Statistical Models for Motif Discovery

Jun S Liu*, Mayetri Gupta*, Xiaole Liu†, and Charles E. Lawrence‡

* Department of Statistics, Harvard University, Cambridge, MA 02138

† Medical Informatics, Stanford University, Stanford, CA 94305.

‡ The Wadsworth Center, Albany, NY 12201.

Abstract. With the completion of genomes of many species and the advances of microarray technologies, we begin to possess a tremendous amount of valuable biological data — but these "raw products" are still far from usable. One of the most challenging problems of this century is to decipher this huge amount of biological information and turn the data into knowledge. The past decade has witnessed a number of successful applications of sophisticated statistical models in computational biology. This article focuses on one of these success stories: using statistical methods to find short repetitive patterns in a set of DNA or protein sequences, a task often referred to as *motif discovery*. In particular, we review a few probabilistic models that have recently been shown useful for motif discovery and provide a novel framework based on a Bayesian segmentation model to unify these approaches. We show how to combine the dictionary model with the Gibbs sampler and how a segmentation-based motif sampler can be implemented. A few interesting open problems are also discussed.

1 Introduction

The human genome and many other genome sequencing projects have resulted in rapidly growing and publicly available databases of DNA and protein sequences (e.g., the GeneBank). The data in these databases are sequences of letters using d-letter ($d = 4$ for DNA or $d = 20$ for proteins) alphabets without punctuation or space characters. Recent advances in microarray technologies futher enable and "entice" biologists to generate a hugh amount of gene expression data. These data are real numbers and measure the relative changes of the mRNA products of many genes (sometimes all the genes in a genome, e.g., yeast or E. Coli). These real numbers are often presented as a $g \times c$ matrix, where g is the number of genes being monitored and c is the number of experimental conditions under investigation. One of the most interesting questions scientists are concerned with is how to get any useful biological information from "looking" at the sequence databases or the microarray data matrices. This task is often termed "data mining" for other types of data. The recent announcement of the near-completion of the human genome makes this interesting question more an urgent task for all interested scientists. However, "mining" a biopolymer database is noticeably different from mining other types of databases because (i) many sophisticated structures have been built in well-organized biopolymer databases, (ii) there is an enormous amount of biological knowledge, and (iii) fundamental laws in physics and chemistry can be applied. Consequently, more sophisticated mathematical/statistical models are often critical in developing a "mining" strategy.

Our focus here is is to find repetitive short sequence segments (called the *motif elements*) in a set of biological sequences. The main motivation for this task is that repetitive patterns in biopolymer sequences often correspond to functionally or structurally important parts of these molecules. For example, a common pattern shared by multiple proteins can often shed light on their functionalities. In Figure 1(b), one can see that a helical part of protein 3CRO is inside the major groove of a DNA double-helix structure. This helical part of the protein, often referred to as the *helix-turn-helix* motif, plays an important role in the binding of 3CRO to a DNA segment and turns out to be a rather conserved part in a large family of proteins responsible for gene regulation. Repetitive patterns in noncoding regions of DNA sequences also have important biological implications: They often correspond to a "regulatory binding site" to which a certain protein (such as 3CRO) binds to control the gene's expression. In Figure 1(b), the segment of DNA (double-helix structure) that are interacting with protein 3CRO is a *binding site* whose pattern is conserved among the $5'$ untranslated regions of a number of genes regulated by the CRO protein. It has been shown that the gene expression data from microarray experiments can furhter assist the discovery of biologically relevant regulatory motifs [11, 6, 9].

The multiple occurrences of a motif pattern in a set of sequences R is thus analogous to the multiple occurrences of a *common word* in an article. What makes things complicated is that although the word (or words) is known to have occurred multiple times in the text, each of its occurrence is not identical to another. In other words, there are often "typos" (sometimes very serious ones) in each occurrence of the word. It is therefore rather natural for us to employ a probabilistic model to describe what a "motif pattern" is (e.g., it is a stochastic word) and let basic statistical principles (i.e., the Bayesian methodology) to guide us in the discovery of these patterns.

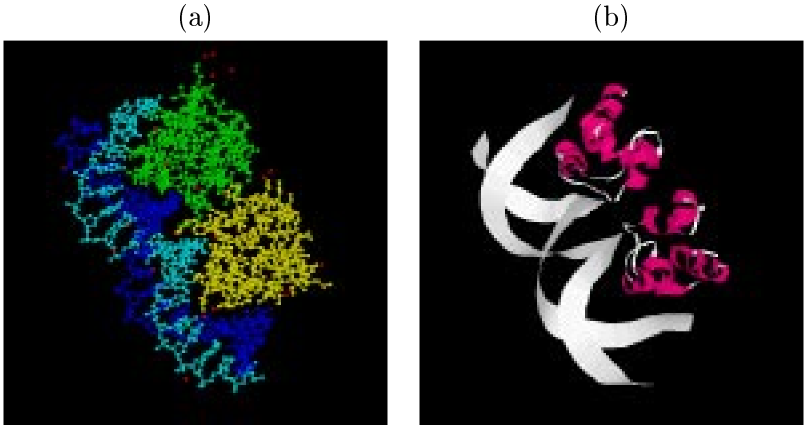

Figure 1: (a) A ball-and-stick plot of the interaction between a regulatory protein, 3CRO, and the DNA segment to which it binds. (b) The same structure as in (a), but expressed by a ribbon representation widely used in the protein structure modeling community.

This article reviews two general statistical approaches: one is based on the product multinomial model (also called the weight matrix) for the motif pattern (Sections 2 and 3) and another based on a segmentation model (Section 4). It is shown that these two models are very much related to each other. A segmentation-based Gibbs sampler is implemented and shown to outperform the original Gibbs motif sampler [8].

2 First-generation Block-Motif model

This model, as depicted in Figure 2, gives rise to the first Gibbs sampling-based motif finding algorithm proposed in [4]. The algorithm will be referred to as the "site sampler" in the later context. In this model, we assume that there are K DNA (or protein) sequences of lengths $n_1, \ldots, n_K$, respectively, in consideration. Each sequence is assumed to be generated by i.i.d. draws from the alphabet $\{A, C, G, T\}$ with the frequency vector $\boldsymbol{\theta}_0 = (\theta_{0a}, \ldots, \theta_{0t})$, except for

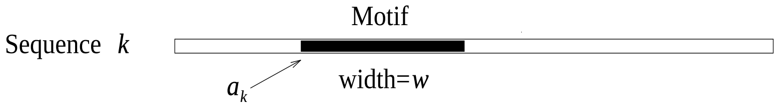

Figure 2: A schematic plot of the block-motif model used for our pattern finding.

a segment, as illustrated by the blackened region in Figure 2, which is a *motif* element of length w. The *motif* is a frequency matrix $\boldsymbol{\Theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_w)$, where each $\boldsymbol{\theta}_j^T$ is a probability vector of length 4 representing the preference of the nucleotide types at position j of a motif element. Thus, position j of a motif element (or an occurrence of the motif) is a draw from the alphabet with frequency vector $\boldsymbol{\theta}_j$. We know neither the motif matrix $\boldsymbol{\Theta}$ nor the positions of the motif elements, but we are interested in finding them out.

2.1 Site sampler

An intuitive solution (for a statistician) to the motif discovery problem just described is to use a missing-data formulation: One can view the position of the motif element, a_j in particular, as missing data. Then one can design an EM algorithm [5] or a data augmentation (DA) algorithm [13]. A slight modification [7] of the DA algorithm gives rise to the so-called *motif site sampler* [4].

In the *site sampler*, the motif locations (sites) are initialized at random; that is, position $a_k^{(0)}$ (for $k = 1, \ldots, K$) is a randomly chosen position of the kth sequence. For $t = 1, \ldots, m$, the algorithm iterates the following steps:

- Select a sequence, say the kth sequence, either deterministically or at random.
- Draw a new motif location a_k according to the predictive distribution

  $$P(a_k \mid a_1^{(t)}, \ldots, a_{k-1}^{(t)}, a_{k+1}^{(t)}, \ldots, a_K^{(t)}) \tag{1}$$

  and update the current motif location $a_k^{(t)}$ to $a_k^{(t+1)} = a_k$.
- Let $a_j^{(t+1)} = a_j^{(t)}$ for $j \neq k$.

Although there are many choices of the predictive distribution function used for updating the alignment [e.g., one can let $P(a_k \mid \ldots)$ be proportional to certain fitness measure of the segment indexed by a_k to the current multinomial profile resulting from the $a_j^{(t)}$, $j \neq k$], those that make the foregoing iteration consistent have to be the ones derived from a complete Bayesian statistical model.

2.2 Some theoretical calculation

To understand the nature of the problem, we consider an analogous problem. Consider K sequences of coins of length n each. For each sequence S_k, we assume that starting from an unknown position a_k there is a segment of w special coins, each has probability θ to show head. The remaining coins has a known probability θ_0 to show head. Our questions are two: (a) Can we estimate θ consistently? (b) How accurate can we predict the location of these special coins? To answer these questions, we start with the basic likelihood analysis. Let $\boldsymbol{Y} = (S_1, \ldots, S_K)$ be the sequence data. Then

$$P(\boldsymbol{Y} \mid \theta) = \prod_{k=1}^{K} \left(\sum_{i=1}^{n-w+1} \left(\frac{\theta}{\theta_0}\right)^{N_{k,i}} \left(\frac{1-\theta}{1-\theta_0}\right)^{w-N_{k,i}} \right),$$

where $N_{k,i}$ is the number of heads in segment $(s_{k,i}, \ldots, s_{k,i+w-1})$. Taking the first derivative of the log-likelihood with respect to θ, we obtain an estimating equation

$$\frac{A}{B} \equiv \sum_{k=1}^{K} \left[\frac{\sum\limits_{i=1}^{n-w+1} \left(\frac{N_{k,i}}{\theta} - \frac{w-N_{k,i}}{1-\theta}\right) \left(\frac{\theta}{\theta_0}\right)^{N_{k,i}} \left(\frac{1-\theta}{1-\theta_0}\right)^{w-N_{k,i}}}{\sum\limits_{i=1}^{n-w+1} \left(\frac{\theta}{\theta_0}\right)^{N_{k,i}} \left(\frac{1-\theta}{1-\theta_0}\right)^{w-N_{k,i}}} \right] = 0. \tag{2}$$

Let us consider the simplest case with $w = 1$. Then

$$\frac{\partial}{\partial\theta} \log P(\boldsymbol{Y} \mid \theta) = \sum_{i=1}^{k} \frac{\frac{N_k}{\theta_0} - \frac{n-N_k}{1-\theta_0}}{\frac{N_k}{\theta_0} + \theta\frac{n-N_k}{1-\theta_0}(1-\theta)}$$

Taking the second derivative, we have the observed Fisher information

$$I_{\text{obs}} = \sum_{k=1}^{K} \left(\frac{\frac{N_k}{\theta_0} - \frac{n-N_k}{1-\theta_0}}{\frac{N_k}{\theta_0}\theta + \frac{n-N_k}{1-\theta_0}(1-\theta)} \right)^2 .$$

If we assume that both K and n goes to infinity, we have

$$I_{\text{exp}} \approx \frac{k}{n\theta_0(1-\theta_0)}.$$

This implies that k/n needs to go to infinity for the estimation of θ to be consistent.

Now consider the case $w > 1$ where we have a more complex expression. In particular, for a single sequence S_k,

$$I_k = \frac{\sum\limits_{i=1}^{n-w+1} \left[\frac{N_{k,i}(N_{k,i}-1)}{\theta^2} - 2\frac{N_{k,i}(w-N_{k,i})}{\theta(1-\theta)} + \frac{(w-N_{k,i})(w-N_{k,i}-1)}{(1-\theta)^2}\right] \left(\frac{\theta}{\theta_0}\right)^{N_{k,i}} \left(\frac{1-\theta}{1-\theta_0}\right)^{w-N_{k,i}}}{\sum\limits_{i=1}^{n-w+1} \left(\frac{\theta}{\theta_0}\right)^{N_{k,i}} \left(\frac{1-\theta}{1-\theta_0}\right)^{w-N_{k,i}}} - \left(\frac{A}{B}\right)^2$$

where A and B are defined as in (2). This formula is not easy to simplify although not impossible. But for the general site-sampler model, this type of information analysis is still lacking.

3 Repetitive Block-Motif Model

There is no loss of generality here in that any biological sequence dataset can be viewed as a long sequence $\boldsymbol{S}$ of letters in an alphabet $\mathcal{A}$. Our focus is to find repetitive motif elements in the sequence.

A simple model that conveys the basic idea of a motif that repeats itself with random variations is the block-motif model as shown in Figure 3. It was first developed in [8]. and has been employed to find subtle repetitive patterns, such as helix-turn-helix structural motifs [10]. or gene regulation motifs [11], in both protein and DNA sequences. The repetitive patterns as represented by the dark solid rectangle occur irregularly in the dataset. The total number of occurrences of the motif is unknown. A simple first model is to assume that at any sequence position i, there is a small probability p_0 that a motif pattern starts from i. It is of interest to find the motif pattern and where it occurs.

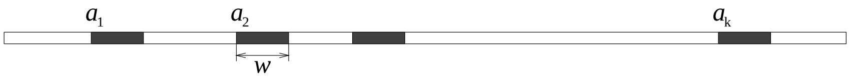

Figure 3: A graphical illustration of the repetitive motif model.

This model says that at unknown locations $A = (a_1, \ldots, a_K)$ there are repeated occurrences of a motif. So the sequence segments at these locations should look similar to each other. In other part of the sequence, called the *background*, the residues follow an independent multinomial model. Suppose the motif's width is w, we need $w + 1$ probability vectors to describe the motif and the background: $\boldsymbol{\theta}_0 = (\theta_{0a}, \ldots, \theta_{0t})$ describe the base frequencies in the background; and each $\boldsymbol{\theta}_k$ describes the base frequency at position k of the motif. The matrix $\Theta = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_w]$ is called the *profile* matrix for the motif. We again use the generic notation $\boldsymbol{\theta}$ to denote the collection of all parameters, $(\boldsymbol{\theta}_0, \Theta)$.

3.1 Scoring the motif candidates

A Bayesian solution to the foregoing alignment problem was derived in [8]. With a Dirichlet prior Dirichlet$(\boldsymbol{\alpha})$, for all the $\boldsymbol{\theta}_i$, we can obtain the Bayes estimates of the $\boldsymbol{\theta}_i$ very easily *if* we know the positions of the motif. To facilitate analysis, we introduce an indicator vector $\boldsymbol{I} = (I_1, \ldots, I_n)$ and treat it as *missing data*. An $I_i = 1$ means that position i is the start of a motif pattern, and $I_i = 0$ means otherwise. We assume *a priori* each I_i has a small probability

p_0 to be equal to 1. With this setup, we can write down the joint posterior distribution:

$$p(\boldsymbol{\theta}, \boldsymbol{I} \mid R) \propto p(R \mid \boldsymbol{I}, \boldsymbol{\theta}) p(\boldsymbol{I} \mid \boldsymbol{\theta}) f_0(\boldsymbol{\theta}) \tag{3}$$

Suppose that the alphabet size is k_0 (=4 for DNA sequences), motif length is w, and the motif pattern can be described by a $k_0 \times w$ weight matrix $\boldsymbol{\Theta}$. Assume also that the background of non-motif regions can be described by a multinomial vector $\boldsymbol{\theta}_0$, which is assumed known in advance. Let $A = (a_1, \ldots, a_k)$ be the location vector of the k occurrences of the motif. Then,

$$\begin{aligned} \log p(\boldsymbol{S}, A) &= \log \int p(\boldsymbol{S} \mid A, \boldsymbol{\Theta}, \boldsymbol{\theta}_0) p(\Theta) p(A) d\boldsymbol{\Theta} \\ &\approx \log p(A) + \log p(\boldsymbol{S} \mid \boldsymbol{\theta}_0) + |A| \sum_{j=1}^{w} \langle \hat{\boldsymbol{\theta}}_j, \log \frac{\hat{\boldsymbol{\theta}}_j}{\boldsymbol{\theta}_0} \rangle, \end{aligned}$$

where $|A| = k$ is the number of sites. Since $P(S \mid \boldsymbol{\theta}_0)$ is constant for all A, the Gibbs motif sampler developed in [8] optimizes the score function

$$\psi(\boldsymbol{S}) = |A| \left[\log p_0 + \sum_{j=1}^{w} I_{\text{ent}}(\hat{\boldsymbol{\theta}}_j \| \boldsymbol{\theta}_0) \right],$$

where the entropy distance I_{ent} between two discrete distributions $\boldsymbol{p} = (p_1, \ldots, p_{k_0})$ and $\boldsymbol{q} = (q_1, \ldots, q_{k_0})$, is defined as

$$I(\boldsymbol{p} \| \boldsymbol{q}) = \langle \boldsymbol{p}, \log \frac{\boldsymbol{p}}{\boldsymbol{q}} \rangle \equiv \sum_{i=1}^{k_0} p_i \log(p_i / q_i).$$

If we also assume that p_0 is unknown and given a prior distribution $f(p_0)$ (say, a Beta(a_0, b_0) distribution), then

$$\begin{aligned} p(A) &\approx \int p_0^{|A|} (1 - p_0)^{N - w|A|} f(p_0) dp_0 \\ &= \frac{\Gamma(a_0 + b_0)}{\Gamma(a_0)\Gamma(b_0)} \int p_0^{|A| + a_0 - 1} (1 - p_0)^{N - w|A| + b_0 - 1} dp_0 \\ &= \frac{\Gamma(a_0 + b_0)\Gamma(|A| + a_0)\Gamma(N - w|A| + b_0)}{\Gamma(a_0)\Gamma(b_0)\Gamma(N - (w - 1)|A| + a_0 + b_0)} \end{aligned}$$

Then the new score function becomes

$$\psi'(\boldsymbol{S}) = \log p(A) + \sum_{j=1}^{w} I_{\text{ent}}(\hat{\boldsymbol{\theta}}_j \| \boldsymbol{\theta}_0).$$

It is of interest to see the behavior of such a scoring function under the null distribution. That is, suppose all the data are from the background, how the scoring function fairs. For example, an alternative score is

$$\phi(\boldsymbol{S}) = \log(|A|) \sum_{j=1}^{w} I_{\text{ent}}(\hat{\boldsymbol{\theta}}_j \| \boldsymbol{\theta}_0).$$

This scoring function allows the total number of sites $|A|$ to play a role when $|A|$ is relatively small, but its role decreases as $|A|$ increases. In other words, the new potential site is added to the collection only if it can increase the total information.

3.2 Searching strategy

The popular searching strategies include the progressive comparison method employed by CONSENSUS [12], expectation-maximization based deterministic search in MEME [5, 2], and the iterative stochastic search in Gibbs Motif Sampler ([8], renamed as AlignAce in [11]). Other word frequency based approaches do not seem to achieve a comparable success rate to CONSENSUS, MEME, and the Gibbs Motif Sampler (or AlignAce). The predictive updating approach used in GMS is of particular interest. More precisely, at each iteration, one uses the current "weight matrix" for the motif pattern to score every segment of width w of all the sequences in the whole dataset and select those "significant" candidates to form a collection of possible motif sites. Then a new weight matrix is computed based on these candidate motif sites. In order to avoid being trapped in a local mode, GMS uses a probabilistic rule to decide whether a sequence segment being examined should be included as a potential motif site or not.

A central question, then, is how to judge "significance" and how to incorporate additional information revealed by, say, cross-species comparisons, gene expression clusterings, or data from the chromatin-immunoprecipitation and microarray hybridization (ChIP-array, see Section 5). A natural route is to build an appropriate statistical model to reflect these knowledges and to construct the search algorithms accordingly. For example, if a sequence segment is located in a region where a cross-species comparison shows that it is highly conserved, then it is highly likely that the segment corresponds to a protein binding site. Otherwise, such a *prior* probability would be small. In the IP experiment, the few 5' UTRs with the highest enrichment level and gene expression levels [1] clearly are most likely to contain the targeted binding sites, and maybe multiple of them. Hence, it is essential to direct the GMS to search these few sequences more thoroughly before it wanders off to other less likely sequences.

A new search strategy as implemented by BIOPROSPECTOR, called the "threshold sampler," is as follows: for each sequence, we give it a high threshold and a low threshold. When we use the current motif pattern matrix to scan the sequence, all segments whose score are above the high threshold are automatically called a motif site and all those that are below the high threshold but above the low threshold are given a chance to be sampled into the set of motif sites. The low-threshold is started as 0 and increase gradually to a suitable level.

Another search strategy is similar to the one used in CONSENSUS: for each candidate k-mers appeared in each sequence, we search the whole dataset to count the number of length-k segments in the data set that have at least c base pairs that match with the seed $k$mer and such segments are called the c-matches. Here c is determined by a p-value computation as follows.

For a pair of randomly generated base pairs, the probability that they match each other is

$$p_m = p_a^2 + p_c^2 + p_g^2 + p_t^2.$$

For the yeast intergentic regions, $p_m \approx .2644$. Hence, the number L of matched positions between two randomly generated k-mers is a binomial random variable, i.e.,

$$P(L = l) = \binom{k}{l} p_m^l (1 - p_m)^l.$$

The cutoff value c is chosen so that $P(L \geq c) \equiv p_c$ is sufficiently small. More precisely, for any given seed k-mer, the number of c-matches to this k-mer in a sequence of length N is approximately distributed as Poisson(Np_c). Hence, by rough approximation, the chance that any given seed k-mer can find at least j_0 c-matches in the dataset is

$$P(N_c \geq j_0) = 1 - \sum_{j=1}^{j_0 - 1} \frac{(Np_c)^j}{j!} e^{-Np_c}.$$

If let $Np_c \approx 1$, then the chance of having more than 5 c-matches for a given k-mer is about 1 out of 2,000.

Empirically, we would choose $p_c \leq 0.002$. For $k = 7, \ldots, 20$, we have

7	8	9	10	11	12	13	14	15	16	17	18	19	20
6	7	7	8	8	9	9	10	10	11	11	12	12	12

Of course, the use of this table is only for a guidance. For $k > 20$, one can use the normal approximation, i.e., $c = kp_m + 3\sqrt{kp_m(1 - p_m)}$. For the yeast genome, this formula is $c = 0.26k + 1.32\sqrt{k} + 1$.

4 Motif, Dictionary, and Segmentation

From Figure 3, it is also possible to view it as a segmentation model. That is, we can think of segmenting the sequences into two types of contiguous pieces, one described by the motif model, and the other by an iid model. In fact, this view can be further generalized into a dictionary model [3]. In this model, one assumes that a list of d words $\{M_1, M_2, \ldots, M_d\}$ are first given.

Then, with the observed sequence data $\boldsymbol{S}$, one can estimate the frequencies of these words. More precisely, under this model, we can write down the likelihood of the data:

$$P(\boldsymbol{S} \mid \boldsymbol{\theta}) = \sum_{\Pi} \prod_{i=1}^{N(\Pi)} \theta(\boldsymbol{S}[P_i]) = \sum_{\Pi} \prod_{j=1}^{d} \theta_j^{N_j(\Pi)}, \tag{4}$$

where $\Pi = (P_1, \ldots, P_k)$ is a partition of the sequence so that each part P_i corresponds to a word in the dictionary, $N(\Pi)$ is the total number of partitions in Π, and $N_j(\Pi)$ is the number of occurrences of word type M_j in the partition. Clearly, this can be viewed as a missing data problem where the partition Π is missing. The summation over all Π can be achieved recursively. Let $L_{i-1}(\boldsymbol{\theta})$ be the sum of all legitimate partitions for partial sequence $\boldsymbol{S}_{[1:(i-1)]}$. Then

$$L_i(\boldsymbol{\theta}) = \sum_{j=1}^{W} \theta(\boldsymbol{S}_{[(i-j):i]}) L_{i-j}, \tag{5}$$

where W is the length of the longest word in the dictionary. In other words, we check whether the last segment of length is a word from the dictionary for all possible word length j. To avoid minor complications, we assume that single letters are always contained in the dictionary (if not, the above recursion needs to be modified slightly).

Clearly, estimating θ from this model is conceptually simple. There are a few approaches: One can directly optimize (4) via a Newton-type algorithm [3]. Alternatively, one can employ an EM algorithm or a Gibbs sampler. The Gibbs sampler is conceptually simplest, but is perhaps slow in comparison with the Newton-type method. In particular, we can derive an estimating equation from (4) by taking derivative with respect to $\theta(M)$ as

$$\theta(M) = E_{\boldsymbol{\theta}}[N_M(\Pi)/N(\Pi)].$$

This is also derived in [3] from a physics viewpoint.

4.1 Connection with the motif model

Let us assume that our dictionary consists of only five words, $D = \{A, C, G, T, M_1\}$, where M_1 is a motif sequence of length L. For example, M_1 can be $TGACA$. Then the above dictionary model is to estimate in the dataset the frequency of this word M_1, with the consideration of its chance-occurrence. In other words, even if the dataset is generated as iid from the four basepairs with frequencies p_A, p_T, p_G, and p_C, the chance of observing M_1 is still $p_A^2 p_T p_G p_C$. Due to this ambiguity, the frequency estimation of M_1 is not as straightforward as counting.

Now let us assume that M_1 is a fuzzy word in which each position is not a letter, but a probability distribution on the four letters (e.g., the second position has 85% chance to be T, 10% to be A, 5% chance to be C and 0% chance to be G). Then the computation of segmentation becomes a motif scanning algorithm. That is, it is equivalent to scan the whole dataset to see whether there are matches to the postulated pattern represented by M_1. What differs from the usual pattern scanning is that the "threshold" for considering a candidate segment as the occurrence of M_1 is not given in advance, but determined by the dataset.

Let us further relax the model by assuming that the pattern of the motif (word) M_1 is in fact unknown. Then the model is equivalent to the block-motif model subject to minor tweaking. However, a serious statistical question is whether in such a model the parameters can be estimated consistently. Since the motif pattern is assumed unknown, the only source of information for its inference is its over-abundance in comparison with those "motif-like" patterns occurred by chance under the "null" model.

4.2 Finding the Unknown word via Gibbs sampling

As with the previous subsection, we let $D = \{A, T, G, C, M_1\}$, where M_1 a fuzzy word of length w. Let the usage frequencies of these words be $\boldsymbol{p} = \{p_\alpha;\ \alpha \in D\}$ and let the stochastic word matrix for M_1 be $\Theta = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_w]$. If the missing partition Π were known, according to (4), we have

$$\mathbf{N} = (N_a, \ldots, N_t, N_1 \mid \Pi) \sim \text{Multinom}(N, \boldsymbol{p})$$

where $N = \sum_{\alpha \in D} N_\alpha$. For M_1, we would have a product multinomial model, consisting of independent multinomial models over each column counts $\mathbf{r_j}= (r_{aj}, \ldots r_{tj})^T$, $(j = 1, \ldots w)$ of the stochastic word matrix. We take advantage of this missing data formulation to set up a Gibbs sampling algorithm under a Bayesian framework. We use a conjugate product Dirichlet prior PD($\boldsymbol{B}$) for Θ, where $\boldsymbol{B} = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots \boldsymbol{\beta}_w)$ is a $4 \times w$ matrix with $\boldsymbol{\beta}_j = (\beta_{aj}, \ldots \beta_{tj})^T$, and put a corresponding conjugate Dirichlet prior on $\boldsymbol{p}$, $\boldsymbol{p} \sim \text{Dir}(\boldsymbol{\gamma})$, $\boldsymbol{\gamma} = (\gamma_a, \ldots, \gamma_t)$. Under these assumptions the posterior distribution of Θ would be product Dirichlet PD($\boldsymbol{B}+\ \boldsymbol{r}$), *i.e.* the pseudo-counts $\mathbf{B}$ updated by the counts $\boldsymbol{r} = (\boldsymbol{r}_1, \ldots, \boldsymbol{r}_w)$, and the posterior distribution of $\mathbf{p}$ is Dir($\mathbf{N}$ + γ).We again use the generic notation $\boldsymbol{\theta}$ to denote the collection of parameters ($\mathbf{p}$, Θ), and $\mathbf{R}$ to denote ($\mathbf{N}$, $\mathbf{r}$).

Under this framework, with $\Pi = (P_1, \ldots, P_k)$ representing the *missing* data relating to the correct partitioning of the sequences, we can write the joint posterior distribution as

$$P(\boldsymbol{\theta}, \Pi \mid \mathbf{R}) = P(\Pi \mid \boldsymbol{\theta}, \mathbf{R}) P(\boldsymbol{\theta} \mid \mathbf{R})$$

Now the Bayes estimate of $\boldsymbol{\theta}$ can be approximated by the Gibbs sampler using the full conditional distributions $P(\Pi \mid \boldsymbol{\theta}, \mathbf{R})$ and $P(\boldsymbol{\theta} \mid \mathbf{R}, \Pi)$. The initial step samples for "words" or segmentations given the current value of the stochastic word matrix, and word probabilities. Then it updates the "word" stochastic matrix Θ given the current partition, by sampling for $\boldsymbol{\theta}$ from its posterior distribution.

Sampling for partitions given a value of the stochastic word matrix can be done efficiently using techniques of dynamic programming. This initially involves recursive summation of probabilities as given in (5) over all legitimate partitions of all sequences. This is followed by "backward sampling" for words, starting from the end of the sequence and progressing backwards. Let $\mathcal{A}_i$ denote the set of words (sampled partitions) from position i onward to position n, where n is the length of the current sequence. At position i, we sample for a word α_j of size j, $(j = 1, \ldots w)$ according to the conditional probability (given words occurring positions $i+1$ and beyond),

$$P_i(\alpha_j \mid \mathcal{A}_{i+1}) = \frac{\theta(\boldsymbol{S}_{[(i-j):i]}) L_{i-j}}{\sum_{j=1}^{w} \theta(\boldsymbol{S}_{[(i-j):i]}) L_{i-j}} = \frac{\theta(\boldsymbol{S}_{[(i-j):i]}) L_{i-j}}{L_i(\boldsymbol{\theta})}, \qquad j = 1, \ldots w$$

After the step of sampling partitions it is easy to sample for the stochastic word matrix from its posterior Dirichlet distribution (with prior parameters updated by nucleotide counts at each position of sampled occurrences of M_1).

It is not difficult to generalize this idea for the case of more than one "word", with differing lengths. It would mean expanding the dictionary to include $\{M_2, \ldots, M_d\}$ with associated stochastic word matrices $\{\Theta_i : 2 \leq i \leq d\}$. The statistical question here is whether the parameters remain distinguishable and whether there are sufficient conditions that guarantee that two words do not *overlap*, and remain unconfounded.

Another phenomenon from the biological viewpoint that is a hurdle to many motif-finding techniques is the presence of numerous short repeats of nucleotides (of lengths 2 or 3 etc.) and also long runs of a single nucleotide. These are often biologically insignificant but tend to trap motif-searching algorithms looking for pattern repeats. An attempt to get around this problem is by using a Markovian background structure instead of a random background. In the dictionary model, a comparable background structure may be included by including over-represented dimers or trimers as a part of the alphabet, to account for first and second order dependence. An interesting point to note here is that we are using short polymers to "discount" from the motif signal, whereas in the original dictionary model [3], the dictionary is built up by successive concatenating over-represented short oligomers, hoping to lead to the correct motifs this way.

It is useful to note here the that the segmentation model here is equivalent to the block motif model used in the Gibbs Motif Sampler (Section 3). The motif sampler progressively looks at the sequences and decides whether the segment focused on is more likely to come from the motif model or the background. In this sense its inference is based on the conditional probability $P(I_i \mid I_{[-i]}, \mathbf{M})$ where I_i is the indicator variable for the start of a motif pattern at i, and $I_{[-i]}$ refers to the other motif sites sampled under the motif model $\mathbf{M}$. The segmentation model focuses on the joint likelihood of all motif sites in order to update its knowledge of motif site and composition, *i.e.* $P(\mathbf{I} \mid \mathbf{M})$. It would be an interesting statistical question to analyze whether the segmentation-based Gibbs sampler gives a more accurate results in the presence of faint motif signals, or vice-versa.

4.3 Fine tuning through Metropolis adjustment

The above can be viewed as a multivariate alignment problem which attempts to line up similar fragments in multiple sequences that are realizations from a common word matrix Θ. One of the characteristic features of such an alignment is the possibility of the algorithm getting trapped in a local mode . For example, let $\mathbf{z} = (z_1, \ldots, z_K)$ be the set of start points of a motif of length w in the K sequences. Then $\mathbf{z} + \delta$, for a small integral δ, are local modes of the distribution, differing from the true mode by a common shift, since $w - \delta$ positions are still correctly aligned. In order to encourage global shifting, we insert a Metropolis step, after a certain degree of stability has been reached in the Gibbs algorithm. A global shift essentially means checking columns on either side of the alignment for a higher degree of nucleotide conservation, which would be reflected in the increased sequence likelihood after the shift. Here we make use of a *collapsing* technique [7] to marginalise out the nuisance parameter Θ and get the unconditional likelihood for the current set of motif start positions $P(\mathbf{z} \mid \mathbf{p}, \mathbf{R}) = \int P(\mathbf{z} \mid \Theta, \mathbf{p}, \mathbf{R}) P(\Theta \mid \mathbf{p}, \mathbf{R}) d\Theta$. The Metropolis adjustment is carried out in the following steps:

- choose $\delta = \pm 1$ with probability 1/2 each.
- update current set of motif start positions $\mathbf{z} \leftarrow \mathbf{z} + \delta$ with probability $\min(1, \eta)$, where $\eta = \frac{P(\mathbf{z}+\delta)}{P(\mathbf{z})}$

If $n_{i,l}$ denotes the number of occurrences of nucleotide i in position l of the motif ($l = 1, \ldots, w$, $l = 0$ and $l = w + 1$ denoting the position beyond each end), the ratio of probabilities, η, can be calculated by integrating out the nuisance parameter Θ, and expressed conveniently as a ratio of products of gamma functions,

$$\eta = \begin{cases} \frac{\prod_{i=1}^{k_0} \Gamma(n_{i,w+1})}{\prod_{i=1}^{k_0} \Gamma(n_{i,1})} & \text{if } \delta = 1 \\ \frac{\prod_{i=1}^{k_0} \Gamma(n_{i,0})}{\prod_{i=1}^{k_0} \Gamma(n_{i,w})} & \text{if } \delta = -1 \end{cases}$$

4.4 Influence of the prior information

In a Bayesian analysis, another point of concern is what would be an ideal choice of prior distribution that would not unduly influence our results yet not fail to use potentially important knowledge we have regarding the data. The prior information in the segmentation model is incorporated in the form of pseudo-counts for the Dirichlet prior over the word probabilities in the starting dictionary. To see the effect of prior parameters on our results, our algorithm was applied on a set of DNA fragments that was previously experimentally verified to contain cyclic AMP receptor protein (CRP) binding sites and also analyzed using the EM algorithm [5], and Gibbs Motif Sampler. The degree of correct detection of binding sites, (total number of true sites is 24) under differing degrees of prior strength for presence of motif, can be seen in the table below (TP and FP denote true and false positives occurring more than 20% of the time, PC denotes pseudo-counts of motif as a fraction of base pseudo-counts):

Motif PC(%)	0.25	1.25	2.5	5	10
Average TP : mean (s.e.)	17 (0.99)	17.3 (1.21)	17.8 (0.35)	18 (0.45)	18.8 (0.37)
Average FP	1.3	2	4.3	7.2	11.2
(%) of correct sites	70.83	71.88	73.96	75.00	78.33

It appears that there is a trade-off between the percentage of correct sites sampled and minimizing the degree of false detection. It is interesting to note that taking the motif pseudo-count around 1.27% of base pseudo-counts lead to an expected frequency of 24 for the motif in this data set, which is the true frequency.

4.5 Finding gapped motifs through segmentation

Another interesting twist to this problem is the question of how to track motifs that may have one or more insertions of nucleotides, *gaps*, within them. The segmentation model can be extended to allow for this new possibility. A mathematical question of interest is, how far these gaps can extend, beyond which the original pattern becomes indiscernible.

The problem of searching for gapped motifs can be thought of as trying to align a $k_0 \times w$ word stochastic matrix with a segment of length $(w+g)$ where g is the total length of the gap(s) occurring in a motif. Our probability model for segmentations now is:

$$P(\Pi, \mathcal{G} \mid \boldsymbol{\theta}, \mathbf{R}) = P(\Pi \mid \boldsymbol{\theta}, \mathbf{R}) P(\mathcal{G} \mid \Pi, \boldsymbol{\theta}, \mathbf{R})$$

where $\mathcal{G}$ denotes the collective set of gap positions within all motifs.

Here we need to additionally sample from the full conditional distribution of gaps $\mathcal{G}$, given the current set of sampled partitions Π and Θ. At this stage, we introduce additional probability parameters, p_m, probability of a match between a nucleotide of the segment with the stochastic word matrix, p_{go}, gap-opening penalty, p_{ge}, gap-extension penalty (typically lower than gap-opening penalty, as a motif is more likely to have few but longer gaps within it than a series of numerous small gaps). At this stage we assume there are no deletions in the motif, hence no gaps in the stochastic word matrix, an assumption which may be later relaxed.

Now we need to deal with the alignment problem, basically sampling for gaps within a segment $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{w+g})^T$ of length $(w+g)$. This may also be done recursively. Let us assume the motif is aligned exactly with the word stochastic matrix at the end, i.e. the last position is not part of a gap. Let M_j denote the number of matches before position j. Let $P_{\theta_0}(x)$ denote probability of nucleotide x if it lies in a gap (i.e., under the background model), and $P_\Theta(x, i)$ denote the probability of x being realized from the i-th column of the motif model. Denote the probabilities of the j-th position being the i-th match, or belonging to a gap after the i-th match, as, respectively,

$$F_{i,j}(M) = P(x_j \in \mathcal{G}^c \mid M_j = i-1), \qquad \text{and} \qquad F_{i,j}(H) = P(x_j \in \mathcal{G} \mid M_j = i)$$

Then we can calculate the above quantities recursively, (with initial conditions $F_{0,0}(H) = 1, F_{1,1}(M) = p_m P_\Theta(x_1, 1)$),

$$\begin{aligned}
F_{0,k}(H) &= F_{0,k-1}(H) P_{\theta_0}(x_k), \quad 2 \le k \le g \\
F_{j,k}(H) &= [F_{j,k-1}(H) p_{ge} + F_{j,k-1}(M) p_{go}] P_{\theta_0}(x_k) \quad j+1 \le k \le g+j, \quad 1 \le j \le w-1 \\
F_{j,k}(M) &= [F_{j,k-1}(H) + F_{j,k-1}(M)] p_m P_\Theta(x_k, j) \quad j \le k \le g+j, \quad 1 \le j \le w-1
\end{aligned}$$

Then, for the segment $\mathbf{x}$,

$$P(\mathbf{x}) = F_{w,w+g}(M) = [F_{w-1,w+g-1}(H) + F_{w-1,w+g-1}(M)] p_m P_\Theta(x_{w+g}, w)$$

Sampling for gaps can now be accomplished by sampling a two-dimensional pathway from the aligned ends of the segment $\mathbf{x}$ and word matrix Θ, up to the starting point of the segment.

In summary, it appears that using the segmentation model provides an elegant Bayesian tool for motif-finding that can be modified without excessive complications to suit different biological contingencies. It makes use of minimal assumptions about the composition and profusion of the motif and updates its information based only on the information contained in the sequence data. A potential drawback is the possibility that it might fail to yield significant results in situations where the motif signal is comparatively faint, as the only source of information it uses in inference is over-abundance of patterns in comparison to chance occurrences under the background (null) model, but this requires further study before conclusions can be made.

5 Microarray Expression and DNA Motif Discovery

Although every cell of a multicellular organism contains the identical genetic materials (the whole genome), cells from different parts of the organism differ greatly. Cells make this differentiation possible by adjusting the amount of its gene product (i.e., proteins) via various kinds of regulations, among which the transcriptional regulation is the most popular one. Similarly, the single-cell organism responds to different environmental changes (e.g., starvation, heat-shock, etc.) also by regulating its gene product, mostly through transcriptional regulation.

The advances of the DNA microarray technology has made it possible for the scientist to observe the transcriptional changes of all the genes in a cell simultaneously. On the other hand, the availability of the complete genome of many species (e.g., E. coli, yeast, C. Elegans, etc.) allows one to extract the regulatory regions (upstream noncoding regions) of the target genes. When one combines these two sources of information (i.e., microarray and sequence data) and applies the motif search algorithm such as the Gibbs motif sampler, one can discover novel regulatory binding sites *in silico* [11]. Briefly, one first identifies a set of genes that are co-regulated (under certain environment) via microarray observations. Then one searches, using the Gibbs motif sampler or a similar algorithm, for repetitive patterns (7 to 20 bps) among the upstream untranslated regions of these genes. The patterns found in this way often correspond to the binding sites of certain transcription factors.

Clearly, the motif analysis and the microarray analysis should be applied jointly since they tend to enhance each other in a biologically meaningful way: The gene clusters inferred from the microarray analysis often reveal genes involved in related biological pathway or genes that are regulated by the same TF. If the motif analysis can indeed reveal some significant motifs for these genes, it not only confirms the clustering result, but also suggests future experimental directions. For example, it is observed in [6] that the logarithm of the "motif-score" for each segment in the dataset (i.e., how likely the segment contains certain motif) is strongly correlated with the median percentile rank of the expression levels in a microarray experiment, confirming that the motif found is perhaps authentic.

More recently, a new protocol called the chromatin-immunoprecipitation followed by microarray hybridization (ChIP-array) has been developed for discovering protein-DNA interaction loci *in vivo*. In these experiments, DNA is crosslinked *in vivo* to proteins at sites of DNA-protein interaction, and sheared to 1-2kb fragments. The DNA-protein complexes are precipitated by antibodies specific to the protein of interest. The precipitated protein-bound DNA fragments are amplified, labeled fluorescently, and hybridized to microarrays containing every ORF and intergenic region of the yeast genome. DNA fragments that are consistently enriched by ChIP-array over repeated experiments are identified as containing the protein-DNA interacting loci.

The DNA fragments selected by the ChIP-array experiments have an average size of 1-2kb, which is determined by the shearing process. It is therefore still rather difficult to pinpoint the exact binding sites by these experiments alone. Additional tedious experimental approaches can be used to further localize the binding sites (restriction cleavage, footprinting, etc.). Fortunately, it is also possible to conduct computational analysis as suggested in the previous sections at this stage to further localize the protein-DNA interaction site and discover the precise binding motif. A new method for dealing with this type of data is under development [9].

References

[1] Ren B, Robert F, Wyrick JJ, Aparicio O, Jennings EG, Simon I, Zeitlinger J, Schreiber J, Hannett N, Kanin E, Volkert TL, Wilson CJ, Bell SP, and Young RA. Genome-wide location and function of dna binding proteins. *Science*, 290:2306–2309, 2000.

[2] T. L. Bailey and C. P. Elkan. Fitting a mixture model by expectation-maximization to discover motifs in biopolymers. *ISMB*, pages 28–36, 1994.

[3] H. J. Bussemaker, H. Li, and E. D. Siggia. Building a dictionary for genomes: Identification of presumptive regulatory sites by statistical analysis. *Proc. Nat'l Acad. Sci. USA*, 97(18):10096–10100, 2000.

[4] C. E. Lawrence, S. F. Altschul, M. S. Boguski, J. S. Liu, A. F. Neuwald, and J. C. Wootton. Detecting subtle sequence signals: a gibbs sampling strategy for multiple alignment. *Science*, 262(5131):208–14, 1993.

[5] C. E. Lawrence and A. A. Reilly. An expectation maximization (em) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences. *Proteins*, 7:41–51, 1990.

[6] J. D. Lieb, X. Liu, D. Botstein, and P. O. Brown. Promoter-specific binding of rap1 revealed by genome-wise maps of protein-dna association. *Nature Genetics*, 28:327–334, 2001.

[7] J. S. Liu. The collapsed gibbs sampler in bayesian computations with applications to a gene-regulation problem. *Journal of the American Statistical Association*, 89(427):958–966, 1994.

[8] J. S. Liu, A. F. Neuwald, and C. E. Lawrence. Bayesian models for multiple local sequence alignment and gibbs sampling strategies. *Journal of the American Statistical Association*, 90(432):1156–1170, 1995.

[9] X. Liu, D. L. Brutlag, and J. S. Liu. A fast computational method for finding protein-dna interaction sites from chromatin immunoprecipitation microarray experiments. Technical report, Department of Statistics, Harvard University, 2001.

[10] A. F. Neuwald, J. S. Liu, and C. E. Lawrence. Gibbs motif sampling : Detection of bacterial outer-membrane protein repeats. *Protein Science*, 4(8):1618–1632, 1995.

[11] F. P. Roth, J. D. Hughes, P. W. Estep, and G. M. Church. Finding dna regulatory motifs within unaligned noncoding sequences clustered by whole-genome mrna quantitation. *Nature Biotechnology*, 16(10):939–945, 1998.

[12] G. D. Stormo and G. W. Hartzell III. Identifying protein-binding sites from unaligned dna fragments. *Proceedings of the Nathional Academy of Science, USA*, 86:1183–1187, 1989.

[13] M. Tanner and W. H. Wong. The calculation of posterior distributions by data augmentation. *Journal of the American Statistical Association*, 82:528–550, 1987.