Statistical Science
2010, Vol. 25, No. 4, 476–491
DOI: 10.1214/09-STS312

The EM Algorithm and the Rise of Computational Biology

Xiaodan Fan, Yuan Yuan and Jun S. Liu

Abstract. In the past decade computational biology has grown from a cottage industry with a handful of researchers to an attractive interdisciplinary field, catching the attention and imagination of many quantitatively-minded scientists. Of interest to us is the key role played by the EM algorithm during this transformation. We survey the use of the EM algorithm in a few important computational biology problems surrounding the "central dogma" of molecular biology: from DNA to RNA and then to proteins. Topics of this article include sequence motif discovery, protein sequence alignment, population genetics, evolutionary models and mRNA expression microarray data analysis.

Key words and phrases: EM algorithm, computational biology, literature review.

1. INTRODUCTION

1.1 Computational Biology

Started by a few quantitatively minded biologists and biologically minded mathematicians in the 1970s, computational biology has been transformed in the past decades to an attractive interdisciplinary field drawing in many scientists. The use of formal statistical modeling and computational tools, the expectation–maximization (EM) algorithm, in particular, contributed significantly to this dramatic transition in solving several key computational biology problems. Our goal here is to review some of the historical developments with technical details, illustrating how biology, traditionally regarded as an empirical science, has come to embrace rigorous statistical modeling and mathematical reasoning.

Before getting into details of various applications of the EM algorithm in computational biology, we first explain some basic concepts of molecular biology. Three kinds of chain biopolymers are the central molecular building blocks of life: DNA, RNA and proteins. The DNA molecule is a double-stranded long sequence composed of four types of nucleotides (A, C, G and T). It has the famous double-helix structure, and stores the hereditary information. RNA molecules are very similar to DNAs, composed also of four nucleotides (A, C, G and U). Proteins are chains of 20 different basic units, called amino acids.

The genome of an organism generally refers to the collection of all its DNA molecules, called the chromosomes. Each chromosome contains both the protein (or RNA) coding regions, called genes, and noncoding regions. The percentage of the coding regions varies a lot among genomes of different species. For example, the coding regions of the genome of baker's yeast are more than 50%, whereas those of the human genome are less than 3%.

RNAs are classified into many types, and the three most basic types are as follows: messenger RNA (mRNA), transfer RNA (tRNA) and ribosomal RNA (rRNA). An mRNA can be viewed as an intermediate copy of its corresponding gene and is used as a template for constructing the target protein. tRNA is needed to recruit various amino acids and transport them to the template mRNA. mRNA, tRNA and amino acids work together with the construction machineries called ribosomes to make the final product, protein.

Xiaodan Fan is Assistant Professor in Statistics, Department of Statistics, the Chinese University of Hong Kong, Hong Kong, China (e-mail: xfan@sta.cuhk.edu.hk). Yuan Yuan is Quantitative Analyst, Google, Mountain View, California, USA (e-mail: yuany@google.com). Jun S. Liu is Professor of Statistics, Department of Statistics, Harvard University, 1 Oxford Street, Cambridge, Massachusetts 02138, USA (e-mail: jliu@stat.harvard.edu).

One of the main components of ribosomes is the third kind of RNA, rRNA.

Proteins carry out almost all essential functions in a cell, such as catalysation, signal transduction, gene regulation, molecular modification, etc. These capabilities of the protein molecules are dependent of their 3-dimensional shapes, which, to a large extent, are uniquely determined by their one-dimensional sequence compositions. In order to make a protein, the corresponding gene has to be *transcribed* into mRNA, and then the mRNA is *translated* into the protein. The "central dogma" refers to the concerted effort of transcription and translation of the cell. The *expression level* of a gene refers to the amount of its mRNA in the cell.

Differences between two living organisms are mostly due to the differences in their genomes. Within a multicellular organism, however, different cells may differ greatly in both physiology and function even though they all carry identical genomic information. These differences are the result of differential gene expression. Since the mid-1990s, scientists have developed microarray techniques that can monitor simultaneously the expression levels of all the genes in a cell, making it possible to construct the molecular "signature" of different cell types. These techniques can be used to study how a cell responds to different interventions, and to decipher gene regulatory networks. A more detailed introduction of the basic biology for statisticians is given by Ji and Wong (2006).

With the help of the recent biotechnology revolution, biologists have generated an enormous amount of molecular data, such as billions of base pairs of DNA sequence data in the GenBank, protein structure data in PDB, gene expression data, biological pathway data, biopolymer interaction data, etc. The explosive growth of various system-level molecular data calls for sophisticated statistical models for information integration and for efficient computational algorithms. Meanwhile, statisticians have acquired a diverse array of tools for developing such models and algorithms, such as the EM algorithm (Dempster, Laird and Rubin, 1977), data augmentation (Tanner and Wong, 1987), Gibbs sampling (Geman and Geman, 1984), the Metropolis–Hastings algorithm (Metropolis and Ulam, 1949; Metropolis et al., 1953; Hastings, 1970), etc.

1.2 The Expectation–Maximization Algorithm

The expectation–maximization (EM) algorithm (Dempster, Laird and Rubin, 1977) is an iterative method for finding the mode of a marginal likelihood function (e.g., the MLE when there is missing data) or a marginal distribution (e.g., the maximum a posteriori estimator). Let $\mathbf{Y}$ denote the observed data, $\mathbf{\Theta}$ the parameters of interest, and $\mathbf{\Gamma}$ the nuisance parameters or missing data. The goal is to maximize the function

$$p(\mathbf{Y}|\mathbf{\Theta}) = \int p(\mathbf{Y}, \mathbf{\Gamma}|\mathbf{\Theta})\, d\mathbf{\Gamma},$$

which cannot be solved analytically. A basic assumption underlying the effectiveness of the EM algorithm is that the complete-data likelihood or the posterior distribution, $p(\mathbf{Y}, \mathbf{\Gamma}|\mathbf{\Theta})$, is easy to deal with. Starting with a crude parameter estimate $\mathbf{\Theta}^{(0)}$, the algorithm iterates between the following Expectation (E-step) and Maximization (M-step) steps until convergence:

- E-step: Compute the Q-function:
$$Q(\mathbf{\Theta}|\mathbf{\Theta}^{(t)}) \equiv E_{\mathbf{\Gamma}|\mathbf{\Theta}^{(t)},\mathbf{Y}}[\log p(\mathbf{Y}, \mathbf{\Gamma}|\mathbf{\Theta})].$$
- M-step: Finding the maximand:
$$\mathbf{\Theta}^{(t+1)} = \arg\max_{\mathbf{\Theta}} Q(\mathbf{\Theta}|\mathbf{\Theta}^{(t)}).$$

Unlike the Newton–Raphson and scoring algorithms, the EM algorithm does not require computing the second derivative or the Hessian matrix. The EM algorithm also has the nice properties of monotone nondecreasing in the marginal likelihood and stable convergence to a local mode (or a saddle point) under weak conditions. More importantly, the EM algorithm is constructed based on the missing data formulation and often conveys useful statistical insights regarding the underlying statistical model. A major drawback of the EM algorithm is that its convergence rate is only linear, proportional to the fraction of "missing information" about $\mathbf{\Theta}$ (Dempster, Laird and Rubin, 1977). In cases with a large proportion of missing information, the convergence rate of the EM algorithm can be very slow. To monitor the convergence rate and the local mode problem, a basic strategy is to start the EM algorithm with multiple initial values. More sophisticated methods are available for specific problems, such as the "backup-buffering" strategy in Qin, Niu and Liu (2002).

1.3 Uses of the EM Algorithm in Biology

The idea of iterating between filling in the missing data and estimating unknown parameters is so intuitive that some special forms of the EM algorithm appeared in the literature long before Dempster, Laird and Rubin (1977) defined it. The earliest example on record is by McKendrick (1926), who invented a special EM algorithm for fitting a Poisson model to a cholera infection data set. Other early forms of the EM algorithm appeared in numerous genetics studies involving allele

frequency estimation, segregation analysis and pedigree data analysis (Ceppellini, Siniscalco and Smith, 1955; Smith, 1957; Ott, 1979). A precursor to the broad recognition of the EM algorithm by the computational biology community is Churchill (1989), who applied the EM algorithm to fit a hidden Markov model (HMM) for partitioning genomic sequences into regions with homogenous base compositions. Lawrence and Reilly (1990) first introduced the EM algorithm for biological sequence motif discovery. Haussler et al. (1993) and Krogh et al. (1994) formulated an innovative HMM and used the EM algorithm for protein sequence alignment. Krogh, Mian and Haussler (1994) extended these algorithms to predict genes in *E. coli* DNA data. During the past two decades, probabilistic modeling and the EM algorithm have become a more and more common practice in computational biology, ranging from multiple sequence alignment for a single protein family (Do et al., 2005) to genome-wide predictions of protein–protein interactions (Deng et al., 2002), and to single-nucleotide polymorphism (SNP) haplotype estimation (Kang et al., 2004).

As noted in Meng and Pedlow (1992) and Meng (1997), there are too many EM-related papers to track. This is true even within the field of computational biology. In this paper we only examine a few key topics in computational biology and use typical examples to show how the EM algorithm has paved the road for these studies. The connection between the EM algorithm and statistical modeling of complex systems is essential in computational biology. It is our hope that this brief survey will stimulate further EM applications and provide insight for the development of new algorithms.

Discrete sequence data and continuous expression data are two of the most common data types in computational biology. We discuss sequence data analysis in Sections 2–5, and gene expression data analysis in Section 6. A main objective of computational biology research surrounding the "central dogma" is to study how the gene sequences affect the gene expression. In Section 2 we attempt to find conserved patterns in functionally related gene sequences as an effort to explain the relationship of their gene expression. In Section 3 we give an EM algorithm for multiple sequence alignment, where the goal is to establish "relatedness" of different sequences. Based on the alignment of evolutionary related DNA sequences, another EM algorithm for detecting potentially expression-related regions is introduced in Section 4. An alternative way to deduce the relationship between gene sequence and gene expression is to check the effect of sequence variation within the population of a species. In Section 5 we provide an EM algorithm to deal with this type of small sequence variation. In Section 6 we review the clustering analysis of microarray gene-expression data, which is important for connecting the phenotype variation among individuals with the expression level variation. Finally, in Section 7 we discuss trends in computational biology research.

2. SEQUENCE MOTIF DISCOVERY AND GENE REGULATION

In order for a gene to be transcribed, special proteins called transcription factors (TFs) are often required to bind to certain sequences, called transcription factor binding sites (TFBSs). These sites are usually 6–20 bp long and are mostly located upstream of the gene. One TF is usually involved in the regulation of many genes, and the TFBSs that the TF recognizes often exhibit strong sequence specificity and conservation (e.g., the first position of the TFBSs is likely T, etc.). This specific pattern is called a TF binding motif (TFBM). For example, Figure 1 shows a motif of length 6. The motif is represented by the position-specific frequency matrix $(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_6)$, which is derived from the alignment of 5 motif sites by calculating position-dependent frequencies of the four nucleotides.

In order to understand how genes' mRNA expression levels are regulated in the cell, it is crucial to identify TFBSs and to characterize TFBMs. Although much progress has been made in developing experimental techniques for identifying these TFBSs, these techniques are typically expensive and time-consuming. They are also limited by experimental conditions, and cannot pinpoint the binding sites exactly. In the past twenty years, computational biologists and statisticians have developed many successful in silico methods to aid biologists in finding TFBSs, and these efforts have contributed significantly to our understanding of transcription regulation.

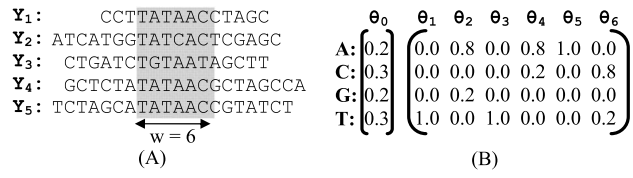

FIG. 1. *Transcription factor binding sites and motifs.* (A) *Each of the five sequences contains a TFBS of length* 6. *The local alignment of these sites is shown in the gray box.* (B) *The frequency of the nucleotides outside of the gray box is shown as* $\boldsymbol{\theta}_0$. *The frequency of the nucleotides in the ith column of the gray box is shown as* $\boldsymbol{\theta}_i$.

Likewise, motif discovery for protein sequences is important for identifying structurally or functionally important regions (domains) and understanding proteins' functional components, or active sites. For example, using a Gibbs sampling-based motif finding algorithm, Lawrence et al. (1993) was able to predict the key *helix-turn-helix* motif among a family of transcription activators. Experimental approaches for determining protein motifs are even more expensive and slower than those for DNAs, whereas computational approaches are more effective than those for TFBSs predictions.

The underlying logic of computational motif discovery is to find patterns that are "enriched" in a given set of sequence data. Common methods include word enumeration (Sinha and Tompa, 2002; Hampson, Kibler and Baldi, 2002; Pavesi et al., 2004), position-specific frequency matrix updating (Stormo and Hartzell, 1989; Lawrence and Reilly, 1990; Lawrence et al., 1993) or a combination of the two (Liu, Brutlag and Liu, 2002). The word enumeration approach uses a specific consensus word to represent a motif. In contrast, the position-specific frequency matrix approach formulates a motif as a weight matrix. Jensen et al. (2004) provide a review of these motif discovery methods. Tompa et al. (2005) compared the performance of various motif discovery tools. Traditionally, researchers have employed various heuristics, such as evaluating excessiveness of word counts or maximizing certain information criteria to guide motif finding. The EM algorithm was introduced by Lawrence and Reilly (1990) to deal with the motif finding problem.

As shown in Figure 1, suppose we are given a set of K sequences $\mathbf{Y} \equiv (\mathbf{Y}_1, \ldots, \mathbf{Y}_K)$, where $\mathbf{Y}_k \equiv (Y_{k,1}, \ldots, Y_{k,L_k})$ and $Y_{k,l}$ takes values in an alphabet of d residues ($d = 4$ for DNA/RNA and 20 for protein). The alphabet is denoted by $\mathbf{R} \equiv (r_1, \ldots, r_d)$. Motif sites in this paper refer to a set of contiguous segments of the same length w (e.g., the marked 6-mers in Figure 1). This concept can be further generalized via a hidden Markov model to allow gaps and position deletions (see Section 3 for HMM discussions). The weight matrix, or *Product-Multinomial* motif model, was first introduced by Stormo and Hartzell (1989) and later formulated rigorously in Liu, Neuwald and Lawrence (1995). It assumes that, if $Y_{k,l}$ is the ith position of a motif site, it follows the multinomial distribution with the probability vector $\boldsymbol{\theta}_i \equiv (\theta_{i1}, \ldots, \theta_{id})$; we denote this model as $PM(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_w)$. If $Y_{k,l}$ does not belong to any motif site, it is generated independently from the multinomial distribution with parameter $\boldsymbol{\theta}_0 \equiv (\theta_{01}, \ldots, \theta_{0d})$.

Let $\boldsymbol{\Theta} \equiv (\boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_w)$. For sequence $\mathbf{Y}_k$, there are $L'_k = L_k - w + 1$ possible positions a motif site of length w may start. To represent the motif locations, we introduce the unobserved indicators $\boldsymbol{\Gamma} \equiv \{\Gamma_{k,l} \mid 1 \le k \le K, 1 \le l \le L'_k\}$, where $\Gamma_{k,l} = 1$ if a motif site starts at position l in sequence $\mathbf{Y}_k$, and $\Gamma_{k,l} = 0$ otherwise. As shown in Figure 1, it is straightforward to estimate $\boldsymbol{\Theta}$ if we know where the motif sites are. The motif location indicators $\boldsymbol{\Gamma}$ are the missing data that makes the EM framework a natural choice for this problem. For illustration, we further assume that there is exactly one motif site within each sequence and that its location in the sequence is uniformly distributed. This means that $\sum_l \Gamma_{k,l} = 1$ for all k and $P(\Gamma_{k,l} = 1) = \frac{1}{L'_k}$.

Given $\Gamma_{k,l} = 1$, the probability of each observed sequence $\mathbf{Y}_k$ is

$$P(\mathbf{Y}_k \mid \Gamma_{k,l} = 1, \boldsymbol{\Theta}) = \boldsymbol{\theta}_0^{\mathbf{h}(\mathbf{B}_{k,l})} \prod_{j=1}^{w} \boldsymbol{\theta}_i^{\mathbf{h}(\mathbf{Y}_{k,l+j-1})}. \tag{1}$$

In this expression, $\mathbf{B}_{k,l} \equiv \{Y_{k,j} : j < l \text{ or } j \ge l + w\}$ is the set of letters of nonsite positions of $\mathbf{Y}_k$. The counting function $\mathbf{h}(\cdot)$ takes a set of letter symbols as input and outputs the column vector $(n_1, \ldots, n_d)^T$, where n_i is the number of base type r_i in the input set. We define the vector power function as $\boldsymbol{\theta}_i^{\mathbf{h}(\cdot)} \equiv \prod_{j=1}^{d} \theta_{ij}^{n_j}$ for $i = 0, \ldots, w$. Thus, the complete-data likelihood function is the product of equation (1) for k from 1 to K, that is,

$$\begin{aligned} P(\mathbf{Y}, \boldsymbol{\Gamma} \mid \boldsymbol{\Theta}) &\propto \prod_{k=1}^{K} \prod_{l=1}^{L'_k} P(\mathbf{Y}_k \mid \Gamma_{k,l} = 1, \boldsymbol{\Theta})^{\Gamma_{k,l}} \\ &= \boldsymbol{\theta}_0^{\mathbf{h}(\mathbf{B}_{\boldsymbol{\Gamma}})} \prod_{i=1}^{w} \boldsymbol{\theta}_i^{\mathbf{h}(\mathbf{M}_{\boldsymbol{\Gamma}}^{(i)})}, \end{aligned}$$

where $\mathbf{B}_{\boldsymbol{\Gamma}}$ is the set of all nonsite bases, and $\mathbf{M}_{\boldsymbol{\Gamma}}^{(i)}$ is the set of nucleotide bases at position i of the TFBSs given the indicators $\boldsymbol{\Gamma}$.

The MLE of $\boldsymbol{\Theta}$ from the complete-data likelihood can be determined by simple counting, that is,

$$\hat{\boldsymbol{\theta}}_i = \frac{\mathbf{h}(\mathbf{M}_{\boldsymbol{\Gamma}}^{(i)})}{K} \quad \text{and} \quad \hat{\boldsymbol{\theta}}_0 = \frac{\mathbf{h}(\mathbf{B}_{\boldsymbol{\Gamma}})}{\sum_{k=1}^{K}(L_k - w)}.$$

The EM algorithm for this problem is quite intuitive. In the E-step, one uses the current parameter values $\boldsymbol{\Theta}^{(t)}$ to compute the expected values of $h(\mathbf{M}_{\boldsymbol{\Gamma}}^{(i)})$ and $h(\mathbf{B}_{\boldsymbol{\Gamma}})$. More precisely, for sequence Y_k, we compute its likelihood of being generated from $\boldsymbol{\Theta}^{(t)}$ conditional on each

possible motif location $\Gamma_{k,l} = 1$,

$$w_{k,l} \equiv P\big(\mathbf{Y}_k|\Gamma_{k,l} = 1, \boldsymbol{\Theta}^{(t)}\big) = \left(\frac{\boldsymbol{\theta}_1}{\boldsymbol{\theta}_0}\right)^{\mathbf{h}(Y_{k,l})} \cdots \left(\frac{\boldsymbol{\theta}_w}{\boldsymbol{\theta}_0}\right)^{\mathbf{h}(Y_{k,l+w-1})} \boldsymbol{\theta}_0^{\mathbf{h}(\mathbf{Y}_k)}.$$

Letting $W_k \equiv \sum_{l=1}^{L'_k} w_{k,l}$, we then compute the expected count vectors as

$$E_{\boldsymbol{\Gamma}|\boldsymbol{\Theta}^{(t)},\mathbf{Y}}\big[\mathbf{h}\big(\mathbf{M}_{\boldsymbol{\Gamma}}^{(i)}\big)\big] = \sum_{k=1}^{K}\sum_{l=1}^{L'_k} \frac{w_{k,l}}{W_k}\mathbf{h}(Y_{k,l+i-1}),$$

$$E_{\boldsymbol{\Gamma}|\boldsymbol{\Theta}^{(t)},\mathbf{Y}}[\mathbf{h}(\mathbf{B}_{\boldsymbol{\Gamma}})] = \mathbf{h}\big(\{Y_{k,l} : 1 \le k \le K, 1 \le l \le L_k\}\big) - \sum_{i=1}^{w} E_{\boldsymbol{\Gamma}|\boldsymbol{\Theta}^{(t)},\mathbf{Y}}\big[\mathbf{h}\big(\mathbf{M}_{\boldsymbol{\Gamma}}^{(i)}\big)\big].$$

In the M-step, one simply computes

$$\boldsymbol{\theta}_i^{(t+1)} = \frac{E_{\boldsymbol{\Gamma}|\boldsymbol{\Theta}^{(t)},\mathbf{Y}}\big[\mathbf{h}\big(\mathbf{M}_{\boldsymbol{\Gamma}}^{(i)}\big)\big]}{K} \quad \text{and}$$

$$\boldsymbol{\theta}_0^{(t+1)} = \frac{E_{\boldsymbol{\Gamma}|\boldsymbol{\Theta}^{(t)},\mathbf{Y}}[\mathbf{h}(\mathbf{B}_{\boldsymbol{\Gamma}})]}{\sum_{k=1}^{K}(L_k - w)}.$$

It is necessary to start with a nonzero initial weight matrix $\boldsymbol{\Theta}^{(0)}$ so as to guarantee that $P(\mathbf{Y}_k|\Gamma_{k,l} = 1, \boldsymbol{\Theta}^{(t)}) > 0$ for all l. At convergence the algorithm yields both the MLE $\hat{\boldsymbol{\Theta}}$ and predictive probabilities for candidate TFBS locations, that is, $P(\Gamma_{k,l} = 1|\hat{\boldsymbol{\Theta}}, \mathbf{Y})$.

Cardon and Stormo (1992) generalized the above simple model to accommodate insertions of variable lengths in the middle of a binding site. To overcome the restriction that each sequence contains exactly one motif site, Bailey and Elkan (1994, 1995a, 1995b) introduced a parameter p_0 describing the prior probability for each sequence position to be the start of a motif site, and designed a modified EM algorithm called the Multiple EM for Motif Elicitation (MEME). Independently, Liu, Neuwald and Lawrence (1995) presented a full Bayesian framework and Gibbs sampling algorithm for this problem. Compared with the EM approach, the Markov chain Monte Carlo (MCMC)-based approach has the advantages of making more flexible moves during the iteration and incorporating additional information such as motif location and orientation preference in the model.

The generalizations in Bailey and Elkan (1994) and Liu, Neuwald and Lawrence (1995) assume that all overlapping subsequences of length w in the sequence data set are from a finite mixture model. More precisely, each subsequence of length w is treated as an independent sample from a mixture of $PM(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_w)$ and $PM(\boldsymbol{\theta}_0, \ldots, \boldsymbol{\theta}_0)$ [independent Multinomial(θ_0) in all w positions]. The EM solution of this mixture model formulation then leads to the MEME algorithm of Bailey and Elkan (1994). To deal with the situation that w may not be known precisely, MEME searches motifs of a range of different widths separately, and then performs model selection by optimizing a heuristic function based on the maximum likelihood ratio test. Since its release, MEME has been one of the most popular motif discovery tools cited in the literature. The Google scholar search gives a count of 1397 citations as of August 30th, 2009. Although it is 15 years old, its performance is still comparable to many new algorithms (Tompa et al., 2005).

3. MULTIPLE SEQUENCE ALIGNMENT

Multiple sequence alignment (MSA) is an important tool for studying structures, functions and the evolution of proteins. Because different parts of a protein may have different functions, they are subject to different selection pressures during evolution. Regions of greater functional or structural importance are generally more conserved than other regions. Thus, a good alignment of protein sequences can yield important evidence about their functional and structural properties.

Many heuristic methods have been proposed to solve the MSA problem. A popular approach is the progressive alignment method (Feng and Doolittle, 1987), in which the MSA is built up by aligning the most closely related sequences first and then adding more distant sequences successively. Many alignment programs are based on this strategy, such as MULTALIGN (Barton and Sternberg, 1987), MULTAL (Taylor, 1988) and, the most influential one, ClustalW (Thompson, Higgins and Gibson, 1994). Usually, a *guide tree* based on pairwise similarities between the protein sequences is constructed prior to the multiple alignment to determine the order for sequences to enter the alignment. Recently, a few new progressive alignment algorithms with significantly improved alignment accuracies and speed have been proposed, including T-Coffee (Notredame, Higgins and Heringa, 2000), MAFFT (Katoh et al., 2005), PROBCONS (Do et al., 2005) and MUSCLE (Edgar, 2004a, 2004b). They differ from previous approaches and each other mainly in the construction of the guide tree and in the objective function for judging the goodness of the alignment. Batzoglou (2005) and Wallace, Blackshields and Higgins (2005) reviewed these algorithms.

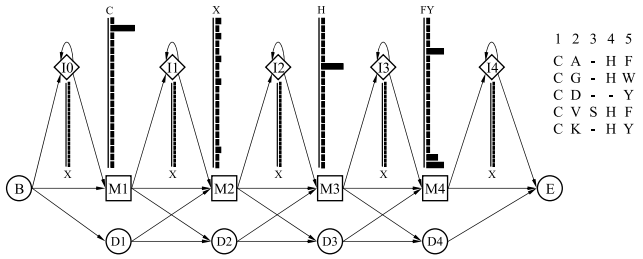

FIG. 2. *Profile hidden Markov model. A modified toy example is adopted from Eddy (1998). It shows the alignment of five sequences, each containing only three to five letters. The first position is enriched with Cysteine (C), the fourth position is enriched with Histidine (H), and the fifth position is enriched with Phenylalanine (F) and Tyrosine (Y). The third sequence has a deletion at the fourth position, and the fourth sequence has an insertion at the third position. This simplified model does not allow insertion and deletion states to follow each other.*

An important breakthrough in solving the MSA problem is the introduction of a probabilistic generative model, the profile hidden Markov model by Krogh et al. (1994). The profile HMM postulates that the N observed sequences are generated as independent but indirect observations (*emissions*) from a Markov chain model illustrated in Figure 2. The underlying unobserved Markov chain consists of three types of states: match, insertion and deletion. Each match or insertion state *emits* a letter chosen from the alphabet $\mathbf{R}$ (size $d = 20$ for proteins) according to a multinomial distribution. The deletion state does not emit any letter, but makes the sequence generating process skip one or more match states. A multiple alignment of the N sequences is produced by aligning the letters that are emitted from the same match state.

Let $\mathbf{\Gamma}_i$ denote the unobserved state path through which the ith sequence is generated from the profile HMM, and $\mathbf{S}$ the set of all states. Let $\mathbf{\Theta}$ denote the set of all global parameters of this model, including emission probabilities in match and insertion states e_{lr} ($l \in \mathbf{S}, r \in \mathbf{R}$), and transition probabilities among all hidden states t_{ab} ($a, b \in \mathbf{S}$). The complete-data log-likelihood function can be written as

$$\begin{aligned}
&\log P(\mathbf{Y}, \mathbf{\Gamma}|\mathbf{\Theta}) \\
&\quad = \sum_{i=1}^{N} [\log P(\mathbf{Y}_i|\mathbf{\Gamma}_i, \mathbf{\Theta}) + \log P(\mathbf{\Gamma}_i|\mathbf{\Theta})] \\
&\quad = \sum_{i=1}^{N} \Bigg[\sum_{l\in\mathbf{S}, r\in\mathbf{R}} M_{lr}(\mathbf{\Gamma}_i) \log e_{lr} \\
&\qquad\qquad + \sum_{a,b\in\mathbf{S}} N_{ab}(\mathbf{\Gamma}_i) \log t_{ab} \Bigg],
\end{aligned}$$

where $M_{lr}(\mathbf{\Gamma}_i)$ is the count of letter r in sequence $\mathbf{Y}_i$ that is generated from state l according to $\mathbf{\Gamma}_i$, and $N_{ab}(\mathbf{\Gamma}_i)$ is the count of state transitions from a to b in the path $\mathbf{\Gamma}_i$ for sequence $\mathbf{Y}_i$.

The E-step involves calculating the expected counts of emissions and transitions, that is, $E[M_{lr}(\mathbf{\Gamma}_i)|\mathbf{\Theta}^{(t)}]$ and $E[N_{ab}(\mathbf{\Gamma}_i)|\mathbf{\Theta}^{(t)}]$, averaging over all possible generating paths $\mathbf{\Gamma}_i$. The Q-function is

$$\begin{aligned}
Q(\mathbf{\Theta}|\mathbf{\Theta}^{(t)}) = \sum_{i=1}^{N} \sum_{\mathbf{\Gamma}_i} \frac{P(\mathbf{\Gamma}_i, \mathbf{Y}_i|\mathbf{\Theta}^{(t)})}{P(\mathbf{Y}_i|\mathbf{\Theta}^{(t)})} \\
\cdot \Bigg[\sum_{l\in\mathbf{S}, r\in\mathbf{R}} \log(e_{lr}) M_{lr}(\mathbf{\Gamma}_i) \\
+ \sum_{a,b\in\mathbf{S}} \log(t_{ab}) N_{ab}(\mathbf{\Gamma}_i) \Bigg].
\end{aligned}$$

A brute-force enumeration of all paths is prohibitively expensive in computation. Fortunately, one can apply a forward–backward dynamic programming technique to compute the expectations for each sequence and then sum them all up.

In the M-step, the emission and transition probabilities are updated as the ratio of the expected event occurrences (sufficient statistics) divided by the total expected emission or transition events:

$$\begin{aligned}
e_{lr}^{(t+1)} &= \frac{\sum_i \{m_{lr}(\mathbf{Y}_i)/P(\mathbf{Y}_i|\mathbf{\Theta}^{(t)})\}}{\sum_i \{m_l(\mathbf{Y}_i)/P(\mathbf{Y}_i|\mathbf{\Theta}^{(t)})\}}, \\
t_{ab}^{(t+1)} &= \frac{\sum_i \{n_{ab}(\mathbf{Y}_i)/P(\mathbf{Y}_i|\mathbf{\Theta}^{(t)})\}}{\sum_i \{n_a(\mathbf{Y}_i)/P(\mathbf{Y}_i|\mathbf{\Theta}^{(t)})\}},
\end{aligned}$$

where

$$\begin{aligned}
m_{lr}(\mathbf{Y}_i) &= \sum_{\mathbf{\Gamma}_i} M_{lr}(\mathbf{\Gamma}_i) P(\mathbf{\Gamma}_i, \mathbf{Y}_i|\mathbf{\Theta}^{(t)}), \\
n_{ab}(\mathbf{Y}_i) &= \sum_{\mathbf{\Gamma}_i} N_{ab}(\mathbf{\Gamma}_i) P(\mathbf{\Gamma}_i, \mathbf{Y}_i|\mathbf{\Theta}^{(t)}), \\
m_l(\mathbf{Y}_i) &= \sum_{r\in\mathbf{R}} m_{lr}(\mathbf{Y}_i), \quad n_a(\mathbf{Y}_i) = \sum_{b\in\mathbf{S}} n_{ab}(\mathbf{Y}_i).
\end{aligned}$$

This method is called the Baum–Welch algorithm (Baum et al., 1970), and is mathematically equivalent to the EM algorithm. Conditional on the MLE $\hat{\mathbf{\Theta}}$, the best alignment path for each sequence can be found efficiently by the Viterbi algorithm (see Durbin et al., 1998, Chapter 5, for details).

The profile HMM provides a rigorous statistical modeling and inference framework for the MSA problem. It has also played a central role in advancing the

understanding of protein families and domains. A protein family database, Pfam (Finn et al., 2006), has been built using profile HMM and has served as an essential source of data in the field of protein structure and function research. Currently there are two popular software packages that use profile HMMs to detect remote protein homologies: HMMER (Eddy, 1998) and SAM (Hughey and Krogh, 1996; Karplus, Barrett and Hughey, 1999). Madera and Gough (2002) gave a comparison of these two packages.

There are several challenges in fitting the profile HMM. First, the size of the model (the number of match, insertion and deletion states) needs to be determined before model fitting. It is common to begin fitting a profile HMM by setting the number of match states equal to the average sequence length. Afterward, a strategy called "model surgery" (Krogh et al., 1994) can be applied to adjust the model size (by adding or removing a match state depending on whether an insertion or a deletion is used too often). Eddy (1998) used a *maximum a posteriori* (MAP) strategy to determine the model size in HMMER. In this method the number of match states is given a prior distribution, which is equivalent to adding a penalty term in the log-likelihood function.

Second, the number of sequences is sometimes too small for parameter estimation. When calculating the conditional expectation of the sufficient statistics, which are counts of residues at each state and state transitions, there may not be enough data, resulting in zero counts which could make the estimation unstable. To avoid the occurrence of zero counts, pseudo-counts can be added. This is equivalent to using a Dirichlet prior for the multinomial parameters in a Bayesian formulation.

Third, the assumption of sequence independence is often violated. Due to the underlying evolutionary relationship (unknown), some of the sequences may share much higher mutual similarities than others. Therefore, treating all sequences as i.i.d. samples may cause serious biases in parameter estimation. One possible solution is to give each sequence a weight according to its importance. For example, if two sequences are identical, it is reasonable to give each of them half the weight of other sequences. The weights can be easily integrated into the M-step of the EM algorithm to update the model parameters. For example, when a sequence has a weight of 0.5, all the emission and transition events contributed by this sequence will be counted by half. Many methods have been proposed to assign weights to the sequences (Durbin et al., 1998), but it is not clear how to set the weights in a principled way to best account for sequence dependency.

Last, since the EM algorithm can only find local modes of the likelihood function, some stochastic perturbation can be introduced to help find better modes and improve the alignment. Starting from multiple random initial parameters is strongly recommended. Krogh et al. (1994) combined simulated annealing into Baum–Welch and showed some improvement. Baldi and Chauvin (1994) developed a generalized EM (GEM) algorithm using a gradient ascent calculation in an attempt to infer HMM parameters in a smoother way.

Despite many advantages of the profile HMM, it is no longer the mainstream MSA tool. A main reason is that the model has too many free parameters, which render the parameter estimation very unstable when there are not enough sequences (fewer than 50, say) in the alignment. In addition, the vanilla EM algorithm and its variations developed by early researchers for the MSA problem almost always converge to suboptimal alignments. Recently, Edlefsen (2009) have developed an ECM algorithm for MSA that appears to have much improved convergence properties. It is also difficult for the profile HMM to incorporate other kinds of information, such as 3D protein structure and guide tree. Some recent programs such as 3D-Coffee (O'Sullivan et al., 2004) and MAFFT are more flexible as they can incorporate this information into the objective function and optimize it. We believe that the Monte Carlo-based Bayesian approaches, which can impose more model constraints (e.g., to capitalize on the "motif" concept) and make more flexible MCMC moves, might be a promising route to rescue profile HMM (see Liu, Neuwald and Lawrence, 1995; Neuwald and Liu, 2004).

4. COMPARATIVE GENOMICS

A main goal of comparative genomics is to identify and characterize functionally important regions in the genome of multiple species. An assumption underlying such studies is that, due to evolutionary pressure, functional regions in the genome evolve much more slowly than most nonfunctional regions due to functional constraints (Wolfe, Sharp and Li, 1989; Boffelli et al., 2003). Regions that evolve more slowly than the background are called evolutionarily conserved elements.

Conservation analysis (comparing genomes of related species) is a powerful tool for identifying functional elements such as protein/RNA coding regions

and transcriptional regulatory elements. It begins with an alignment of multiple orthologous sequences (sequences evolved from the same common ancestral sequence) and a conservation score for each column of the alignment. The scores are calculated based on the likelihood that each column is located in a conserved element. The phylogenetic hidden Markov model (Phylo-HMM) was introduced to infer the conserved regions in the genome (Yang, 1995; Felsenstein and Churchill, 1996; Siepel et al., 2005). The statistical power of Phylo-HMM has been systematically studied by Fan et al. (2007). Siepel et al. (2005) used the EM algorithm for estimating parameters in Phylo-HMM. Their results, provided by the UCSC genome browser database (Karolchik et al., 2003), are very influential in the computational biology community. By August 2009, the paper of Siepel et al. (2005) had been cited 413 times according to the Web of Science database.

As shown in Figure 3, the alignment modeled by Phylo-HMM can be seen as generated from two steps. First, a sequence of L sites is generated from a two-state HMM, with the hidden states being conserved or nonconserved sites. Second, a nucleotide is generated for each site of the common ancestral sequence and evolved to the contemporary nucleotides along all branches of a phylogenetic tree independently according to the corresponding phylogenetic model.

Let μ and ν be the transition probabilities between the two states, and let the phylogenetic models for nonconserved and conserved states be $\psi_n = (Q, \pi, \tau, \boldsymbol{\beta})$ and $\psi_c = (Q, \pi, \tau, \rho\boldsymbol{\beta})$, respectively. Here π is the emission probability vector of the four nucleotides (A, C, G and T) in the common ancestral sequence $\mathbf{x}_0$; τ is the tree topology of the corresponding phylogeny; $\boldsymbol{\beta}$ is a vector of non-negative real numbers representing branch lengths of the tree, which are measured by the expected number of substitutions per site. The difference between the two states is characterized by a scaling parameter $\rho \in [0, 1)$ applied to the branch lengths of only the conserved state, which means fewer substitutions. The nucleotide substitution model considers a descendent nucleotide to have evolved from its ancestor by a continuous-time time-homogeneous Markov process with transition kernel Q, also called the substitution rate matrix (Tavaré, 1986). The transition kernels for all branches are assumed to be the same. Many parametric forms are available for the 4-by-4 nucleotide substitution rate matrix Q, such as the Jukes–Cantor substitution matrix and the general time-reversible substitution matrix (Yang, 1997). The

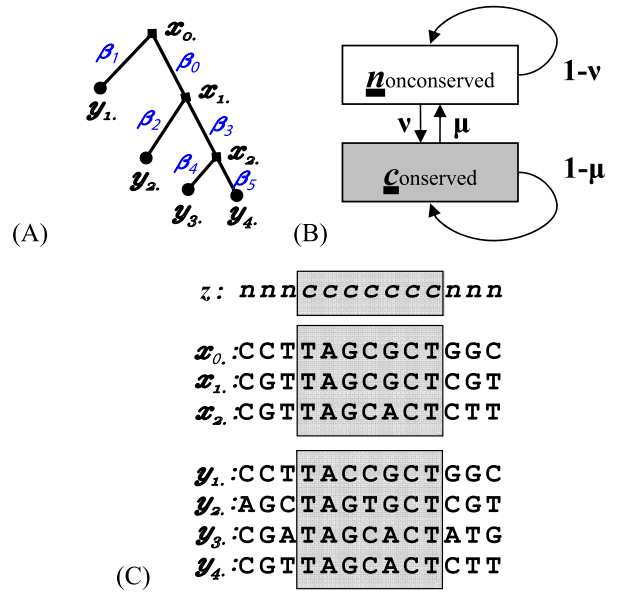

FIG. 3. *Two-state Phylo-HMM*. (A) *Phylogenetic tree*: *The tree shows the evolutionary relationship of four contemporary sequences* $(\mathbf{y}_{1\cdot}, \mathbf{y}_{2\cdot}, \mathbf{y}_{3\cdot}, \mathbf{y}_{4\cdot})$. *They are evolved from the common ancestral sequence* $\mathbf{x}_{0\cdot}$, *with two additional internal nodes* (*ancestors*), $\mathbf{x}_{1\cdot}$ *and* $\mathbf{x}_{2\cdot}$. *The branch lengths* $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5)$ *indicate the evolutionary distance between two nodes, which are measured by the expected number of substitutions per site*. (B) *HMM state-transition diagram*: *The system consists of a state for conserved sites and a state for nonconserved sites* (*c and n, respectively*). *The two states are associated with different phylogenetic models* (ψ_c *and* ψ_n), *which differ by a scaling parameter* ρ. (C) *An illustrative alignment generated by this model*: *A state sequence* (z) *is generated according to* μ *and* ν. *For each site in the state sequence, a nucleotide is generated for the root node in the phylogenetic tree and then for subsequent child nodes according to the phylogenetic model* (ψ_c *or* ψ_n). *The observed alignment* $\mathbf{Y} = (\mathbf{y}_{1\cdot}, \mathbf{y}_{2\cdot}, \mathbf{y}_{3\cdot}, \mathbf{y}_{4\cdot})$ *is composed of all nucleotides in the leaf nodes. The state sequence* $\mathbf{z}$ *and all ancestral sequences* $\mathbf{X} = (\mathbf{x}_{0\cdot}, \mathbf{x}_{1\cdot}, \mathbf{x}_{2\cdot})$ *are unobserved*.

nucleotide transition probability matrix for a branch of length β_i is $e^{\beta_i Q}$.

Siepel et al. (2005) assumed that the tree topology τ and the emission probability vector π are known. In this case, the observed alignment $\mathbf{Y} = (\mathbf{y}_{1\cdot}, \mathbf{y}_{2\cdot}, \mathbf{y}_{3\cdot}, \mathbf{y}_{4\cdot})$ is a matrix of nucleotides. The parameter of interest is $\boldsymbol{\Theta} = (\mu, \nu, Q, \rho, \boldsymbol{\beta})$. The missing information $\boldsymbol{\Gamma} = (\mathbf{z}, \mathbf{X})$ includes the state sequence $\mathbf{z}$ and the ancestral DNA sequences $\mathbf{X}$. The complete-data likelihood is written as

$$P(\mathbf{Y}, \boldsymbol{\Gamma} | \boldsymbol{\Theta}) = b_{z_1} P(\mathbf{y}_{\cdot 1}, \mathbf{x}_{\cdot 1} | \psi_{z_1}) \prod_{i=2}^{L} a_{z_{i-1} z_i} P(\mathbf{y}_{\cdot i}, \mathbf{x}_{\cdot i} | \psi_{z_i}).$$

Here $\mathbf{y}_{\cdot i}$ is the ith column of the alignment $\mathbf{Y}$, $z_i \in \{c, n\}$ is the hidden state of the ith column, $(b_c, b_n) = (\frac{\nu}{\mu+\nu}, \frac{\mu}{\mu+\nu})$ is the initial state probability of the HMM if the chain is stationary, and $a_{z_{i-1}z_i}$ is the transition probability (as illustrated in Figure 3).

The EM algorithm is applied to obtain the MLE of $\boldsymbol{\Theta}$. In the E-step, we calculate the expectation of the complete-data log-likelihood under the distribution $P(\mathbf{z}, \mathbf{X}|\boldsymbol{\Theta}^{(t)}, \mathbf{Y})$. The marginalization of $\mathbf{X}$, conditional on $\mathbf{z}$ and other variables, can be accomplished efficiently site-by-site using the peeling or pruning algorithm for the phylogenetic tree (Felsenstein, 1981). The marginalization of $\mathbf{z}$ can be done efficiently by the forward–backward procedure for HMM (Baum et al., 1970; Rabiner, 1989). For the M-step, we can use the Broyden–Fletcher–Goldfarb–Shanno (BFGS) quasi-Newton algorithm. After we obtain the MLE of $\boldsymbol{\Theta}$, a forward–backward dynamic programming method (Liu, 2001) can then be used to compute the posterior probability that a given hidden state is conserved, that is, $P(z_i = c|\hat{\boldsymbol{\Theta}}, \mathbf{Y})$, which is the desired conservation score.

As shown in the Phylo-HMM example, the phylogenetic tree model is key to integrating multiple sequences for evolutionary analysis. This model is also used for comparing protein or RNA sequences. Due to its intuitive and efficient handling of the missing evolutionary history, the EM algorithm has always been a main approach for estimating parameters of the tree. For example, Felsenstein (1981) used the EM algorithm to estimate the branch length $\boldsymbol{\beta}$, Bruno (1996) and Holmes and Rubin (2002) used the EM algorithm to estimate the residue usage π and the substitution rate matrix Q, Friedman et al. (2002) used an extension of the EM algorithm to estimate the phylogenetic tree topology τ, and Holmes (2005) used the EM algorithm for estimating insertion and deletion rates. Yang (1997) implemented some of the above algorithms in the phylogenetic analysis software PAML. A limitation of the Phylo-HMM model is the assumption of a good multiple sequence alignment, which is often not available.

5. SNP HAPLOTYPE INFERENCE

A Single Nucleotide Polymorphism (SNP) is a DNA sequence variation in which a single base is altered that occurs in at least 1% of the population. For example, the DNA fragments CCTG*A*GGAG and CCTG*T*GGAG from two homologous chromosomes (the paired chromosomes of the same individual, one from each parent) differ at a single locus. This example is actually a real SNP in the human β-globin gene, and it is associated with the sickle-cell disease. The different forms (A and T in this example) of a SNP are called alleles. Most SNPs have only two alleles in the population. Diploid organisms, such as humans, have two homologous copies of each chromosome. Thus, the genotype (i.e., the specific allelic makeup) of an individual may be AA, TT or AT in this example. A phenotype is a morphological feature of the organism controlled or affected by a genotype. Different genotypes may produce the same phenotype. In this example, individuals with genotype TT have a very high risk of the sickle-cell disease. A haplotype is a combination of alleles at multiple SNP loci that are transmitted together on the same chromosome. In other words, haplotypes are sets of *phased* genotypes. An example is given in Figure 4, which shows the genotypes of three individuals at four SNP loci. For the first individual, the arrangement of its alleles on two chromosomes must be ACAC and ACGC, which are the haplotypes compatible with its observed genotype data.

One of the main tasks of genetic studies is to locate genetic variants (mainly SNPs) that are associated with inheritable diseases. If we know the haplotypes of all related individuals, it will be easier to rebuild the evolutionary history and locate the disease mutations. Unfortunately, the phase information needed to build haplotypes from genotype information is usually unavailable because laboratory haplotyping methods, unlike genotyping technologies, are expensive and low-throughput.

The use of the EM algorithm has a long history in population genetics, some of which predates Dempster, Laird and Rubin (1977). For example, Ceppellini, Siniscalco and Smith (1955) invented an EM algorithm

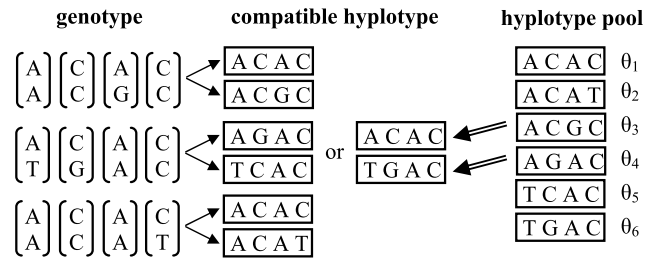

FIG. 4. *Haplotype reconstruction. We observed the genotypes of three individuals at* 4 *SNP loci. The* 1*st and* 3*rd individuals each have a unique haplotype phase, whereas the* 2*nd individual has two compatible haplotype phases. We pool all possible haplotypes together and associated with them a haplotype frequency vector* $(\theta_1, \ldots, \theta_6)$. *Each individual's two haplotypes are then assumed to be random draws (with replacement) from this pool of weighted haplotypes.*

to estimate allele frequencies when there is no one-to-one correspondence between phenotype and genotype; Smith (1957) used an EM algorithm to estimate the recombination frequency; and Ott (1979) used an EM algorithm to study genotype-phenotype relationships from pedigree data. Weeks and Lange (1989) reformulated these earlier applications in the modern EM framework of Dempster, Laird and Rubin (1977). Most early works were single-SNP Association studies. Thompson (1984) and Lander and Green (1987) designed EM algorithms for joint linkage analysis of three or more SNPs. With the accumulation of SNP data, more and more researchers have come to realize the importance of haplotype analysis (Liu et al., 2001). Haplotype reconstruction based on genotype data has therefore become a very important intermediate step in disease association studies.

The haplotype reconstruction problem is illustrated in Figure 4. Suppose we observed the genotype data $\mathbf{Y} = (Y_1, \ldots, Y_n)$ for n individuals, and we wish to predict the corresponding haplotypes $\mathbf{\Gamma} = (\Gamma_1, \ldots, \Gamma_n)$, where $\Gamma_i = (\Gamma_i^+, \Gamma_i^-)$ is the haplotype pair of the ith individual. The haplotype pair Γ_i is said to be compatible with the genotype Y_i, which is expressed as $\Gamma_i^+ \oplus \Gamma_i^- = Y_i$, if the genotype Y_i can be generated from the haplotype pair. Let $\mathbf{H} = (H_1, \ldots, H_m)$ be the pool of all distinct haplotypes and let $\mathbf{\Theta} = (\theta_1, \ldots, \theta_m)$ be the corresponding frequencies in the population.

The first simple model considered in the literature assumes that each individual's genotype vector is generated by two haplotypes from the pool chosen independently with probability vector $\mathbf{\Theta}$. This is a very good model if the region spanned by the markers in consideration is sufficiently short that no recombination has occurred, and if mating in the population is random. Under this model, we have

$$P(\mathbf{Y}|\mathbf{\Theta}) = \prod_{i=1}^{n} \Bigg(\sum_{(j,k):H_j \oplus H_k = Y_i} \theta_j \theta_k \Bigg).$$

If $\mathbf{\Gamma}$ is known, we can directly write down the MLE of $\mathbf{\Theta}$ as $\theta_j = \frac{n_j}{2n}$, where the sufficient statistic n_j is the number of occurrences of haplotype H_j in $\mathbf{\Gamma}$. Therefore, in the EM framework, we simply replace n_j by its expected value over the distribution of $\mathbf{\Gamma}$ when $\mathbf{\Gamma}$ is unobserved. More specifically, the EM algorithm is a simple iteration of

$$\theta_j^{(t+1)} = \frac{E_{\mathbf{\Gamma}|\mathbf{\Theta}^{(t)},\mathbf{Y}}(n_j)}{2n},$$

where $\mathbf{\Theta}^{(t)}$ is the current estimate of the haplotype frequencies, and n_j is the count of haplotypes H_j that exist in $\mathbf{Y}$.

The use of the EM algorithm for haplotype analysis has been coupled with the large-scale generation of SNP data. Early attempts include Excoffier and Slatkin (1995), Long, Williams and Urbanek (1995), Hawley and Kidd (1995) and Chiano and Clayton (1998). One problem of these traditional EM approaches is that the computational complexity of the E-step grows exponentially as the number of SNPs in the haplotype increases. Qin, Niu and Liu (2002) incorporated a "partition–ligation" strategy into the EM algorithm in an effort to surpass this limitation. Lu, Niu and Liu (2003) used the EM for haplotype analysis in the scenario of case-control studies. Kang et al. (2004) extended the traditional EM haplotype inference algorithm by incorporating genotype uncertainty. Niu (2004) gave a review of general algorithms for haplotype reconstruction.

6. FINITE MIXTURE CLUSTERING FOR MICROARRAY DATA

In cluster analysis one seeks to partition observed data into groups such that coherence within each group and separation between groups are maximized jointly. Although this goal is subjectively defined (depending on how one defines "coherence" and "separation"), clustering can serve as an initial exploratory analysis for high-dimensional data. One example in computational biology is microarray data analysis. Microarrays are used to measure the mRNA expression levels of thousands of genes at the same time. Microarray data are usually displayed as a matrix $\mathbf{Y}$. The rows of $\mathbf{Y}$ represent the genes in a study and the columns are arrays obtained in different experiment conditions, in different stages of a biological system or from different biological samples. Cluster analysis of microarray data has been a hot research field because groups of genes that share similar expression patterns (clustering the rows of $\mathbf{Y}$) are often involved in the same or related biological functions, and groups of samples having a similar gene expression profile (clustering the columns of $\mathbf{Y}$) are often indicative of the relatedness of these samples (e.g., the same cancer type).

Finite mixture models have long been used in cluster analysis (see Fraley and Raftery, 2002 for a review). The observations are assumed to be generated from a finite mixture of distributions. The likelihood of a mixture model with K components can be written as

$$P(\mathbf{Y}|\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K; \tau_1, \ldots, \tau_K) = \prod_{i=1}^{n} \sum_{k=1}^{K} \tau_k f_k(\mathbf{Y}_i|\boldsymbol{\theta}_k),$$

where f_k is the density function of the kth component in the mixture, $\boldsymbol{\theta}_k$ are the corresponding parameters, and τ_k is the probability that an observed datum is generated from this component model ($\tau_k \geq 0, \sum_k \tau_k = 1$). One of the most commonly used finite mixture models is the Gaussian mixture model, in which $\boldsymbol{\theta}_k$ is composed of mean $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$. Outliers can be accommodated by a special component in the mixture that allows for a larger variance or extreme values.

A standard way to simplify the statistical computation with mixture models is to introduce a variable indicating which component an observation $\mathbf{Y}_i$ was generated from. Thus, the "complete data" can be expressed as $\mathbf{X}_i = (\mathbf{Y}_i, \boldsymbol{\Gamma}_i)$, where $\boldsymbol{\Gamma}_i = (\gamma_{i1}, \ldots, \gamma_{iK})$, and $\gamma_{ik} = 1$ if $\mathbf{Y}_i$ is generated by the kth component and $\gamma_{ik} = 0$ otherwise. The complete-data log-likelihood function is

$$\log P(\mathbf{Y}, \boldsymbol{\Gamma} | \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K; \tau_1, \ldots, \tau_K) = \sum_{i=1}^{n} \sum_{k=1}^{K} \gamma_{ik} \log[\tau_k f_k(\mathbf{Y}_i | \boldsymbol{\theta}_i)].$$

Since the complete-data log-likelihood function is linear in the γ_{jk}'s, in the E-step we only need to compute

$$\hat{\gamma}_{ik} \equiv E(\gamma_{ik} | \boldsymbol{\Theta}^{(t)}, \mathbf{Y}) = \frac{\tau_k^{(t)} f_k(\mathbf{Y}_i | \boldsymbol{\theta}_k^{(t)})}{\sum_{j=1}^{K} \tau_j^{(t)} f_j(\mathbf{Y}_i | \boldsymbol{\theta}_j^{(t)})}.$$

The Q-function can be calculated as

$$Q(\boldsymbol{\Theta} | \boldsymbol{\Theta}^{(t)}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \hat{\gamma}_{ik} \log[\tau_k f_k(\mathbf{Y}_i | \boldsymbol{\theta}_i)]. \tag{2}$$

The M-step updates the component probability τ_k as

$$\tau_k^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} \hat{\gamma}_{ik},$$

and the updating of $\boldsymbol{\theta}_k$ would depend on the density function. In mixture Gaussian models, the Q-function is quadratic in the mean vector and can be maximized to achieve the M-step.

Yeung et al. (2001) are among the pioneers who applied the model-based clustering method in microarray data analysis. They adopted the Gaussian mixture model framework and represented the covariance matrix in terms of its eigenvalue decomposition

$$\boldsymbol{\Sigma}_k = \lambda_k D_k A_k D_k^T.$$

In this way, the orientation, shape and volume of the multivariate normal distribution for each cluster can be modeled separately by eigenvector matrix D_k, eigenvalue matrix A_k and scalar λ_k, respectively. Simplified models are straightforward under this general model setting, such as setting λ_k, D_k or A_k to be identical for all clusters or restricting the covariance matrices to take some special forms (e.g., $\boldsymbol{\Sigma}_k = \lambda_k I$). Yeung and colleagues used the EM algorithm to estimate the model parameters. To improve convergence, the EM algorithm can be initialized with a model-based hierarchical clustering step (Dasgupta and Raftery, 1998).

When $\mathbf{Y}_i$ has some dimensions that are highly correlated, it can be helpful to project the data onto a lower-dimensional subspace. For example, McLachlan, Bean and Peel (2002) attempted to cluster tissue samples instead of genes. Each tissue sample is represented as a vector of length equal to the number of genes, which can be up to several thousand. Factor analysis (Ghahramani and Hinton, 1997) can be used to reduce the dimensionality, and can be seen as a Gaussian model with a special constraint on the covariance matrix. In their study, McLachlan, Bean and Peel used a mixture of factor analyzers, equivalent to a mixture Gaussian model, but with fewer free parameters to estimate because of the constraints. A variant of the EM algorithm, the Alternating Expectation–Conditional Maximization (AECM) algorithm (Meng and van Dyk, 1997), was applied to fit this mixture model.

Many microarray data sets are composed of several arrays in a series of time points so as to study biological system dynamics and regulatory networks (e.g., cell cycle studies). It is advantageous to model the gene expression profile by taking into account the smoothness of these time series. Ji et al. (2004) clustered the time course microarray data using a mixture of HMMs. Bar-Joseph et al. (2002) and Luan and Li (2003) implemented mixture models with spline components. The time-course expression data were treated as samples from a continuous smooth process. The coefficients of the spline bases can be either fixed effect, random effect or a mixture effect to accommodate different modeling needs. Ma et al. (2006) improved upon these methods by adding a gene-specific effect into the model:

$$y_{ij} = \mu_k(t_{ij}) + b_i + \varepsilon_{ij},$$

where $\mu_k(t)$ is the mean expression of cluster k at time t, composed of smoothing spline components; $b_i \sim N(0, \sigma_{bk}^2)$ explains the gene specific deviation from the cluster mean; and $\varepsilon_{ij} \sim N(0, \sigma^2)$ is the

measurement error. The Q-function in this case is a weighted version of the penalized log-likelihood:

$$-\sum_{k=1}^{K}\Biggl\{\sum_{i=1}^{n}\hat{\gamma}_{ik}\Biggl(\sum_{j=1}^{T}\frac{(y_{ij}-\mu_k(t_{ij})-b_i)^2}{2\sigma^2}+\frac{b_i^2}{2\sigma_{bk}^2}\Biggr) - \lambda_k T\int[\mu_k''(t)]^2\,dt\Biggr\}, \tag{3}$$

where the integral is the smoothness penalty term. A generalized cross-validation method was applied to choose the values for σ_{bk}^2 and λ_k.

An interesting variation on the EM algorithm, the rejection-controlled EM (RCEM), was introduced in Ma et al. (2006) to reduce the computational complexity of the EM algorithm for mixture models. In all mixture models, the E-step computes the membership probabilities (weights) for each gene to belong to each cluster, and the M-step maximizes a weighted sum function as in Luan and Li (2003). To reduce the computational burden of the M-step, we can "throw away" some terms with very small weights in an unbiased weight using the rejection control method (Liu, Chen and Wong, 1998). More precisely, a threshold c (e.g., $c = 0.05$) is chosen. Then, the new weights are computed as

$$\tilde{\gamma}_{ik}=\begin{cases}\max\{\hat{\gamma}_{ik},c\}, & \text{with probability } \min\{1,\hat{\gamma}_{ik}/c\},\\ 0, & \text{otherwise.}\end{cases}$$

The new weight $\tilde{\gamma}_{ik}$ then replaces the old weight $\hat{\gamma}_{ik}$ in the Q-function calculation in (2) in general, and in (3) more specifically. For cluster k, genes with a membership probability higher than c are not affected, while the membership probabilities of other genes will be set to c or 0, with probabilities $\hat{\gamma}_{ik}/c$ and $1-\hat{\gamma}_{ik}/c$, respectively. By giving a zero weight to many genes with low $\hat{\gamma}_{ik}/c$, the number of terms to be summed in the Q-function is greatly reduced.

In many ways finite mixture models are similar to the K-means algorithm, and they may produce very similar clustering results. However, finite mixture models are more flexible in the sense that the inferred clusters do not necessarily have a sphere shape, and the shapes of the clusters can be learned from the data. Researchers such as Suresh, Dinakaran and Valarmathie (2009) tried to combine the two ways of thinking to make better clustering algorithms.

For cluster analysis, one intriguing question is how to set the total number of clusters. Bayesian information criterion (BIC) is often used to determine the number of clusters (Yeung et al., 2001; Fraley and Raftery, 2002; Ma et al., 2006). A random subsampling approach is suggested by Dudoit, Fridlyand and Speed (2002) for the same purpose. When external information of genes or samples is available, cross-validation can be used to determine the number of clusters.

7. TRENDS TOWARD INTEGRATION

Biological systems are generally too complex to be fully characterized by a snapshot from a single viewpoint. Modern high-throughput experimental techniques have been used to collect massive amounts of data to interrogate biological systems from various angles and under diverse conditions. For instance, biologists have collected many types of genomic data, including microarray gene expression data, genomic sequence data, ChIP–chip binding data and protein–protein interaction data. Coupled with this trend, there is a growing interest in computational methods for integrating multiple sources of information in an effort to gain a deeper understanding of the biological systems and to overcome the limitations of divided approaches. For example, the Phylo-HMM in Section 4 takes as input an alignment of multiple sequences, which, as shown in Section 3, is a hard problem by itself. On the other hand, the construction of the alignment can be improved a lot if we know the underlying phylogeny. It is therefore preferable to infer the multiple alignment and the phylogenetic tree jointly (Lunter et al., 2005).

Hierarchical modeling is a principled way of integrating multiple data sets or multiple analysis steps. Because of the complexity of the problems, the inclusion of nuisance parameters or missing data at some level of the hierarchical models is usually either structurally inevitable or conceptually preferable. The EM algorithm and Markov chain Monte Carlo algorithms are often the methods of choice for these models due to their close connection with the underlying statistical model and the missing data structure.

For example, EM algorithms have been used to combine motif discovery with evolutionary information. The underlying logic is that the motif sites such as TFBSs evolved slower than the surrounding genomic sequences (the background) because of functional constraints and natural selection. Moses, Chiang and Eisen (2004) developed EMnEM (Expectation–Maximization on Evolutionary Mixtures), which is a generalization of the mixture model formulation for motif discovery (Bailey and Elkan, 1994). More precisely, they treat an alignment of multiple orthologous sequences as a series of alignments of length w,

each of which is a sample from the mixture of a motif model and a background model. All observed sequences are assumed to evolve from a common ancestor sequence according to an evolutionary process parameterized by a Jukes–Cantor substitution matrix. PhyME (Sinha, Blanchette and Tompa, 2004) is another EM approach for motif discovery in orthologous sequences. Instead of modeling the common ancestor, they modeled one designated "reference species" using a two-state HMM (motif state or background state). Only the well-aligned part of the reference sequence was assumed to share a common evolutionary origin with other species. PhyME assumes a symmetric star topology instead of a binary phylogenetic tree for the evolutionary process. OrthoMEME (Prakash et al., 2004) deals with pairs of orthologous sequences and is a natural extension of the EM algorithm of Lawrence and Reilly (1990) described in Section 2.

Steps have also been taken to incorporate microarray gene expression data into motif discovery (Bussemaker, Li and Siggia, 2001; Conlon et al., 2003). Kundaje et al. (2005) used a graphical model and the EM algorithm to combine DNA sequence data with time-series expression data for gene clustering. Its basic logic is that co-regulated genes should show both similar TFBS occurrence in their upstream sequences and similar gene-expression time-series curves. The graphical model assumes that the TFBS occurrence and gene-expression are independent, conditional on the co-regulation cluster assignment. Based on predicted TFBSs in promoter regions and cell-cycle time-series gene-expression data on budding yeast, this algorithm infers model parameters by integrating out the latent variables for cluster assignment. In a similar setting, Chen and Blanchette (2007) used a Bayesian network and an EM-like algorithm to integrate TFBS information, TF expression data and target gene expression data for identifying the combination of motifs that are responsible for tissue-specific expression. The relationships among different data are modeled by the connections of different nodes in the Bayesian network. Wang et al. (2005) used a mixture model to describe the joint probability of TFBS and target gene expression data. Using the EM algorithm, they provide a refined representation of the TFBS and calculate the probability that each gene is a true target.

As we show in this review, the EM algorithm has enjoyed many applications in computational biology. This is partly driven by the need for complex statistical models to describe biological knowledge and data. The missing data formulation of the EM algorithm addresses many computational biology problems naturally. The efficiency of a specific EM algorithm depends on how efficiently we can integrate out unobserved variables (missing data/nuisance parameters) in the E-step and how complex the optimization problem is in the M-step. Special dependence structures can often be imposed on the unobserved variables to greatly ease the computational burden of the E-step. For example, the computation is simple if latent variables are independent in the conditional posterior distribution, such as in the mixture motif example in Section 2 and the haplotype example in Section 5. Efficient exact calculation may also be available for structured latent variables, such as the forward–backward procedure for HMMs (Baum et al., 1970), the pruning algorithm for phylogenetic trees (Felsenstein, 1981) and the inside–outside algorithm for the probabilistic context-free grammar in predicting RNA secondary structures (Eddy and Durbin, 1994). As one of the drawbacks of the EM algorithm, the M-step can sometimes be too complicated to compute directly, such as in the PhyloHMM example in Section 4 and the smoothing spline mixture model in Section 6, in which cases innovative numerical tricks are called for.

ACKNOWLEDGMENTS

We thank Paul T. Edlefsen for helpful discussions about the profile hidden Markov model, as well as to Yves Chretien for polishing the language. This research is supported in part by the NIH Grant R01-HG02518-02 and the NSF Grant DMS-07-06989. The first two authors should be regarded as joint first authors.

REFERENCES

BAILEY, T. L. and ELKAN, C. (1994). Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proc. Int. Conf. Intell. Syst. Mol. Biol.* **2** 28–36.

BAILEY, T. L. and ELKAN, C. (1995a). Unsupervised learning of multiple motifs in biopolymers using EM. *Machine Learning* **21** 51–58.

BAILEY, T. L. and ELKAN, C. (1995b). The value of prior knowledge in discovering motifs with MEME. *Proc. Int. Conf. Intell. Syst. Mol. Biol.* **3** 21–29.

BALDI, P. and CHAUVIN, Y. (1994). Smooth on-line learning algorithms for hidden Markov models. *Neural Computation* **6** 305–316.

BAR-JOSEPH, Z., GERBER, G., GIFFORD, D., JAAKKOLA, T. and SIMON, I. (2002). A new approach to analyzing gene expression time series data. In *Proc. Sixth Ann. Inter. Conf. Comp. Biol.* 39–48. ACM Press, New York.

BARTON, G. and STERNBERG, M. (1987). A strategy for the rapid multiple alignment of protein sequences. *J. Mol. Biol.* **198** 327–337.

BATZOGLOU, S. (2005). The many faces of sequence alignment. *Briefings in Bioinformatics* **6** 6–22.

BAUM, L. E., PETRIE, T., SOULES, G. and WEISS, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Ann. Math. Statist.* **41** 164–171. MR0287613

BOFFELLI, D., MCAULIFFE, J., OVCHARENKO, D., LEWIS, K. D., OVCHARENKO, I., PACHTER, L. and RUBIN, E. M. (2003). Phylogenetic shadowing of primate sequences to find functional regions of the human genome. *Science* **299** 1391–1394.

BRUNO, W. (1996). Modeling residue usage in aligned protein sequences via maximum likelihood. *Mol. Biol. Evol.* **13** 1368–1374.

BUSSEMAKER, H. J., LI, H. and SIGGIA, E. D. (2001). Regulatory element detection using correlation with expression. *Nature Genetics* **27** 167–171.

CARDON, L. R. and STORMO, G. D. (1992). Expectation maximization algorithm for identifying protein-binding sites with variable lengths from unaligned DNA fragments. *J. Mol. Biol.* **223** 159–170.

CEPPELLINI, R., SINISCALCO, M. and SMITH, C. A. B. (1955). The estimation of gene frequencies in a random-mating population. *Annals of Human Genetics* **20** 97–115. MR0075523

CHEN, X. and BLANCHETTE, M. (2007). Prediction of tissue-specific cis-regulatory modules using Bayesian networks and regression trees. *BMC Bioinformatics* **8** (Suppl 10) S2.

CHIANO, M. N. and CLAYTON, D. G. (1998). Fine genetic mapping using haplotype analysis and the missing data problem. *Annals of Human Genetics* **62** 55–60.

CHURCHILL, G. A. (1989). Stochastic models for heterogeneous DNA sequences. *Bull. Math. Biol.* **51** 79–94. MR0978904

CONLON, E. M., LIU, X. S., LIEB, J. D. and LIU, J. S. (2003). Integrating regulatory motif discovery and genome-wide expression analysis. *Proc. Natl. Acad. Sci. USA* **100** 3339–3344.

DASGUPTA, A. and RAFTERY, A. (1998). Detecting features in spatial point processes with clutter via model-based clustering. *J. Amer. Statist. Assoc.* **93** 294–302.

DEMPSTER, A., LAIRD, N. and RUBIN, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Statist. Soc. Ser. B* **39** 1–38. MR0501537

DENG, M., MEHTA, S., SUN, F. and CHEN, T. (2002). Inferring domain–domain interactions from protein–protein interactions. *Genome Res.* **12** 1540–1548.

DO, C. B., MAHABHASHYAM, M. S. P., BRUDNO, M. and BATZOGLOU, S. (2005). Probcons: Probabilistic consistency-based multiple sequence alignment. *Genome Res.* **15** 330–340.

DUDOIT, S., FRIDLYAND, J. and SPEED, T. (2002). Comparison of discrimination methods for the classification of tumors using gene expression data. *J. Amer. Statist. Assoc.* **97** 77–87. MR1963389

DURBIN, R., EDDY, S., KROGH, A. and MITCHISON, G. (1998). *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge Univ. Press, Cambridge.

EDDY, S. R. (1998). Profile hidden Markov models. *Bioinformatics* **14** 755–763.

EDDY, S. R. and DURBIN, R. (1994). RNA sequence analysis using covariance models. *Nucleic Acids Res.* **22** 2079–2088.

EDGAR, R. (2004a). MUSCLE: A multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics* **5** 113.

EDGAR, R. (2004b). MUSCLE: Multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res.* **32** 1792–1797.

EDLEFSEN, P. T. (2009). Conditional Baum–Welch, dynamic model surgery, and the three Poisson Dempster–Shafer model. Ph.D. thesis, Dept. Statistics, Harvard Univ.

EXCOFFIER, L. and SLATKIN, M. (1995). Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population. *Mol. Biol. Evol.* **12** 921–927.

FAN, X., ZHU, J., SCHADT, E. and LIU, J. (2007). Statistical power of phylo-HMM for evolutionarily conserved element detection. *BMC Bioinformatics* **8** 374.

FELSENSTEIN, J. (1981). Evolutionary trees from DNA sequences: A maximum likelihood approach. *J. Mol. Evol.* **17** 368–376.

FELSENSTEIN, J. and CHURCHILL, G. A. (1996). A hidden Markov model approach to variation among sites in rate of evolution. *Mol. Biol. Evol.* **13** 93–104.

FENG, D. and DOOLITTLE, R. (1987). Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *J. Mol. Evol.* **25** 351–360.

FINN, R., MISTRY, J., SCHUSTER-BÖCKLER, B., GRIFFITHS-JONES, S., HOLLICH, V., LASSMANN, T., MOXON, S., MARSHALL, M., KHANNA, A., DURBIN, R., EDDY, S., SONNHAMMER, E. and BATEMAN, A. (2006). Pfam: Clans, web tools and services. *Nucleic Acids Res.* **Database Issue 34** D247–D251.

FRALEY, C. and RAFTERY, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *J. Amer. Statist. Assoc.* **97** 611–631. MR1951635

FRIEDMAN, N., NINIO, M., PE'ER, I. and PUPKO, T. (2002). A structural EM algorithm for phylogenetic inference. *J. Comput. Biol.* **9** 331–353.

GEMAN, S. and GEMAN, D. (1984). Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.* **6** 721–741.

GHAHRAMANI, Z. and HINTON, G. E. (1997). The EM algorithm for factor analyzers. Technical Report CRG-TR-96-1, Univ. Toronto, Toronto.

HAMPSON, S., KIBLER, D. and BALDI, P. (2002). Distribution patterns of over-represented k-mers in non-coding yeast DNA. *Bioinformatics* **18** 513–528.

HASTINGS, W. K. (1970). Monte Carlo sampling methods usings Markov chains and their applications. *Biometrika* **57** 97–109.

HAUSSLER, D., KROGH, A., MIAN, I. S. and SJOLANDER, K. (1993). Protein modeling using hidden Markov models: Analysis of globins. In *Proc. Hawaii Inter. Conf. Sys. Sci.* 792–802. IEEE Computer Society Press, Los Alamitos, CA.

HAWLEY, M. E. and KIDD, K. K. (1995). HAPLO: A program using the EM algorithm to estimate the frequencies of multi-site haplotypes. *Journal of Heredity* **86** 409–411.

HOLMES, I. (2005). Using evolutionary expectation maximization to estimate indel rates. *Bioinformatics* **21** 2294–2300.

HOLMES, I. and RUBIN, G. M. (2002). An expectation maximization algorithm for training hidden substitution models. *J. Mol. Biol.* **317** 753–764.

HUGHEY, R. and KROGH, A. (1996). Hidden Markov models for sequence analysis. Extension and analysis of the basic method. *Comput. Appl. Biosci.* **12** 95–107.

JENSEN, S. T., LIU, X. S., ZHOU, Q. and LIU, J. S. (2004). Computational discovery of gene regulatory binding motifs: A Bayesian perspective. *Statist. Sci.* **19** 188–204. MR2082154

JI, H. and WONG, W. H. (2006). Computational biology: Toward deciphering gene regulatory information in mammalian genomes. *Biometrics* **62** 645–663. MR2247187

JI, X., YUAN, Y., SUN, Z. and LI, Y. (2004). HMMGEP: Clustering gene expression data using hidden Markov models. *Bioinformatics* **20** 1799–1800.

KANG, H., QIN, Z. S., NIU, T. and LIU, J. S. (2004). Incorporating genotyping uncertainty in haplotype inference for single-nucleotide polymorphisms. *American Journal of Human Genetics* **74** 495–510.

KAROLCHIK, D., BAERTSCH, R., DIEKHANS, M., FUREY, T. S., HINRICHS, A., LU, Y. T., ROSKIN, K. M., SCHWARTZ, M., SUGNET, C. W., THOMAS, D. J., WEBER, R. J., HAUSSLER, D. and KENT, W. J. (2003). The UCSC genome browser database. *Nucleic Acids Res.* **31** 51–54.

KARPLUS, K., BARRETT, C. and HUGHEY, R. (1999). Hidden Markov models for detecting remote protein homologies. *Bioinformatics* **14** 846–856.

KATOH, K., KUMA, K., TOH, H. and MIYATA, T. (2005). MAFFT version 5: Improvement in accuracy of multiple sequence alignment. *Nucleic Acids Res.* **33** 511–518.

KROGH, A., BROWN, M., MIAN, I. S., SJOLANDER, K. and HAUSSLER, D. (1994). Hidden Markov models in computational biology applications to protein modeling. *J. Mol. Biol.* **235** 1501–1531.

KROGH, A., MIAN, I. S. and HAUSSLER, D. (1994). A hidden Markov model that finds genes in *E. coli* DNA. *Nucleic Acids Res.* **22** 4768–4778.

KUNDAJE, A., MIDDENDORF, M., GAO, F., WIGGINS, C. and LESLIE, C. (2005). Combining sequence and time series expression data to learn transcriptional modules. *IEEE/ACM Trans. Comp. Biol. Bioinfo.* **2** 194–202.

LANDER, E. S. and GREEN, P. (1987). Construction of multilocus genetic linkage maps in humans. *Proc. Natl. Acad. Sci. USA* **84** 2363–2367.

LAWRENCE, C. E. and REILLY, A. A. (1990). An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences. *Proteins* **7** 41–51.

LAWRENCE, C. E., ALTSCHUL, S. F., BOGUSKI, M. S., LIU, J. S., NEUWALD, A. F. and WOOTTON, J. C. (1993). Detecting subtle sequence signals: A Gibbs sampling strategy for multiple alignment. *Science* **262** 208–214.

LIU, J. S. (2001). *Monte Carlo Strategies in Scientific Computing*. Springer, New York. MR1842342

LIU, J. S., CHEN, R. and WONG, W. H. (1998). Rejection control and sequential importance sampling. *J. Amer. Statist. Assoc.* **93** 1022–1031. MR1649197

LIU, J. S., NEUWALD, A. F. and LAWRENCE, C. E. (1995). Bayesian models for multiple local sequence alignment and Gibbs sampling strategies. *J. Amer. Statist. Assoc.* **90** 1156–1170.

LIU, J. S., SABATTI, C., TENG, J., KEATS, B. J. and RISCH, N. (2001). Bayesian analysis of haplotypes for linkage disequilibrium mapping. *Genome Res.* **11** 1716–1724.

LIU, X. S., BRUTLAG, D. L. and LIU, J. S. (2002). An algorithm for finding protein-DNA binding sites with applications to chromatin-immunoprecipitation microarray experiments. *Nature Biotechnology* **20** 835–839.

LONG, J. C., WILLIAMS, R. C. and URBANEK, M. (1995). An E-M algorithm and testing strategy for multiple-locus haplotypes. *American Journal of Human Genetics* **56** 799–810.

LU, X., NIU, T. and LIU, J. S. (2003). Haplotype information and linkage disequilibrium mapping for single nucleotide polymorphisms. *Genome Res.* **13** 2112–2117.

LUAN, Y. and LI, H. (2003). Clustering of time-course gene expression data using a mixed-effects model with B-splines. *Bioinformatics* **19** 474–482.

LUNTER, G., MIKLOS, I., DRUMMOND, A., JENSEN, J. and HEIN, J. (2005). Bayesian coestimation of phylogeny and sequence alignment. *BMC Bioinformatics* **6** 83.

MA, P., CASTILLO-DAVIS, C., ZHONG, W. and LIU, J. (2006). A data-driven clustering method for time course gene expression data. *Nucleic Acids Res.* **34** 1261–1269.

MADERA, M. and GOUGH, J. (2002). A comparison of profile hidden Markov model procedures for remote homology detection. *Nucleic Acids Res.* **30** 4321–4328.

MCKENDRICK, A. G. (1926). Applications of mathematics to medical problems. *Proceedings Edinburgh Methematics Society* **44** 98–130.

MCLACHLAN, G. J., BEAN, R. W. and PEEL, D. (2002). A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics* **18** 413–422.

MENG, X. and VAN DYK, D. (1997). The EM algorithm—An old folk song sung to a fast new tune (with discussion). *J. Roy. Statist. Soc. Ser. B* **59** 511–567. MR1452025

MENG, X.-L. (1997). The EM algorithm and medical studies: A historical linik. *Statistical Methods in Medical Research* **6** 3–23.

MENG, X.-L. and PEDLOW, S. (1992). EM: A bibliographic review with missing articles. In *Proc. Stat. Comp. Sec.* 24–27. Amer. Statist. Assoc., Washington, DC.

METROPOLIS, N., ROSENBLUTH, A., ROSENBLUTH, M., TELLER, A. and TELLER, E. (1953). Equation of state calculations by fast computing machines. *Journal of Chemical Physics* **21** 1087–1092.

METROPOLIS, N. and ULAM, S. (1949). The Monte Carlo method. *J. Amer. Statist. Assoc.* **44** 335–341. MR0031341

MOSES, A., CHIANG, D. and EISEN, M. (2004). Phylogenetic motif detection by expectation–maximization on evolutionary mixtures. In *Pacific Symposium on Biocomputing* 324–335. World Scientific, Singapore.

NEUWALD, A. and LIU, J. (2004). Gapped alignment of protein sequence motifs through Monte Carlo optimization of a hidden Markov model. *BMC Bioinformatics* **5** 157.

NIU, T. (2004). Algorithms for inferring haplotypes. *Genetic Epidemiology* **27** 334–347.

NOTREDAME, C., HIGGINS, D. and HERINGA, J. (2000). T-Coffee: A novel method for fast and accurate multiple sequence alignment. *J. Mol. Biol.* **302** 205–217.

O'SULLIVAN, O., SUHRE, K., ABERGEL, C., HIGGINS, D. G. and NOTREDAME, C. (2004). 3DCoffee: Combining protein sequences and structures within multiple sequence alignments. *J. Mol. Biol.* **340** 385–395.

OTT, J. (1979). Maximum likelihood estimation by counting methods under polygenic and mixed models in human pedigrees. *American Journal of Human Genetics* **31** 161–175.

PAVESI, G., MEREGHETTI, P., MAURI, G. and PESOLE, G. (2004). Weeder Web: Discovery of transcription factor binding sites in a set of sequences from co-regulated genes. *Nucleic Acids Res.* **32** W199–W203.

PRAKASH, A., BLANCHETTE, M., SINHA, S. and TOMPA, M. (2004). Motif discovery in heterogeneous sequence data. In *Pacific Symposium on Biocomputing* 348–359. World Scientific, Singapore.

QIN, Z. S., NIU, T. and LIU, J. S. (2002). Partition–ligation–expectation–maximization algorithm for haplotype inference with single-nucleotide polymorphisms. *American Journal of Human Genetics* **71** 1242–1247.

RABINER, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE* **77** 257–286.

SIEPEL, A., BEJERANO, G., PEDERSEN, J. S., HINRICHS, A. S., HOU, M., ROSENBLOOM, K., CLAWSON, H., SPIETH, J., HILLIER, L. W., RICHARDS, S., WEINSTOCK, G. M., WILSON, R. K., GIBBS, R. A., KENT, W. J., MILLER, W. and HAUSSLER, D. (2005). Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome Res.* **15** 1034–1050.

SINHA, S. and TOMPA, M. (2002). Discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic Acids Res.* **30** 5549–5560.

SINHA, S., BLANCHETTE, M. and TOMPA, M. (2004). PhyME: A probabilistic algorithm for finding motifs in sets of orthologous sequences. *BMC Bioinformatics* **5** 170.

SMITH, C. A. B. (1957). Counting methods in genetical statistics. *Annals of Human Genetics* **35** 254–276. MR0088408

STORMO, G. D. and HARTZELL, G. W. I. (1989). Identifying protein-binding sites from unaligned DNA fragments. *Proc. Natl. Acad. Sci. USA* **86** 1183–1187.

SURESH, R. M., DINAKARAN, K. and VALARMATHIE, P. (2009). Model based modified K-means clustering for microarray data. In *International Conference on Information Management and Engineering* 271–273. IEEE Computer Society, Los Alamitos, CA.

TANNER, M. A. and WONG, W. H. (1987). The calculation of posterior distributions by data augmentation (with discussion). *J. Amer. Statist. Assoc.* **82** 528–540. MR0898357

TAVARÉ, S. (1986). Some probabilistic and statistical problems in the analysis of DNA sequences. In *Some Mathematical Questions in Biology—DNA Sequence Analysis (New York, 1984). Lectures on Mathematics in the Life Sciences* **17** 57–86. Amer. Math. Soc., Providence, RI. MR0846877

TAYLOR, W. (1988). A flexible method to align large numbers of biological sequences. *J. Mol. Evol.* **28** 161–169.

THOMPSON, E. A. (1984). Information gain in joint linkage analysis. *Math. Med. Biol.* **1** 31–49.

THOMPSON, J., HIGGINS, D. and GIBSON, T. (1994). CLUSTAL W: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.* **22** 4673–4680.

TOMPA, M., LI, N., BAILEY, T. L., CHURCH, G. M., DE MOOR, B., ESKIN, E., FAVOROV, A. V., FRITH, M. C., FU, Y., KENT, W. J., MAKEEV, V. J., MIRONOV, A. A., NOBLE, W. S., PAVESI, G., PESOLE, G., RÉGNIER, M., SIMONIS, N., SINHA, S., THIJS, G., VAN HELDEN, J., VANDENBOGAERT, M., WENG, Z., WORKMAN, C., YE, C. and ZHU, Z. (2005). Assessing computational tools for the discovery of transcription factor binding sites. *Nature Biotechnology* **23** 137–144.

WALLACE, I. M., BLACKSHIELDS, G. and HIGGINS, D. G. (2005). Multiple sequence alignments. *Current Opinion in Structural Biology* **15** 261–266.

WANG, W., CHERRY, J. M., NOCHOMOVITZ, Y., JOLLY, E., BOTSTEIN, D. and LI, H. (2005). Inference of combinatorial regulation in yeast transcriptional networks: A case study of sporulation. *Proc. Natl. Acad. Sci. USA* **102** 1998–2003.

WEEKS, D. E. and LANGE, K. (1989). Trials, tribulations, and triumphs of the EM algorithm in pedigree analysis. *Math. Med. Biol.* **6** 209–232. MR1052291

WOLFE, K. H., SHARP, P. M. and LI, W. H. (1989). Mutation rates differ among regions of the mammalian genome. *Nature* **337** 283–285.

YANG, Z. (1995). A space–time process model for the evolution of DNA sequences. *Genetics* **139** 993–1005.

YANG, Z. (1997). PAML: A program package for phylogenetic analysis by maximum likelihood. *Comput. Appl. Biosci.* **13** 555–556.

YEUNG, K. Y., FRALEY, C., MURUA, A., RAFTERY, A. E. and RUZZO, W. L. (2001). Model-based clustering and data transformations for gene expression data. *Bioinformatics* **17** 977–987.