parameter is $p = 0.5$, $n = 100$ samples are drawn, with $y_1 = 100$. (The true values of x_1 and x_2 are 25 and 38, respectively, but the algorithm does not know this.) Table 1 illustrates the result of the algorithm starting from $p^{[0]} = 0$. The final estimate $p^* = 0.52$ is in fact the ML estimate of p that would have been obtained by maximizing Eq. (1) with respect to p, had the x data been available.

General Statement of the EM Algorithm

Let Y denote the sample space of the observations, and let $\mathbf{y} \in \mathbb{R}^m$ denote an observation from Y. Let χ denote the underlying space and let $\mathbf{x} \in \mathbb{R}^n$ be an outcome from χ, with $m < n$. The data $\mathbf{x}$ is referred to as the *complete data*. The complete data $\mathbf{x}$ is not observed directly, but only by means of $\mathbf{y}$, where $\mathbf{y} = \mathbf{y}(\mathbf{x})$, and $\mathbf{y}(\mathbf{x})$ is a many-to-one mapping. An observation $\mathbf{y}$ determines a subset of χ, which is denoted as $\chi(\mathbf{y})$. Figure 2 illustrates the mapping.

The probability density function (pdf) of the complete data is $f_X(\mathbf{x}|\theta) = f(\mathbf{x}|\theta)$, where $\theta \in \Theta \subset \mathbb{R}^r$ is the set of parameters of the density. (We will refer to the *density* of the random variables for convenience, even for discrete random variables for which *probability mass function* (pmf) would be appropriate. Subscripts indicating the random variable are suppressed, with the argument to the density indicating the random variable.) The pdf f is assumed to be a continuous function of θ and appropriately differentiable. The ML estimate of θ is assumed to lie within the region Θ. The pdf of the incomplete data is

$$g(\mathbf{y}|\theta) = \int_{\chi(\mathbf{y})} f(\mathbf{x}|\theta)d\mathbf{x}$$

Let

$$l_y(\theta) = g(\mathbf{y}|\theta)$$

denote the likelihood function and let

$$L_y(\theta) = \log g(\mathbf{y}|\theta)$$

denote the log-likelihood function.

The basic idea behind the EM algorithm is that we would like to find θ to maximize $\log f(\mathbf{x}|\theta)$, but we do not have the data $\mathbf{x}$ to compute the log-likelihood. So instead, we maximize the expectation of $\log f(\mathbf{x}|\theta)$ given the data $\mathbf{y}$ and our current estimate of θ. This can be expressed in two steps. Let $\theta^{[k]}$ be our estimate of the parameters at the kth iteration.

For the E-step compute:

$$Q(\theta|\theta^{[k]}) = E[\log f(\mathbf{x}|\theta)|\mathbf{y}, \theta^{[k]}]. \tag{9}$$

It is important to distinguish between the first and second arguments of the Q functions. The second argument is a

Box 1: Maximum-likelihood estimation

Maximum-likelihood (ML) estimation is a means of estimating the parameters of a distribution based upon observed data drawn according to that distribution. Let $\theta = [\theta_1, \theta_2, ..., \theta_r]^T$ denote a set of parameters. Let x be data observed from a distribution X with pdf (or pmf) $f_X(\mathbf{x}|\theta) = f(\mathbf{x}|\theta)$, parameterized by the set of parameters θ. Let $x_1, x_2, ..., x_N$ be a sequence of outcomes of the random variables $X_1, X_2, ..., X_N$ that have been observed. It is often assumed that X_i is independent of X_j for $i \neq j$. The key idea in ML estimation is to determine the parameter θ for which the probability of observing the outcome $\mathbf{x} = x_1, x_2, ..., x_N$ is as high as possible.

The function

$$l_X(\theta; x_1, x_2, ..., x_N) = f(x_1, x_2, ..., x_N|\theta) = f(\mathbf{x}|\theta)$$

is the *likelihood function*. It is viewed as a function of the parameter θ with the samples $\mathbf{x}$ fixed, in contrast to the pdf, in which the parameter is considered fixed. Because the data is assumed fixed in the likelihood function, it is common to suppress the dependence on the data and write simply $l_x(\theta)$ or even $l(\theta)$. The ML estimate of the parameter is that value of parameters which maximizes the likelihood function:

$$\theta_{ML} = \arg\max_{\theta} l_x(\theta)$$

Because it is the maximizing value (the argument) that is important in ML estimation, not the value of the maximum (the function), it is common to ignore or suppress constants in the likelihood function that do not depend upon the parameter. Also, in many applications it is more convenient to consider the logarithm of the likelihood function, called the log-likelihood function:

$$L_x(\theta) = \log l_x(\theta).$$

Since the logarithm is monotonically increasing, maximizing the log-likelihood is equivalent to maximizing the likelihood.

In many (but not all) cases, the log-likelihood function is a continuous differentiable function of the parameter and the maximizing θ lies in the interior of its range. In this case, a necessary (but not sufficient) condition to maximize the (log) likelihood is for the gradient to vanish at the value of θ that is the ML value:

$$\nabla_\theta l_x(\theta)|_{\theta=\theta_{ML}} = \nabla_\theta \log L_x(\theta)|_{\theta=\theta_{ML}} = 0$$

where

$$\nabla_\theta = \begin{bmatrix} \frac{\partial}{\partial\theta_1} \\ \frac{\partial}{\partial\theta_2} \\ \vdots \\ \frac{\partial}{\partial\theta_r} \end{bmatrix}.$$

As an example of ML estimation, let $X_1, X_2, ..., X_N$ be independent Gaussian random variables with unknown mean μ and variance σ^2 and let $x_1, x_2, ..., x_N$ be samples of these random variables. It is straightforward to show [50] that the ML estimate of the mean and the variance are

$$\mu_{ML} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\sigma^2_{ML} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2$$

For more details and examples of ML estimation, including results about the quality (variance) of the resulting estimates, the interested reader is encouraged to consult texts such as [50, 51].

conditioning argument to the expectation and is regarded as fixed and known at every E-step. The first argument conditions the likelihood of the complete data.

For the M-step let $\theta^{[k+1]}$ be that value of θ which maximizes $Q(\theta|\theta^{[k]})$:

$$\theta^{[k+1]} = \arg\max_{\theta} Q(\theta|\theta^{[k]}) \quad (10)$$

It is important to note that the maximization is with respect to the first argument of the Q function, the conditioner of the complete data likelihood.

The EM algorithm consists of choosing an initial $\theta^{[k]}$, then performing the E-step and the M-step successively until convergence. Convergence may be determined by examining when the parameters quit changing, i.e., stop when $\|\theta^{[k]} - \theta^{[k-1]}\| < \epsilon$ for some ϵ and some appropriate distance measure $\|\cdot\|$.

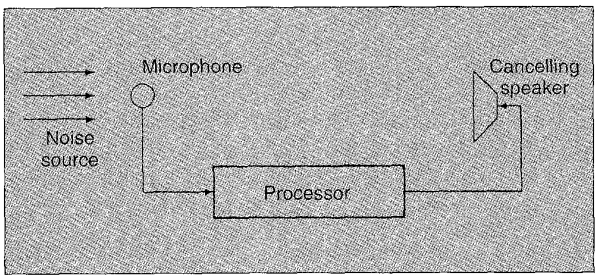
The general form of the EM algorithm as stated in Eqs. (9) and (10) may be specialized and simplified somewhat by restriction to distributions in the *exponential family*. These are pdfs (or pmfs) of the form

$$f(\mathbf{x}|\theta) = b(\mathbf{x}) \exp[\mathbf{c}(\theta)^T \mathbf{t}(\mathbf{x})]/a(\theta) \quad (11)$$

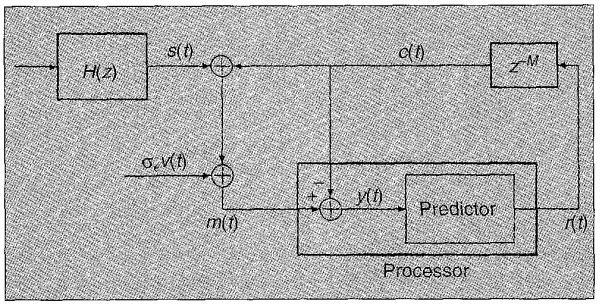
where θ is a vector of parameters for the family [35, 36]. The function $\mathbf{t}(\mathbf{x})$ is called the *sufficient statistic* of the family (a statistic is sufficient if it provides all of the information necessary to estimate the parameters of the distribution from the data [35, 36]). Members of the exponential family include most distributions of engineering interest, including Gaussian, Poisson, binomial, uniform, Rayleigh, and others. For exponential families, the E-step can be written as

$$Q(\theta|\theta^{[k]}) = E[\log b(\mathbf{x})|\mathbf{y}, \theta^{[k]}] + \mathbf{c}(\theta)^T E[\mathbf{t}(\mathbf{x})|\mathbf{y}, \theta^{[k]}] - \log a(\theta)$$

Let $\mathbf{t}^{[k+1]} = E[\mathbf{t}(\mathbf{x})|\mathbf{y}, \theta^{[k]}]$. As a conditional expectation is an estimator, $\mathbf{t}^{[k+1]}$ is an estimate of the sufficient statistic (The



4. Single-microphone ANC system.



5. Processor block diagram of the ANC system.

EM algorithm is sometimes called the estimation/maximization algorithm because, for exponential families, the first step is an estimator. It has also been called the expectation/modification algorithm [9]). In light of the fact that the M-step will be maximizing

$$E[\log b(\mathbf{x})|\mathbf{y},\theta^{[k]}] + \mathbf{c}(\theta)^T \mathbf{t}^{[k+1]} - \log a(\theta)$$

with respect to θ and that $E[\log b(\mathbf{x})|\mathbf{y},q^{[k]}]$ does not depend upon θ, it is sufficient to write:

E-step Compute

$$\mathbf{t}^{[k+1]} = E[\mathbf{t}(\mathbf{x})|\mathbf{y},\theta^{[k]}]. \tag{12}$$

M-step Compute

$$\theta^{[k+1]} = \arg\max_{\theta} \mathbf{c}(\theta)^T \mathbf{t}^{[k+1]} - \log a(\theta). \tag{13}$$

The EM algorithm may be diagrammed starting from an initial guess of the parameter $\theta^{[0]}$ as follows:

$$\theta^{[0]} \xrightarrow{\text{E-step}} \mathbf{t}^{[1]} \xrightarrow{\text{M-step}} \theta^{[1]} \xrightarrow{\text{E-step}} \mathbf{t}^{[2]} \xrightarrow{\text{M-step}} \dots,$$

The EM algorithm has the advantage of being simple, at least in principle; actually computing the expectations and performing the maximizations may be computationally taxing. In addition, as discussed in the next section, every iteration of the EM algorithm increases the likelihood function until a point of (local) maximum is reached. Unlike other optimization techniques, it is not necessary to compute gradients or Hessians, nor is it necessary to worry about setting step-size parameters, as algorithms such as gradient descent require.

Convergence of the EM Algorithm

For every iterative algorithm, the question of convergence needs to be addressed: does the algorithm come finally to a solution, or does it iterate *ad nauseum*, ever learning but never coming to a knowledge of the truth? For the EM algorithm, the convergence may be stated simply: at every iteration of the EM algorithm, a value of the parameter is computed so that the likelihood function does not decrease. That is, at every iteration the estimated parameter provides an increase in the likelihood function increases until a local maximum is achieved, at which point the likelihood function cannot increase (but will not decrease). Box 3 contains a more precise statement of this convergence for the general EM algorithm.

Despite the convergence theorem in Box 3, there is no guarantee that the convergence will be to a *global* maximum. For likelihood functions with multiple maxima, convergence will be to a local maximum which depends on the initial starting point $\theta^{[0]}$.

The convergence rate of the EM algorithm is also of interest. Based on mathematical and empirical examinations, it has been determined that the convergence rate is usually slower than the quadratic convergence typically available with a Newton's-type method [4]. However, as observed by Dempster [1], the convergence near the maximum (at least for exponential families) depends upon the eigenvalues of the Hessian of the update function M, so that rapid convergence may be possible. In any event, even with potentially slow convergence there are advantages to EM algorithms over Newton's algorithms. In the first place, no Hessian needs to be computed. Also, there is no chance of "overshooting" the target or diverging away from the maximum. The EM algorithm is guaranteed to be stable and to converge to an ML estimate. Further discussion of convergence appears in [37, 38].

Applications of the EM Algorithm

In this section several applications of the EM algorithm to problems of signal processing interest are presented to illustrate the computations required in the steps of the algorithm and also to demonstrate the breadth of applications to which it may be applied. The example in ET image reconstruction section and the previous introductory example illustrate the case in which the densities are members of the exponential family. The other examples in this section treat densities that are not in the exponential family, so the more general statement of the EM algorithm must be applied. The focus of the examples is on the EM algorithm; assumptions and details of the systems involved are therefore not presented. The interested reader is encouraged to examine the references for details.

Introductory Example, Revisited

The multinomial distribution of the introductory example is a member of the exponential family with $\mathbf{t}(\mathbf{x}) = \mathbf{x}$:

Box 2: Combination and conditional expectations multinomials

Let X_1,X_2,X_3 have a multinomial distribution with class probabilities (p_1,p_2,p_3), so

$$P(X_1=x_1,X_2=x_2,X_3=x_3)=\frac{(x_1+x_2+x_3)!}{x_1!x_2!x_3!}p_1^{x_1}p_2^{x_2}p_3^{x_3}$$

This multinomial in n outcomes can be combined to form a multinomial in $(n-1)$ outcomes. Let $Y=X_1+X_2$. The probability $P(Y,X_3)$ can be determined as follows:

$$\begin{aligned}P(X_1+X_2=y,X_3=x_3) &= \sum_{i=10}^{y}P(X_1=i,X_2=y-1,X_3=x_3)\\ &=\frac{(y+x_3)!}{y!x_3!}p_3^{x_3}\sum_{i=10}^{y}\frac{y!}{i!(y-i!)}p_1^i p_2^{y-i}\\ &=\frac{(y+x_3)!}{y!x_3!}(p_1+p_2)^y p_3^{x_3}\end{aligned}$$

where the last step follows from the binomial theorem. So (X_1+X_2,X_3) is binomial with class probabilities (p_1+p_2,p_3). This generalizes by induction to other multinomials.

To compute the conditional expectation $E[X_1|Y=y]$, it is first necessary to determine the conditional probability, $P(X_1=x_1|Y=y)=P(X_1=x_1|X_1+X_2=y)$. The conditional probability can be written as

$$\begin{aligned}P(X_1=x_1|Y=y) &= \frac{P(X_1=x_1,Y=y)}{P(Y=y)}\\ &=\frac{P(X_1=x_1,X_2=y-x_1)}{P(Y=y)}\end{aligned}$$

where the numerator probability is over the trinomial, out of $n=x_1+x_2+x_3$ trials, and the denominator probability is over the binomial out of n trials. Then

$$P(X_1=x_1|Y=y)=\frac{y!}{x_1!(y-x_1)!}p_1^{x_1}p_2^{y-x_1}\frac{1}{(p_1+p_2)^y}.$$

The conditional expectation is then

$$\begin{aligned}E[X_1|X_1+X_2=y] &= \sum_{x_1=0}^{y}x_1\frac{y!}{x_1!(y-x_1)!}p_1^{x_1}p_2^{x_2}\frac{1}{(p_1-p_2)^y}\\ &=y\frac{p_1}{p_1-p_2}\end{aligned}\quad(37)$$

Similarly it can be shown that

$$E[X_2|X_2+X_2=y]=y\frac{p_2}{p_1+p_2}$$

Computations are similar for Poisson random variables.

$$f(x_1,x_2,x_3|p)=\left(\frac{n!}{x_1!x_2!x_3!}\right)\exp\left[\left[\log\frac{x_1/4}{\frac{1}{2}-\frac{p}{4}},\log\frac{\frac{1}{4}+\frac{p}{4}}{\frac{1}{2}-\frac{p}{4}}\right]\begin{bmatrix}x_1\\x_2\end{bmatrix}\right]\left(\frac{1}{2}-\frac{p}{4}\right)^n.$$

The E-step consists simply of estimating the underlying data, given the current estimate and the data. This is followed by a straightforward maximization.

ET Image Reconstruction

In ET [7], tissues within a body are stimulated to emit photons. These photons are detected by detectors surrounding the tissue. For purposes of computation the body is divided into B boxes. The number of photons generated in each box is denoted by $n(b)$, $b=1,2,...,B$. The number of photons detected in each detector is denoted by $y(d)$, $d=1,2,...,D$, as shown in Fig. 3. Let $\mathbf{y}=[y(1),y(2),...,y(d)]$ denote the vector of observations.

The generation of the photons from box b can be described as a Poisson process with mean $\lambda(b)$, i.e.,

$$f(n|\lambda(b))=P(n(b)=n|\lambda(b))=e^{-\lambda(b)}\frac{\lambda(b)^n}{n!}.$$

The parameter $\lambda(b)$ is a function of the tissue density so that by estimating the parameters $\lambda(b)$ in each box it is possible to construct an image of the body. The boxes are assumed to be independent of each other. Let the set of unknown parameters be denoted by $\lambda=\{\lambda(1),\lambda(2),...,\lambda(B)\}$.

A photon emission from box b is detected in tube d with probability $p(b,d)$, and it may be assumed that all emitted photons are detected by some detector, so that

$$\sum_{d=1}^{D}p(b,d)=1.\quad(14)$$

Based upon the geometry of the sensors and the body it is possible to determine $p(b,d)$. The detector variables $y(d)$ are Poisson distributed,

$$f(y|\lambda(d))=P(y(d)=y)=e^{-\lambda(d)}\frac{\lambda(d)^y}{y!}$$

and it can be shown that

$$\lambda(d)=E[y(d)]=\sum_{b=1}^{B}\lambda(b)p(b,d).$$

Let $x(b,d)$ be the number of emissions from box b detected in detector d and let $\mathbf{x}=\{x(b,d),b=1,...,B,\ d=1,...,D\}$. For any given set of detector data $\{y(d)\}$, there are many different ways that the photons could have been generated. There is thus a many-to-one mapping from $x(b,d)$ to $y(d)$, and $\mathbf{x}$ constitutes the complete data set. Each variable of the complete data $x(b,d)$ is Poisson with mean

$$\lambda(b,d) = \lambda(b)p(b,d). \tag{15}$$

Assuming that each box generates independently of every other box and that the detectors operate independently, the likelihood function of the complete data is

$$l_x(\lambda) = f(x|\lambda) = \prod_{\substack{b=1,\ldots,B \\ d=1,\ldots,D}} e^{-\lambda(b,d)} \frac{\lambda(b,d)^{x(b,d)}}{x(b,d)!} \tag{16}$$

and, using Eq. (15), the log-likelihood function is

$$L_y(\lambda) = \log l_y(\lambda) = \sum_{\substack{b=1,\ldots,B \\ d=1,\ldots,D}} -\lambda(b)p(b,d) + x(b,d)\log\lambda(b) + x(b,d)\log p(b,d) - \log x(b,d)! \tag{17}$$

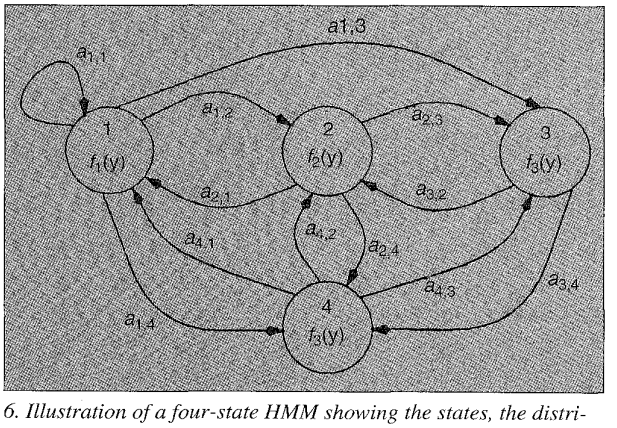
Application of the EM algorithm is straightforward. Poisson distributions are in the exponential family. The sufficient statistics for the distribution are the data, $\mathbf{t}(\mathbf{x}) = \mathbf{x}$. Let $\lambda^{[k]}$ be the estimate of the parameters at the kth iteration and let $\mathbf{x}^{[k]}(b,d)$ be the estimate of the complete data. For the E-step, compute

$$x^{[k+1]}(b,d) = E[x(b,d)|\mathbf{y},\lambda^{[k]}] = E[x(b,d)|y(d),\lambda^{[k]}]$$

where the latter equality follows since each box is independent. Since $x(b,d)$ is Poisson with mean $\lambda^{[k]}(b,d)$ and $y(d) = \sum_{b=1}^{B} x(b,d)$ is Poisson with mean $\lambda^{[k]}(d) = \sum_{b=1}^{B} \lambda^{[k]}(b,d)$, the conditional expectation may be computed (using techniques similar to those in Box 2)

$$x^{[k+1]}(b,d) = \frac{y(d)\lambda^{[k]}(b,d)}{\sum_{b'=1}^{B} \lambda^{[k]}(b',d)}, \quad b = 1,2,\ldots,B, d = 1,2,\ldots,D \tag{18}$$

For the M-step, $x^{[k+1]}(b,d)$ is used in the likelihood function (17), which is maximized with respect to $\lambda(b)$:



6. Illustration of a four-state HMM showing the states, the distributions in each state, and some probabilistic transitions between the states.

$$0 = \frac{\partial}{\partial\lambda(b)} \sum_{\substack{b=1,\ldots,B \\ d=1,\ldots,D}} -\lambda(b)p(b,d) + x^{[k+1]}(b,d)\log\lambda(b) + x^{[k+1]}(b,d)\log p(b,d) - \log x^{[k+1]}(b,d)! \tag{19}$$

$$\rightarrow \lambda^{[k+1]}(b) = \sum_{d=1}^{D} x^{[k+1]}(b,d)p(b,d)$$

where Eq. (14) has been used.

Equations (18) and (19) may be iterated until convergence. The overhead of storing $x^{[k+1]}(b,d)$ at each iteration may be eliminated by substituting Eq. (18) into Eq. (19) using Eq. (15), much as was done in the introductory example. This gives

$$\lambda^{[k+1]}(b) = \lambda^{[k]}(b) \sum_{d=1}^{D} \frac{y(d)p(b,d)}{\sum_{b'=1}^{B} \lambda^{[k]}(b')p(b',d)}.$$

Active Noise Cancellation (ANC)

Active noise cancellation is accomplished by measuring a noise signal and using a speaker driven out of phase with the noise to cancel it. In many traditional ANC techniques, two microphones are used in conjunction with an adaptive filter to provide cancellation (see, e.g., [39, 40]). Using the EM algorithm, ANC may be achieved with only one microphone [41]. The physical system is depicted in Fig. 4, with a block diagram for the ANC in Fig. 5.

The signal to be canceled is modeled as the output of an all-pole filter,

$$s(t) = -\sum_{k=1}^{p} a_k s(t-k) + \sigma_s u(t) = -\mathbf{s}_{p-1}^{T}(t-1)\mathbf{a} + \sigma_s u(t)$$

where

$$\mathbf{s}_p(t) = [s(t-p), s(t-p+1), \ldots, s(t)]^T,$$

and $u(t)$ is a white, unit-variance, zero-mean Gaussian process. The signal $r(t)$ is generated by the processor and corresponds to the input of the speaker; the delay z^{-M} is the delay from the speaker to the microphone. The signal $\sigma_\epsilon v(t)$ models the measurement error at the microphone. According to Fig. 5, the input to the processor can be written as

$$y(t) = s(t) + \sigma_\epsilon v(t);$$

we assume that $v(t)$ is a unit-variance, white Gaussian process. The set of unknown parameters is $\theta = [\mathbf{a}^T, \sigma_s^2, \sigma_\epsilon^2]^T$.

A block of N measurements is used for processing. The observed data vector is

$$\mathbf{y} = [y(1), y(2), \ldots, y(N)]^T;$$

these observations span a set of autoregressive samples given by

$\mathbf{s} = [s(1-p), s(2-p), ..., s(N)]^T$.

The complete data set is $\mathbf{x} = [\mathbf{y}^T, \mathbf{s}^T]^T$. If we knew $\mathbf{s}$, estimation of the AR parameters would be straightforward using familiar spectrum estimation techniques.

The likelihood function for the complete data is

$$f(\mathbf{x}|\theta) = f(\mathbf{y},\mathbf{s}|\theta) = f(\mathbf{y}|\mathbf{s},\theta)\, f(\mathbf{s}|\theta).$$

The conditioning step provides important leverage because it is straightforward to determine $f(y|\mathbf{s},\theta)$. The conditioning can be further broken down as

$$f(\mathbf{x}|\theta) = f(\mathbf{y}|\mathbf{s},\theta)\, f(s(1), s(2),..., s(N)|\mathbf{s}_{p-1}(0),\theta) f(\mathbf{s}_{p-1}(0)|\theta).$$

Then

$$f(\mathbf{y}|\mathbf{s},\theta) = \frac{1}{(2\pi\sigma_\epsilon^2)^{N/2}} \exp\left[-\frac{1}{2\sigma_\epsilon^2}\sum_{t=1}^{N}(y(t)-s(t))^2\right]$$

and (see [42, page 187])

$$f(s(1),s(2),...,s(N)|\mathbf{s}_{p-1}(0),\theta) = \frac{1}{(2\pi\sigma_s^2)^{(N-p)/2}} \exp\left[-\frac{1}{2\sigma_s^2}\sum_{t=1}^{N}(s(t)+\mathbf{a}^T\mathbf{s}_{p-1}(t))^2\right].$$

The E-step may be computed as

$$E[\log f(\mathbf{x}|\theta)|\mathbf{y},\theta^{[k]}] = \log f(\mathbf{s}_{p-1}(0)|\theta) - N\log\sigma_s - N\log\sigma_\epsilon$$
$$-\frac{1}{2\sigma_s^2}\sum_{t=1}^{N}\left[E[s^2(t)|\mathbf{y},\theta^{[k]}] + 2\mathbf{a}^T E[\mathbf{s}_{p-1}(t-1)s(t)|\mathbf{y},\theta]\right.$$
$$\left.+\mathbf{a}^T E[\mathbf{s}_{p-1}(t-1)\mathbf{s}_{p-1}(t-1)|\mathbf{y},\theta^{[k]}]\mathbf{a}\right]$$
$$-\frac{1}{2\theta_\epsilon^2}\sum_{t=1}^{N}\left[y^2(t) - 2y(t)E[s(t)|\mathbf{y},\theta^{[k]}] + E[s^2(t)|\mathbf{y},\theta^{[k]}]\right]$$

Taking the gradient with respect to $\mathbf{a}$ and derivatives with respect to σ_s and σ_ϵ to maximize yields

$$\mathbf{a}^{[k+1]} = -\left[\sum_{t=1}^{N} E[\mathbf{s}_{p-1}(t-1)\mathbf{s}_{p-1}^T(t-1)|\mathbf{y},\theta^{[k]}]\right]^{-1} \sum_{t=1}^{N} E[\mathbf{s}_{p-1}(t-1)s(t)|\mathbf{y},\theta^{[k]}] \tag{20}$$

$$(\sigma_s^2)^{[k+1]} = \frac{1}{N}\sum_{t=1}^{N} E[s^2(t)|\mathbf{y},\theta^{[k]}] + (\mathbf{a}^{[k]})^T \sum_{t=1}^{N} E[\mathbf{s}_{p-1}(t-1)s(t)|\mathbf{y},\theta^{[k]}] \tag{21}$$

$$(\sigma_\epsilon^2)^{[k+1]} = \frac{1}{N}\sum_{t=1}^{N}[y^2(t) - 2y(t)E[s(t)|\mathbf{y},\theta^{[k]}] + E[s^2(t)|\mathbf{y},\theta^{[k]}] \tag{22}$$

The expectations in Eqs. (20), (21), and (22) are first and second moments of Gaussians, conditioned upon observation which may be computed using a Kalman smoother. The variable $\mathbf{s}_p$ may be put into state-space form as

$$\mathbf{s}_p(t) = \Phi\mathbf{s}_p(t-1) + \mathbf{g}u(t)$$
$$y(t) = \mathbf{h}^T\mathbf{s}_p(t) + \sigma_\epsilon v(t)$$

where

$$\Phi = \begin{bmatrix} 0 & I \\ 0 & -\mathbf{a}^T \end{bmatrix},$$
$$\mathbf{g}^T = [0,0,...,0,\sigma_s]$$

and

Box 3: A convergence theorem for the EM algorithm

Let

$$k(\mathbf{x}|\mathbf{y},\theta) = \frac{f(\mathbf{x}|\theta)}{g(\mathbf{y}|\theta)}$$

and note that $k(\mathbf{x}|\mathbf{y},\theta)$ may be interpreted as a conditional density. Then the log-likelihood function $L_y(\theta) = \log g(\mathbf{y}|\theta)$ may be written as

$$L_y(\theta) = \log f(\mathbf{x}|\theta) - \log \kappa(\mathbf{x}|\mathbf{y},\theta).$$

Define

$$H(\theta'|\theta) = E[\log k(\mathbf{x}|\mathbf{y},\theta')|\,\mathbf{y},\theta].$$

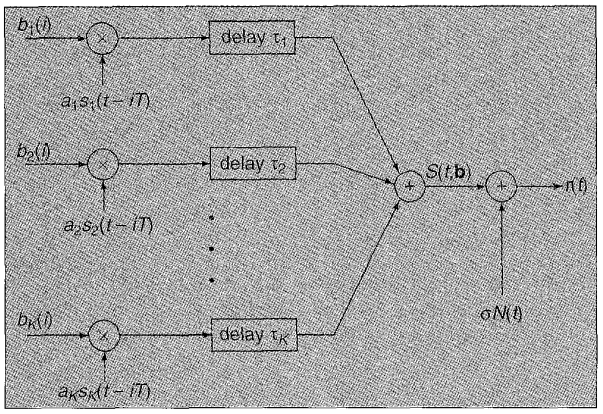
Let $M:\theta^{[k]} \rightarrow \theta^{[k+1]}$ represent the mapping defined by the EM algorithm in Eqs. (9), (10), so that $\theta^{[k+1]} = M(\theta^{[k]})$.

Theorem 1 *$L_y(M(\theta^{[k+1]})) \geq L_y(\theta)$, with equality if and only if*

$$Q(M(\theta)|\theta) = Q(\theta|\theta)$$
and
$$k(\mathbf{x}|\mathbf{y},M(\theta)) = k(\mathbf{x}|\mathbf{y},\theta).$$

That is, the likelihood function increases at each iteration of the EM algorithm, until the conditions for equality are satisfied and a fixed point of the iteration is reached. A proof of the theorem may be foundd in [1]. If θ^* is an ML parameter estimate, so that $L_y(\theta^*) \geq L_y(\theta)$ for all $\theta \in \Theta$, then $L_y(M(\theta^*)) = L_y(\theta^*)$. In other words, ML estimates are fixed points of the EM algorithm. Since the likelihood function is bounded (for distributions of practical interest), the sequence of parameter estimates $\theta^{[0]}, \theta^{[1]}, ..., \theta^{[k]}$ yields a bounded nondecreasing sequence $L_y(\theta^{[0]}) \leq L_y(\theta^{[1]}) \leq \cdots \leq L_y(\theta^{[k]})$ which must converge as $k \rightarrow \infty$.

The theorem falls short of proving that the fixed points of the EM algorithm are in fact ML estimates. The latter is true, under rather general conditions, but the proof is somewhat involved and is not presented here.



7. Representation of signals in a spread-spectrum multiple-access system.

$$\mathbf{h}^T = [0,...,0,1].$$

With an estimate of the parameters, the canceling signal $c(t+M)$ is obtained by estimating $\mathbf{s}(t+M)$ using $\mathrm{E}[\mathbf{s}_p(t)|\mathbf{y},\theta]$ and $\theta^{[i]}$.

HMMs

The hidden Markov model is a stochastic model of a process that exhibits features that change over time. It has been applied in a broad variety of sequential pattern-recognition problems such as speech recognition and handwriting recognition [9, 43]. An overview appeared in *Signal Processing Magazine* in [44]. Detailed descriptions of HMMs and their application are given in [9, 10, 11].

A Markov chain is a stochastic model of a system that is capable of being in a finite number of states $\{1,2,...,S\}$. The current state of the system is denoted by s_t. The probability of transition from a state at the current (discrete) time t to any other state at time $t+1$ depends only on the current state, and not on any prior states:

$$P(s_{t+1} = j \mid s_t = i, s_{t-1} = i_1, ...) = P(s_{t+1} = j \mid s_t = i).$$

It is common to express the transition probabilities as a matrix A with elements $P(s_{t+1} = j | s_t = i) = a_{ij}$. The initial state s_0 is chosen according to the probability

$$\pi = [P(s_0 = 1), ..., P(s_0 = S)]^T = [\pi_1,...,\pi_S]^T$$

In each state at time t, s_t, a (possibly vector) random variable is $\mathbf{Y}_t$ selected according to the density $f(\mathbf{Y}_t = \mathbf{y}_t | s_t = i) = f_{s_t}(y)$, as shown in Fig. 6. The variable $\mathbf{y}$ is observed, but the underlying state is not, hence the name *hidden* Markov model. The set of densities $f_1, f_2, ..., f_S$ is denoted as $f_{\{s\}}$. The triple $(A,\pi,f_{\{s\}})$ defines the HMM.

The HMM operates as follows: an initial state s_0 is chosen according to the probability law π. A succeeding state s_1 is chosen according to the Markov probability transition A. An output y_1 is chosen according to f_{s1}. Then a new state is chosen, and the process continues.

Let the elements of the HMM be parameterized by θ, i.e., there is a mapping $\theta \rightarrow (A(\theta),\pi(\theta),f_{\{s\}}(\cdot|\theta))$. The mapping is assumed to be appropriately smooth. In practice, the initial probability and transition probabilities are some of the elements of θ. The parameter estimation problem for an HMM is this: given a sequence of observations, $\mathbf{y} = \{\mathbf{y}_1,\mathbf{y}_2,...,\mathbf{y}_T\}$, determine the parameter θ which maximizes the likelihood function

$$\begin{aligned} l_y(\theta) &= f(\mathbf{y}_1,\mathbf{y}_2,...,\mathbf{y}_T|(A(\theta),\pi(\theta),f_{\{s\}}(\theta))) \\ &= \sum_{s_0,s_1,...,s_T=1}^{S} \pi_{s_0}(\theta) a_{s_0,s_1}(\theta) f_{s_1}(y_1|\theta) a_{s_1,s_2}(\theta) \\ &\quad f_{s_2}(y_2|\theta)\cdots a_{s_{T-1},s_T}(\theta) f_{s_T}(y_T|\theta). \end{aligned} \tag{23}$$

That is, determine the initial state probabilities and the transition probabilities, as well as any parameters of the density functions which maximize the likelihood function (23). From the complicated structure of (23), it is clear that this is a complicated maximization problem. The EM algorithm, however, provides the power necessary to compute without difficulty.

Let $\mathbf{s} = [s_0,s_1,s_2,...,s_T]^T$ be a vector of the (unobserved) states. The complete data vector can be expressed as $\mathbf{x} = (\mathbf{y},\mathbf{s})$. The pdf of the complete data can be written as

$$f(\mathbf{x}|\theta) = f(\mathbf{y},\mathbf{s}|\theta) = f(\mathbf{y}|\mathbf{s},\theta) f(\mathbf{s}|\theta) \tag{24}$$

This factorization, with the pdf of the observation conditioned upon the unknown state sequence and the distribution of the unknown state sequence, turns out to be the key step in the application of the EM algorithm.

Because of the Markov structure of the state, the state probabilities in Eq. (24) may be written

$$f(\mathbf{s}|\theta) = \pi_{s_0}(\theta) \prod_{t=1}^{T} a_{s_{t-1},s_t}(\theta) \tag{25}$$

The pdf of the observations, conditioned upon the unobserved states, factors as

$$f(\mathbf{y}|\mathbf{s},\theta) = \prod_{t=1}^{T} f(\mathbf{y}_t|s_t,\theta) \tag{26}$$

We will assume that the density in each state is Gaussian with known diagonal covariance and unknown mean, μ_s. (Many other distributions are possible, e.g., discrete selection, Poisson, exponential, or Gaussian with unknown mean and variance [45].) Then

$$f(\mathbf{y}_t|s_t,\theta) = \frac{1}{2\pi\sigma^2} \exp\left[-\frac{1}{2\sigma^2}(\mathbf{y}_t - \mu_{s_t}(\theta))^T(\mathbf{y}_t - \mu_{s_t}(\theta))\right] \tag{27}$$

Let $\mathcal{S} = \{1,2,...,S\}^{T+1}$ denote the set of all possible state sequences, including the initial state s_0. In the E-step

$$Q(\theta|\theta^{[k]}) = E[\log f(\mathbf{y},\mathbf{s}|\theta)|\mathbf{y},\theta],$$

since the expectation is conditioned upon the observations, the only random component comes from the state variable. The E-step can thus be written as

$$Q(\theta|\theta^{[k]}) = \sum_{\mathbf{s}\in S} f(\mathbf{s}|\mathbf{y},\theta^{[k]}) \log[f(\mathbf{y}|\mathbf{s},\theta) f(\mathbf{s},\theta)]$$

The conditional probability is

$$f(\mathbf{s}|\mathbf{y},\theta^{[k]}) = \frac{f(\mathbf{y}|\mathbf{s},\theta^{[k]}) f(\mathbf{s}|\theta^{[k]})}{g(\mathbf{y}|\theta^{[k]})}. \tag{28}$$

Substituting from Eqs. (27) and (28),

$$Q(\theta|\theta^{[k]}) = \frac{1}{g(\mathbf{y}|\theta^{[k]})} \sum_{\mathbf{s}\in S} f(\mathbf{y}|\mathbf{s},\theta^{[k]}) f(\mathbf{s}|\theta^{[k]}) \tag{29}$$
$$\left[\sum_{t=1}^{T}\left(-\log(2\pi\sigma^2 - \frac{1}{2\sigma^2}(\mathbf{y}_t - \mu_{s_t}(\theta))^T(\mathbf{y}_t - \mu_{s_t}(\theta))\right)\right.$$
$$\left. + \log \pi_{s_0}(\theta) + \sum_{t=1}^{T} \log a_{s_{t-1},s_t}(\theta)\right]$$

The updated parameters are then obtained by the M-step. For the means of the pdfs,

$$\mu_s^{[k+1]} = \arg\max_{\mu_s} Q(\theta|\theta^{[k]})$$

The maximizations may be accomplished by differentiating and equating the result to zero and solving for the appropriate argument. For the mean, the result is

$$\mu_s^{[k+1]} = \frac{\Sigma_{\mathbf{s}\in S} f(\mathbf{y}|\mathbf{s},\theta^{[k]}) f(\mathbf{s}|\theta^{[k]}) \Sigma_{t:s_t=s} \mathbf{y}_t}{\Sigma_{\mathbf{s}\in S} f(\mathbf{y}|\mathbf{s},\theta^{[k]}) f(\mathbf{s}|\theta^{[k]}) \Sigma_{t:s_t=s} 1}$$

Efficient algorithms for computing this expression have been developed based upon forward and backward inductive computation (dynamic programming or the Viterbi algorithm); see e.g. [10, 11].

The Markov chain parameters π_i and $a_{i,j}$ may also be obtained by maximizing Eq. (29) with constraints to preserve the probabilistic nature of the parameters:

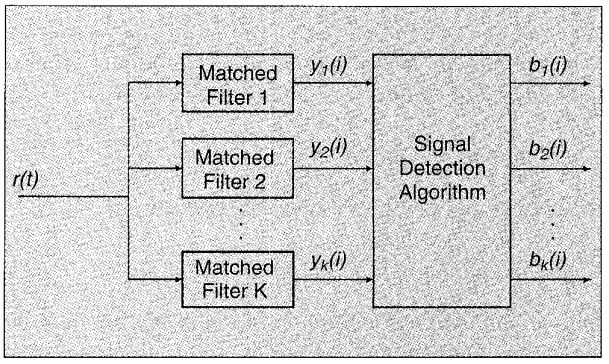
$$\pi_i^{[k+1]} = \arg\max_{\pi_i} Q(\theta|\theta^{[k]}) \text{ subject to } \sum_{i=1}^{S} \pi_i = 1,\ \pi_i \geq 0$$

$$a_{i,j}^{[k+1]} = \arg\max_{a_{i,j}} Q(\theta|\theta^{[k]}) \text{ subject to } \sum_{j=1}^{S} a_{i,j}^{[k-1]} = 1,\ a_{i,j} \geq 0$$

This may be accomplished using Lagrange multipliers. Then the condition

$$\frac{\partial}{\partial \pi_s} Q(\theta|\theta^{[k]}) + \lambda \sum_{i=1}^{S} \pi_i = 0$$

(with λ a Lagrange multiplier) leads to



8. Multiple-access receiver matched-filter bank.

$$\pi_i^{[k+1]} = \frac{\Sigma_{\mathbf{s}\in S,\, s_0=i} f(\mathbf{y}|\mathbf{s},\theta^{[k]}) f(\mathbf{s}|\theta^{[k]})}{\Sigma_{\mathbf{s}\in S} f(\mathbf{y}|\mathbf{s},\theta) f(\mathbf{s}|\theta)},$$

and similarly,

$$a_{i,j}^{[k+1]} = \frac{\Sigma_{\mathbf{s}\in S} f(\mathbf{y}|\mathbf{s},\theta^{[k]}) f(\mathbf{s}|\theta^{[k]}) \Sigma_{s_{t-1}=i,s_t=j} 1}{\Sigma_{\mathbf{s}\in S} f(\mathbf{y}|\mathbf{s},\theta^{[k]}) f(\mathbf{s}|\theta^{[k]}) \Sigma_{s_{t-1}=i} 1}.$$

Spread-Spectrum Multi-User Communication

In direct-sequence spread-spectrum multiple-access (SSMA) communications, all users in a channel transmit simultaneously, using quasi-orthogonal spreading codes to reduce the inter-user interference [46]. The system block diagram is shown in Fig. 7. A signal received in a K-user system through a Gaussian channel may be written as

$$r(t) = S(t,\mathbf{b}) + \sigma N(t)$$

where $N(t)$ is unit-variance, zero-mean, white Gaussian noise and

$$S(t,\mathbf{b}) = \sum_{k=1}^{K} a_k \sum_{i=-m}^{M} b_k(i) s_k(t - iT - \tau_k)$$

is the composite signal from all K transmitters. Here a_k is the amplitude of the kth transmitted signal (as seen at the receiver), $\mathbf{b}$ represents the symbols of all the users, $b_k(i)$ is the ith bit of the kth user, τ_k is the channel propagation delay for the kth user, and $s_k(t)$ is the signaling waveform of the kth user including the spreading code. For this example, coherent reception of each user is assumed so that the amplitudes are real.

At the receiver the signal is passed through a bank of matched filters, with a filter matched to the spreading signal of each of the users, as shown in Fig. 8. (This assumes that synchronization for each user has been obtained.) The set of matched filter outputs for the ith bit interval is

$$\mathbf{y}(i) = [y_1(i), y_2(i), \ldots, y_K(i)]^{TD}.$$

Because the interference among the users is similar to intersymbol interference, optimal detection requires dealing with the entire sequence of matched filter vectors

$$\mathbf{y} = [\mathbf{y}(-M)^T, \mathbf{y}(-M+1)^T, \ldots, \mathbf{y}(M)^T]^T.$$

For a Gaussian channel, it may be shown that

$$\mathbf{y} = H(\mathbf{b})\mathbf{a} + \mathbf{z}, \tag{30}$$

where $H(\mathbf{b})$ depends upon the correlations between the spreading signals and the bits transmitted and $\mathbf{z}$ is non-white, zero-mean Gaussian noise. The likelihood function for the received sequence may be written as (see [47])

$$f(\mathbf{y}|\mathbf{a},\mathbf{b}) = c\exp\left[\frac{1}{2\sigma_2}(2\mathbf{a}^T R(\mathbf{b})\mathbf{y} - \mathbf{a}^T S(\mathbf{b})\mathbf{a})\right] \tag{31}$$

where $R(\mathbf{b})$ and $S(\mathbf{b})$ depend upon the bits and correlations and c is a constant that makes the density integrate to 1. Note that even though the noise is Gaussian, which is in the exponential family, the overall likelihood function is not Gaussian because of the presence of the random bits — it is actually a mixture of Gaussians. For the special case of only a single user the likelihood function becomes

$$f(\mathbf{y}|\mathbf{a},\mathbf{b}) = c\exp\left[\frac{1}{2\sigma^2}(2a_1\sum_{i=-M}^{M} b_1(i)y_1(i) - a_1^2)\right].$$

What is ultimately desired from the detector is the set of bits for each user. It has been shown [46] that the inter-user interference degrades the probability of error very little, provided that sophisticated detection algorithms are employed after the matched filters. However, most of the algorithms that have been developed require knowledge of the amplitudes of each user [48]. Therefore, in order to determine the bits reliably, the amplitude of each user must also be known. Seen from the point of view of amplitude estimation, the bits are unknown nuisance parameters. (Other estimation schemes relying on decision feedback may take a different point of view.)

If the bits were known, an ML estimate of the amplitudes could be easily obtained: $\mathbf{a}_{ml} = S(\mathbf{b})^{-1} R(\mathbf{b})\mathbf{y}$. Lacking the bits, however, more sophisticated tools for obtaining the amplitudes must be applied as a precursor to detecting the bits. One approach to estimating the signal amplitudes is the EM algorithm [47]. For purposes of applying the EM algorithm, the complete data set is $\mathbf{x} = \{\mathbf{y}, \mathbf{b}\}$ and the parameter set is $\theta = \mathbf{a}$. To compute the expectations in the E-step, it is assumed that the bits are independent and equally likely ± 1.

The likelihood function of the complete data is

$$f(\mathbf{x}|\mathbf{a}) = f(\mathbf{y},\mathbf{b}|\mathbf{a}) = f(\mathbf{y}|\mathbf{b},\mathbf{a})f(\mathbf{b}|\mathbf{a}). \tag{32}$$

This conditioning is similar to that of Eqs. (19) and (24): the complete-data likelihood is broken into a likelihood of the observation, conditioned upon the unobserved data times a likelihood of the unobserved data. From Eq. (31), $f(\mathbf{y}|\mathbf{b},\mathbf{a})$ is Gaussian. To compute the E-step

$$E[\log f(\mathbf{x}|\mathbf{a})|\mathbf{y},\mathbf{a}^{[k]}] = \sum_{\mathbf{b}\in\{\pm 1\}^{(M+1)K}} f(\mathbf{b}|\mathbf{y},\mathbf{a}^{[k]})\log f(\mathbf{x}|\mathbf{a})$$

it is necessary to determine the conditional probability $f(\mathbf{b}|\mathbf{y},\mathbf{a}^{[k]})$.

It is revealing to consider a single-user system. In this case the log-likelihood function is

$$\log f(\mathbf{x}|a_1) = \frac{a_1}{\sigma^2}\sum_{i=-M}^{M} b_1(i)y_1(i) - \frac{a_1^2}{2\sigma^2}(2M+1) + \text{constant},$$

and the E-step becomes

$$E[\log f(\mathbf{x}|\mathbf{a})|\mathbf{y},\mathbf{a}^{[k]} = \sum_{i=-M}^{M}\sum_{b_1(i)\in\pm 1} f(b_1(i)|y_1(i)a_1^{[k]})\log f(x_1(i)|a_1) \tag{33}$$

The conditional probability required for the expectation is

$$\begin{aligned} f(b_1(i)|y_1(i),a_1^{[k]}) &= \frac{f(b_1(i),y_1(i)|a_1^{[k]})}{f(y_1(i)|a_1^{[k]})} = \frac{f(b_1(i),y_1(i)|a_1^{[k]})}{\Sigma_{b\in\pm 1} f(b,y_1(i)|a_1^{[k]}} \\ &= \frac{\exp\left[\frac{1}{2\sigma^2}(2a_1^{[k]}b_1(i)y_1(i) - a_1^{[k]2})\right]}{\Sigma_{b\in\pm 1}\exp\left[\frac{1}{2\sigma^2}(2a_1^{[k]}by_1(i) - a_1^{[k]2})\right]} \\ &= \frac{\exp\left[\frac{1}{\sigma^2}(a_1^{[k]}b_1(i)y_1(i))\right]}{\cosh(y_1(i)a_1^{[k]}/\sigma^2)} \end{aligned} \tag{34}$$

Substituting Eq. (34) into Eq. (33) yields

$$E[\log f(\mathbf{x}|a_1)|\mathbf{y},a_1^{[k]}] = \frac{a_1}{\sigma^2}\sum_{i=-M}^{M} y_1(i)\tanh(a_1^{[k]}y_1(i)/\sigma^2) - \frac{a_1^2}{2\sigma^2}(2M+1) + \text{constant} \tag{35}$$

Conveniently, Eq. (35) is quadratic in a_1 and the M-step is easily computed by differentiating Eq. (35) with respect to a_1, giving

$$a_1^{[k+1]} = \frac{1}{2M+1}\sum_{i=-M}^{M} y_1(i)\tanh(a_1^{[k]}y_1(i)/\sigma^2) \tag{36}$$

Equation (36) gives the update equation for the amplitude estimate, which may be iterated until convergence. For multiple-users, the E-step and M-step are structurally similar, but more involved computationally [47].

Summary

The EM algorithm may be employed when there is an underlying set with a known distribution function that is observed by means of a many-to-one mapping. If the distribution of the underlying complete data is exponential, the EM algorithm

may be specialized as in Eqs. (12) and (13). Otherwise, it will be necessary to use the general statement of the EM algorithm (Eqs. (9) and (10)). In many cases, the type of conditioning exhibited in Eqs. (19), (24) or (32) may be used: the observed data is conditioned upon data not observed so that the likelihood function may be computed. In general, if the complete data set is $\mathbf{x} = (\mathbf{y},\mathbf{z})$ for some unobserved $\mathbf{z}$, then

$$E[\log f(\mathbf{x}|\theta)|\mathbf{y},\theta^{[k]}] = \int f(\mathbf{z}|\mathbf{y},\theta^{[k]}) \log f(\mathbf{x}|\theta)\, d\mathbf{z},$$

since, conditioned upon $\mathbf{y}$ the only random component of $\mathbf{x}$ is $\mathbf{z}$.

Analytically, the most difficult portion of the EM algorithm is the E-step. This is also often the most difficult computational step; for the general EM algorithm, the expectation must be computed over all values of the unobserved variables. There may be, as in the case of the HMM, efficient algorithms to ease the computation, but even these cannot completely eliminate the computational burden.

In most instances where the EM algorithm applies, there are other algorithms that also apply, such as gradient descent (see, e.g., [49]). As already observed, however, these algorithms may have problems of their own such as requiring derivatives or setting of convergence-rate parameters. Because of its generality and the guaranteed convergence, the EM algorithm is a good choice to consider for many estimation problems. Future work will include application in new and different areas, as well as developments to improve convergence speed and computational structure.

Todd K. Moon is Associate Professor at the Electrical and Computer Engineering Department and Center for Self-Organizing Intelligent Systems at Utah State University.

References

1. A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Statiscal Soc., Ser. B*, vol. 39, no.1, pp.1–38, 1977.

2. For an extensive list of references to papers describing applications of the EM algorithm, see http://www.engineering/usu.edu/Departments/ece/Publications/Moon on the World-Wide Web.

3. C. Jiang, "The use of mixture models to detect effects of major genes on quantitative characteristics in a plant-breeding experiment," *Genetics*, vol. 136, no. 1, pp. 383–394, 1994.

4. R. Redner and H.F. Walker, "Mixture densities, maximum-likelihood estimation and the EM algorithm (review)," *SIAM Rev.*, vol. 26, no. 2, pp. 195–237, 1984.

5. J. Schmee and G.J. Hahn, "Simple method for regression analysis with censored data," *Technometrics*, vol. 21, no. 4, pp. 417-432, 1979.

6. R.Little and D.Rubin, "On jointly estimating parameters and missing data by maximizing the complete-data likelihood," *Am. Statistn.*, vol. 37, no. 3, pp. 218–200, 1983.

7. L.A. Shepp and Y.Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE Med. Im.*, vol.1, pp. 113-122, October 1982.

8. D.L. Snyder and D.G. Politte, "Image reconstruction from list-mode data in an emission tomography system having time-of-flight measurements," *IEEE Nucl. S.*, vol. 30, no. 3, pp. 1843-1849, 1983.

9. L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *P. IEEE*, vol. 77, no. 2, pp. 257-286, 1989.

10. L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*. Prentice-Hall, 1993.

11. J.R. Deller, J.G. Proakis, and J.H.L. Hansen, *Discrete-Time Processing of Speech Signals*. Macmillan, 1993.

12. M. Segal and E. Weinstein, "Parameter estimation of continuous dynamical linear systems given discrete time observations," *P. IEEE*, vol.7 5, no. 5, pp. 727–729, 1987.

13. S. Zabin and H. Poor, "Efficient estimation of class-A noise parameters via the EM algorithm," *IEEE Trans. Info. T.*, vol. 37, no. 1, pp. 60–72, 1991.

14. A. Isaksson, "Identification of ARX models subject to missing data," *IEEE Auto C*, vol. 38, no. 5, pp. 813-819, 1993.

15. I. Ziskind and D. Hertz, "Maximum likelihood localization of narrowband autoregressive sources via the EM algorithm," *IEEE Trans. Sig. Proc.*, vol. 41, no. 8, pp. 2719-2724, 1993.

16. R. Lagendijk, J. Biemond, and D. Boekee, "Identification and restoration of noisy blurred images using the expectation-maximization algorithm," *IEEE Trans. ASSP*, vol. 38, no. 7, pp. 1180-1191, 1990.

17. A. Katsaggelos and K. Lay, "Maximum likelihood blur identification and image restoration using the algorithm," *IEEE Trans. Sig. Proc.*, vol. 39, no. 3, pp. 729-733, 1991.

18. A. Ansari and R. Viswanathan, "Application of EM algorithm to the detection of direct sequence signal in pulsed noise jamming," *IEEE Trans. Com.*, vol. 41, no. 8, pp. 1151-1154, 1993.

19. M. Feder, "Parameter estimation and extraction of helicopter signals observed with a wide-band interference," *IEEE Trans. Sig. Proc.*, vol. 41, no. 1, pp. 232–244, 1993.

20. G. Kaleh, "Joint parameter estimation and symbol detection for linear and nonlinear unknown channels," *IEEE Trans. Com.*, vol. 42, no. 7, pp. 2506-2413, 1994.

21. W. Byrne, "Alternating minimization and Boltzman machine learning," *IEEE Trans. Neural Net.*, vol. 3, no. 4, pp. 612-620, 1992.

22. M. Jordan and R. Jacobs, "Hierarchical mixtures of experts and the EM algorithm," *Neural Comp.*, vol. 6, no. 2, pp. 181-214, 1994.

23. R. Streit and T. Luginbuh, "ML training of probabilistic neural networks," *IEEE Trans. Neural Net.*, vol. 5, no. 5, pp. 764-783, 1994.

24. M. Miller and D. Fuhrmann, "Maximum likelihood narrow-band direction finding and the EM algorithm," *IEEE Trans. ASSP*, vol. 38, no. 9, pp. 1560-1577, 1990.

25. S. Vaseghi and P. Rayner, "Detection and suppression of impulsive noise in speech communication systems," *IEE Proc-I*, vol. 137, no. 1, pp. 38-46, 1990.

26. E. Weinstein, A. Oppenheim, M. Feder, and J. Buck, "Iterative and sequential algorithms for multisensor signal enhancement," *IEEE Trans. Sig. Proc.*, vol. 42, no. 4, pp. 846-859, 1994.

27. S.E. Bialkowski, "Expectation-maximization (EM) algorithm for regression, deconvolution, and smoothing of shot-noise limited data," *Journal of Chemometrics*, 1991.

28. C. Georghiades and D. Snyder, "The EM algorithm for symbol unsynchronized sequence detection," *IEEE Comun.*, vol. 39, no. 1, pp. 54-61, 1991.

29. N. Antoniadis and A. Hero, "Time-delay estimation for filtered Poisson processes using an EM-type algorithm," *IEEE Trans. Sig. Proc.*, vol. 42, no. 8, pp. 2112-2123, 1994.

30. M. Segal and E. Weinstein, "The cascade EM algorithm," *P. IEEE*, vol. 76, no. 10, pp. 1388-1390, 1988.

31. C. Gyulai, S. Bialkowski, G. S. Stiles, and L. Powers, "A comparison of three multi-platform message-passing interfaces on an expecation-maximization algorithm," in *Proceedings of the 1993 World Conference on Transputers*, pp. 451–464, 1993.

32. R.E. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Trans. Infor. Th.*, vol. 18, pp. 460-473, July 1972.

33. I. Csiszar and G. Tusnday, "Information geometry and alternating minimization procedures," *Statistics and Decisions, Supplement Issue 1*, 1984.

34. J.G. Proakis, *Digital Communications*. McGraw Hill, 3rd ed., 1995.

35. R.O. Duda and P.E. Hart, *Pattern Classification and Scene Analysis*. Wiley, 1973.

36. P.J. Bickel and K.A. Doksum, *Mathematical Statistics*. Holden-Day, 1977.

37. C. Wu, "On the convergence properties of the EM algorithm," *Ann. Statist.*, vol. 11, no. 1, pp. 95-103, 1983.

38. R.A. Boyles, "On the convergence of the EM algorithm," *J. Roy. Sta. B.*, vol.45, no. 1, pp. 47-50, 1983.

39. B. Widrow and S.D. Stearns, *Adaptive Signal Processing*. Prentice-Hall, 1985.

40. J.C. Stevens and K.K. Ahuja, "Recent advances in active noise control," *AIAA Journal*, vol. 29, no. 7, pp. 1058-1067, 1991.

41. M. Feder, A. Oppenheim, and E. Weinstein, "Maximum likelihood noise cancellation using the EM algorithm," *IEEE Trans. ASSP.*, vol. 37, no. 2, pp. 204-216, 1989.

42. S.M. Kay, *Modern Spectral Estimation*. Prentice-Hall, 1988.

43. Y. Singer, "Dynamical encoding of cursive handwriting," *Biol. Cybern.*, vol. 71, no. 3, pp. 227-237, 1994.

44. J. Picone, "Continuous speech recognition using hidden Markov models," *Signal Processing Magazine*, vol. 7, p. 41, July 1990.

45. L.E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," *Ann. Math. Stat.*, vol. 41, no. 1, pp. 164-171, 1970.

46. S. Verdu, "Optimum multiuser asymptotic efficiency," *IEEE Trans. Com.*, vol. COM-34, no. 9, pp. 890-896, September 1986.

47. H.V. Poor, "On parameter estimation in DS/SSMA formats," in *Proceedings of the International Conference on Advances in Communications and Control Systems*, 1988.

48. R. Lupas and S. Verdu, "Near-far resistance of multiuser detectors in asynchronous channels," *IEEE Trans. Comm*, vol. 38, pp. 496-508, April 1990.

49. A.V. Oppenheim, E. Weinsten, K. C. Zangi, M. Feder, and D. Gauger, "Single-sensor active noise cancellation based on the EM algorithm," *ICASSP*, 1992.

50. L. Scharf, *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*. Addison Wesley, 1991.

51. H.L.V. Trees, *Detection, Estimation, and Modulation Theory, Part I*. New York: John Wiley and Sons, 1968.