Chapter 8

Regulatory Motif Discovery: from Decoding to Meta-Analysis

Qing Zhou[*] Mayetri Gupta[†]

Abstract

Gene transcription is regulated by interactions between transcription factors and their target binding sites in the genome. A motif is the sequence pattern recognized by a transcription factor to mediate such interactions. With the availability of high-throughput genomic data, computational identification of transcription factor binding motifs has become a major research problem in computational biology and bioinformatics. In this chapter, we present a series of Bayesian approaches to motif discovery. We start from a basic statistical framework for motif finding, extend it to the identification of *cis*-regulatory modules, and then discuss methods that combine motif finding with phylogenetic footprinting, gene expression or ChIP-chip data, and nucleosome positioning information. Simulation studies and applications to biological data sets are presented to illustrate the utility of these methods.

Keywords: Transcriptional regulation; motif discover; cis-regulatory; Gene expression; DNA sequence; ChIP-chip; Bayesian model; Markov Chain Monte Carlo.

1 Introduction

The goal of motif discovery is to locate short repetitive patterns ("words") in DNA that are involved in the regulation of genes of interest. In *transcriptional* regulation, sequence signals upstream of each gene provide a target (the *promoter region*) for an enzyme complex called RNA polymerase (RNAP) to bind and initiate the transcription of the gene into *messenger* RNA (mRNA). Certain proteins called *transcription factors* (TFs) can bind to the promoter regions, either interfering with the action of RNAP and inhibiting gene expression, or enhancing gene expression. TFs recognize sequence sites that give a favorable binding energy, which often translates into a sequence-specific pattern (∼8-20 base pairs long). Binding sites thus tend to be relatively well-conserved in composition – such a conserved

[*] Department of Statistics, University of California at Los Angeles, Los Angeles, CA 90095, USA, E-mail: zhou@stat.ucla.edu

[†] Department of Biostatistics, University of North Carolina at Chapel Hill, Chapel Hill, NC 27516-7420, USA, E-mail:mgupta@unc.edu

pattern is termed as a "motif". Experimental detection of TF-binding sites (TFBSs) on a gene-by-gene and site-by-site basis is possible but remains an extremely difficult and expensive task at a genomic level, hence computational methods that assume no prior knowledge of the motif become necessary.

With the availability of complete genome sequences, biologists can now use techniques such as DNA gene expression microarrays to measure the expression level of each gene in an organism under various conditions. A collection of expressions of each gene measured under various conditions is called the gene expression profile. Genes can be divided into clusters according to similarities in their expression profiles–genes in the same cluster respond similarly to environmental and developmental changes and thus may be co-regulated by the same TF or the same group of TFs. Therefore, computational analysis is focused on the search for TFBSs in the upstream of genes in a particular cluster. Another powerful experimental procedure called Chromatin ImmunoPrecipitation followed by microarray (ChIP-chip) can measure where a particular TF binds to DNA in the whole genome under a given experimental condition at a coarse resolution of 500 to 2000 bases. Again, computational analysis is required to pinpoint the short binding sites of the transcription factor from all potential TF binding regions.

With these high throughput gene expression and ChIP-chip binding data, *de novo* methods for motif finding have become a major research topic in computational biology. The main constituents a statistical motif discovery procedure requires are: (i) a probabilistic structure for generating the observed text (i.e. in what context a word is "significantly enriched") and (ii) an efficient computational strategy to find all enriched words. In the genomic context, the problem is more difficult because the "words" used by the nature are never "exact", i.e., certain "mis-spellings" can be tolerated. Thus, one also needs a probabilistic model to describe a fuzzy word.

An early motif-finding approach was CONSENSUS, an information theory-based progressive alignment procedure [42]. Other methods included an EM-algorithm [11] based on a missing-data formulation [24], and a Gibbs sampling algorithm [23]. Later generalizations that allowed for a variable number of motif sites per sequence were a Gibbs sampler [28, 33] and an EM algorithm for finite mixture models [2].

Another class of methods approach the motif discovery problem from a "segmentation" perspective. MobyDick [6] treats the motifs as "words" used by nature to construct the "sentences" of DNA and estimates word frequencies using a Newton-Raphson optimization procedure. The dictionary model was later extended to include "stochastic" words in order to account for variations in the motif sites [16, 36] and a data augmentation (DA) [43] procedure introduced for finding such words.

Recent approaches to motif discovery have improved upon the previous methods in at least two primary ways: (i) improving and sensitizing the basic model to reflect realistic biological phenomena, such as multiple motif types in the same sequence, "gapped" motifs, and clustering of motif sites (cis-regulatory modules) [30, 51, 17], and (ii) using auxiliary data sources, such as gene expression microarrays, ChIP-chip data, phylogenetic information and the physical structure of DNA

[9, 21, 52, 18]. In the following section we will discuss the general framework of *de-novo* methods for discovering uncharacterized motifs in biological sequences, focusing especially on the Bayesian approach.

2 A Bayesian approach to motif discovery

In this section, unless otherwise specified, we assume that the data set is a set of N unaligned DNA fragments. Let $\boldsymbol{S} = (S_1, \cdots, S_N)$ denote the N sequences of the data set, where sequence S_i is of length L_i $(i = 1, \cdots, N)$. Multiple instances of the same pattern in the data are referred to as motif *sites* or *elements* while different patterns are termed motifs. Motif type k (of, say, width w_k) is characterized by a Position-Specific Weight matrix (PWM) $\Theta_k = (\boldsymbol{\theta}_{k1}, \cdots, \boldsymbol{\theta}_{kw_k})$, where the J-dimensional ($J = 4$ for DNA) vector $\boldsymbol{\theta}_{ki} = (\theta_{ki1}, \cdots, \theta_{kiJ})^T$ represents the probabilities of occurrence of the J letters in column i, $(i = 1, \cdots, w_k)$. The corresponding letter occurrence probabilities in the *background* are denoted by $\boldsymbol{\theta}_0 = (\theta_{01}, \cdots, \theta_{0J})$. Let $\boldsymbol{\Theta} = \{\Theta_1, \cdots, \Theta_K\}$.

We assume for now that the motif widths, w_k $(k = 1, \cdots, K)$ are known (this assumption will be relaxed later). The locations of the motif sites are unknown, and are denoted by an array of missing indicator variables $\boldsymbol{A} = (A_{ijk})$, where $A_{ijk} = 1$ if position j $(j = 1, \cdots, L_i)$ in sequence i $(i = 1, \cdots, N)$ is the starting point of a motif of type k $(k = 1, \cdots, K)$. For motif type k, we let $\boldsymbol{A}_k = \{A_{ijk} : i = 1, \cdots, N;\ j = 1, \cdots, L_i\}$, i.e., the indicator matrix for the site locations corresponding to this motif type, and define the alignment:

$$\begin{aligned}
S_1^{(\boldsymbol{A}_k)} &= \{S_{ij} : A_{ijk} = 1; i = 1, \cdots, N;\ j = 1, \cdots, L_i\},\\
S_2^{(\boldsymbol{A}_k)} &= \{S_{i,j+1} : A_{ijk} = 1; i = 1, \cdots, N;\ j = 1, \cdots, L_i\},\\
&\cdots\\
S_{w_k}^{(\boldsymbol{A}_k)} &= \{S_{i,j+w_k-1} : A_{ijk} = 1; i = 1, \cdots, N;\ j = 1, \cdots, L_i\}.
\end{aligned}$$

In words, $S_i^{(\boldsymbol{A}_k)}$ is the set of letters occurring at position i of all the instances of the type-k motif.

In a similar fashion, we use $S^{(A^c)}$ to denote the set of all letters occurring in the background, where $S^{(A^c)} = \boldsymbol{S} \setminus \bigcup_{k=1}^{K} \bigcup_{l=1}^{w_k} S_l^{(A_k)}$ (For two sets $A, B,\ A \subset B$, $B \setminus A \equiv B \cap A^c$). Further, let $\mathcal{C} : \boldsymbol{S} \to \mathbb{Z}^4$ denote a "counting" function that gives the frequencies of the J letters in a specified subset of $\boldsymbol{S}$. For example, if after taking the set of all instances of motif k, in the first column, we observe a total occurrence of 10 'A's, 50 'T's and no 'C' or 'G's, $\mathcal{C}(S_1^{(\boldsymbol{A}_k)}) = (10, 0, 0, 50)$. Assuming that the motif columns are independent, we have

$$[\mathcal{C}(S_1^{(A_k)}), \cdots, \mathcal{C}(S_{w_k}^{(A_k)})] \sim \text{Product-Multinomial}[\Theta_k = (\boldsymbol{\theta}_{k1}, \cdots, \boldsymbol{\theta}_{kw_k})],$$

i.e., the i th vector of column frequencies for motif k follows a multinomial distribution parametrized by θ_{ki}.

We next introduce some general mathematical notation. For vectors $\mathbf{v} = (v_1, \cdots, v_p)^T$, let us define $|\mathbf{v}| = |v_1| + \cdots + |v_p|$, and $\Gamma(\mathbf{v}) = \Gamma(v_1) \cdots \Gamma(v_p)$.

Then the normalizing constant for a p-dimensional Dirichlet distribution with parameters $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_p)^T$ can be denoted as $\Gamma(|\boldsymbol{\alpha}|)/\Gamma(\boldsymbol{\alpha})$. For notational convenience, we will denote the inverse of the Dirichlet normalizing constant as $ID(\boldsymbol{\alpha}) = \Gamma(\boldsymbol{\alpha})/\Gamma(|\boldsymbol{\alpha}|)$. Finally, for vectors $\mathbf{v}$ and $\mathbf{u} = (u_1, \cdots, u_p)$, we use the shorthand $\boldsymbol{u}^{\boldsymbol{v}} = \prod_{i=1}^{p} u_i^{v_i}$.

The probability of observing $\boldsymbol{S}$ conditional on the indicator matrix $\boldsymbol{A}$ can then be written as

$$P(\boldsymbol{S} \mid \boldsymbol{\Theta}, \boldsymbol{\theta}_0, \boldsymbol{A}) \propto \boldsymbol{\theta}_0^{\mathcal{C}(S^{(A^c)})} \prod_{k=1}^{K} \prod_{i=1}^{w_k} \boldsymbol{\theta}_{ki}^{\mathcal{C}(S_i^{(A_k)})}.$$

For a Bayesian analysis, we assume a conjugate Dirichlet prior distribution for $\boldsymbol{\theta}_0$, $\boldsymbol{\theta}_0 \sim \text{Dirichlet}(\boldsymbol{\beta}_0)$, $\boldsymbol{\beta}_0 = (\beta_{01}, \cdots, \beta_{0D})$, and a corresponding product-Dirichlet prior (i.e., independent priors over the columns) PD($\boldsymbol{B}$) for Θ_k $(k = 1, \cdots, K)$, where $\boldsymbol{B}{=}(\boldsymbol{\beta}_{k1}, \boldsymbol{\beta}_{k2}, \cdots, \boldsymbol{\beta}_{kw_k})$ is a $J{\times}w_k$ matrix with $\boldsymbol{\beta}_{ki}{=}(\beta_{ki1}, \cdots, \beta_{kiJ})^T$. Then the conditional posterior distribution of the parameters given $\boldsymbol{A}$ is:

$$P(\boldsymbol{\Theta}, \boldsymbol{\theta} \mid \boldsymbol{S}, \boldsymbol{A}) \propto \boldsymbol{\theta}_0^{\mathcal{C}(S^{(A^c)})+\boldsymbol{\beta}_0} \prod_{k=1}^{K} \prod_{i=1}^{w_k} \boldsymbol{\theta}_{ki}^{\mathcal{C}(S_i^{(A_k)})+\boldsymbol{\beta}_{ki}}.$$

For the complete joint posterior of all unknowns $(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{A})$, we further need to prescribe a prior distribution for $\boldsymbol{A}$. In the original model [23], a single motif site per sequence with equal probability to occur anywhere was assumed. However, in a later model [28] that can allow multiple sites, a Bernoulli(π) model is proposed for motif site occurrence. More precisely, assuming that a motif site of width w can occur at any of the sequence positions, $1, 2, \cdots, L^* - w + 1$ in a sequence of length L^*, with probability π, the joint posterior distribution is:

$$P(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{A} \mid \boldsymbol{S}) \propto \boldsymbol{\theta}_0^{\mathcal{C}(S^{(A^c)})+\boldsymbol{\beta}_0} \prod_{k=1}^{K} \prod_{i=1}^{w_k} \boldsymbol{\theta}_{ki}^{\mathcal{C}(S_i^{(A_k)})+\boldsymbol{\beta}_{ki}} \pi^{|\boldsymbol{A}|} (1-\pi)^{L-|\boldsymbol{A}|}, \tag{2.1}$$

where $L = \sum_{i=1}^{N} (L_i - w)$ is the adjusted total length of all sequences and $|\boldsymbol{A}| = \sum_{k=1}^{K} \sum_{i=1}^{N} \sum_{j=1}^{L_i} A_{ijk}$. If we have reason to believe that motif occurrences are not independent, but occur as clusters (as in regulatory modules), we can instead adopt a prior Markovian model for motif occurrence [17, 44] which is discussed further in Section 3.

2.1 Markov chain Monte Carlo computation

Under the model described in (2.1), it is straightforward to implement a Gibbs sampling (GS) scheme to iteratively update the parameters, i.e., sampling from $[\Theta, \theta_0 \mid \mathcal{C}, \boldsymbol{A}]$, and impute the missing data, i.e., sampling from $[\boldsymbol{A} \mid \mathcal{C}, \Theta, \theta_0]$. However, drawing Θ from its posterior at every iteration can be computationally inefficient. Liu et al. [28] demonstrated that *marginalizing* out (Θ, θ_0) from the posterior distribution can lead to much faster convergence of the algorithm [29]. In other words, one can use the Gibbs sampler to draw from the marginal distribution

$$p(\boldsymbol{A} \mid \boldsymbol{S},\pi)=\int\!\!\int p(\boldsymbol{\Theta},\boldsymbol{\theta}_0 \mid \boldsymbol{S},\boldsymbol{A},\pi)p(\boldsymbol{A})p(\boldsymbol{\Theta},\boldsymbol{\theta}_0)d\boldsymbol{\Theta}d\boldsymbol{\theta}_0, \tag{2.2}$$

which can be easily evaluated analytically.

If π is unknown, one can assume a beta prior distribution Beta (α_1, α_2) and marginalize out π from the posterior, in which case $p(\boldsymbol{A} \mid \boldsymbol{S})$ can be derived from (2.2) by altering the last term in (2.2) to the ratio of normalizing constants for the Beta distribution, $B(|A|+\alpha_1, L-|A|+\alpha_2)/B(\alpha_1,\alpha_2)$. Based on (2.2), Liu et al. [28] derived a *predictive updating* algorithm for $\boldsymbol{A}$, which is to iteratively sample each component of $\boldsymbol{A}$ according to the predictive distribution

$$\frac{P(A_{ijk}=1 \mid \boldsymbol{S})}{P(A_{ijk}=0 \mid \boldsymbol{S})} = \frac{\pi}{1-\pi}\prod_{l=1}^{w_k}\left(\frac{\hat{\boldsymbol{\theta}}_{kl}}{\hat{\boldsymbol{\theta}}_0}\right)^{\mathcal{C}(S_{i,j+l,k})}, \tag{2.3}$$

where the posterior means are $\hat{\boldsymbol{\theta}}_{kl}=\frac{\mathcal{C}(S_l^{(A_k)})+\boldsymbol{\beta}_{kl}}{|\mathcal{C}(S_l^{(A_k)})+\boldsymbol{\beta}_{kl}|}$ and $\hat{\boldsymbol{\theta}}_0=\frac{\mathcal{C}(S^{(A^c)})+\boldsymbol{\beta}_0}{|\mathcal{C}(S^{(A^c)})+\boldsymbol{\beta}_0|}$.

Under the model specified above, it is also possible to implement a "partition-based" data augmentation (DA) approach [16] that is motivated by the recursive algorithm used in Auger and Lawrence [1]. The DA approach samples $\boldsymbol{A}$ jointly according to the conditional distribution

$$P(\boldsymbol{A} \mid \boldsymbol{\Theta},\boldsymbol{S}) = \prod_{i=1}^{N} P(\boldsymbol{A}_{iL_i} \mid \boldsymbol{\Theta},\boldsymbol{S}) \prod_{j=1}^{L_i-1} P(\boldsymbol{A}_{ij}|\boldsymbol{A}_{i,j+1},\cdots,\boldsymbol{A}_{iL_i},\boldsymbol{S},\boldsymbol{\Theta}).$$

At a position j, the current knowledge of motif positions is updated using the conditional probability $P(\boldsymbol{A}_{ij} \mid \boldsymbol{A}_{i,j+1},\cdots,\boldsymbol{A}_{iL_i},\boldsymbol{\Theta})$ (backward sampling), with $\boldsymbol{A}_{i,j-1},\cdots,\boldsymbol{A}_{i1}$ marginalized out using a forward summation procedure (an example will be given in Section 3.1). In contrast, at each iteration, GS iteratively draws from the conditional distribution: $P(A_{ijk}|\boldsymbol{A} \setminus A_{ijk},\boldsymbol{S})$, iteratively visiting each sequence position i, updating its motif indicator conditional on the indicators for other positions. The Gibbs approach tends to be "sticky" when the motif sites are abundant. For example, once we have set $A_{ijk}=1$ (for some k), we will not be able to allow segment $S_{[i,j+1:j+w_k]}$ to be a motif site. The DA method corresponds to a *grouping* scheme (with $\boldsymbol{A}$ sampled together), whereas the GMS corresponds to a *collapsing* approach (with $\boldsymbol{\Theta}$ integrated out). Both have been shown to improve upon the original scheme [29].

2.2 Some extensions of the product-multinomial model

The product-multinomial model used for Θ is a first approximation to a realistic model for transcription factor binding sites. In empirical observations, it has been reported that certain specific features often characterize functional binding sites. We mention here a few extensions of the primary motif model that have been recently implemented to improve the performance of motif discovery algorithms.

In the previous discussion, the width w of a motif Θ was assumed to be known and fixed; we may instead view w as an additional unknown model parameter. Jointly sampling from the posterior distribution of $(\boldsymbol{A}, \Theta, w)$ is difficult as the dimensionality of Θ changes with w. One way to update (w, Θ) jointly would be through a reversible jump procedure [15]. However, note that we can integrate out Θ from the posterior distribution to avoid a dimensionality change during the updating. By placing an appropriate prior distribution $p(w)$ on w (a possible choice is a Poisson(λ)), we can update w using a Metropolis step. Using a Beta(α_1, α_2) prior on π, the marginalized posterior distribution is $P(\boldsymbol{A}, w|\boldsymbol{S}) \propto ID(\mathcal{C}(S^{(A^c)}) + \boldsymbol{\beta}_0) \prod_{i=1}^{w} \dfrac{ID(\mathcal{C}(S_i^{(A)}) + \boldsymbol{\beta}_i)}{ID(\boldsymbol{\beta}_i)} \dfrac{B(|A| + \alpha_1, L - |A| + \alpha_2)}{B(\alpha_1, \alpha_2)} p(w)$.

Another assumption in the product multinomial model is that all columns of a weight matrix are independent– however, it has been observed that about 25% of experimentally validated motifs show statistically significant positional correlations. Zhou and Liu [49] extend the independent weight matrix model to including one or more correlated column pairs, under the restriction that no two pairs of correlated columns can share a column in common. A Metropolis-Hastings step is added in the Gibbs sampler [28] that deletes or adds a pair of correlated column at each iteration. Other proposed models are a Bayesian tree-like network modeling the possible correlation structure among all the positions within a motif model [4], and a permuted Markov model in which the assumption is that an unobserved permutation has acted on the positions of all the motif sites and that the original ordered positions can be described by a Markov chain [48]. Mathematically, the model [49] is a sub-case of [48], which is, in turn, a sub-case of [4].

3 Discovery of regulatory modules

Motif predictions for higher eukaryotic genomes are more challenging than that for simpler organisms such as bacteria or yeast, for reasons such as (i) large sections of low-complexity regions (repeat sequences), (ii) weak motif signals, (iii) sparseness of signals compared to entire region under study-binding sites may occur as far as 2000—3000 bases away from the transcription start site, either upstream or downstream. In addition, in complex eukaryotes, regulatory proteins often work in combination to regulate target genes, and their binding sites have often been observed to occur in spatial clusters, or *cis-regulatory modules* (Figure 1). One approach to locating cis-regulatory modules (CRMs) is by predicting novel motifs and looking for co-occurrences [41]. However, since individual motifs in the cluster may not be well-conserved, such an approach often leads to a large number of false negatives. Here, we describe a strategy to first use existing *de novo* motif finding algorithms and motif databases to compose a list of putative binding motifs, $\mathcal{D} = \{\Theta_1, \cdots, \Theta_D\}$, where D is in the range of 50 to 100, and then simultaneously update these motifs and estimate the posterior probability for each of them to be included in the CRM [17].

Let $\boldsymbol{S}$ denote the set of n sequences with lengths $L_1, L_2, \cdots, L_n$, respectively,

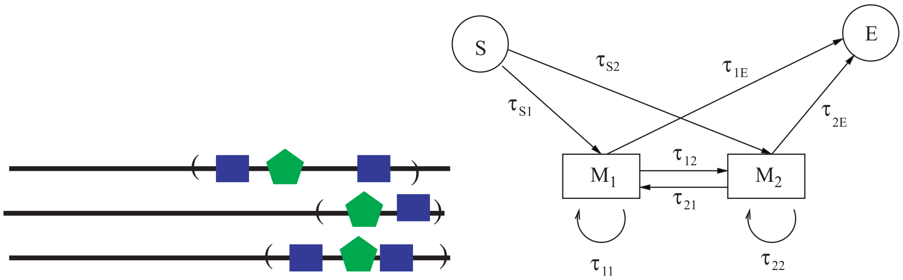

Figure 1: Graphical illustration of a CRM

corresponding to the upstream regions of n co-regulated genes. We assume that the CRM consists of K different kinds of motifs with distinctive PWMs. Both the PWMs and K are unknown and need to be inferred from the data. In addition to the indicator variable $\boldsymbol{A}$ defined in Section 2, we define a new variable $a_{i,j}$, that denotes the location of the jth site (irrespective of motif type) in the ith sequence. Let $\boldsymbol{a} = \{a_{ij}; i = 1, \cdots, n;\ j = 1, \cdots, L_i\}$. Associated with each site is its *type* indicator $T_{i,j}$, with $T_{i,j}$ taking one of the K values (Let $\boldsymbol{T} = (T_{ij})$). Note that the specification $(\boldsymbol{a}, \boldsymbol{T})$ is essentially equivalent to $\boldsymbol{A}$.

Next, we model the dependence between $T_{i,j}$ and $T_{i,j+1}$ by a $K \times K$ probability transition matrix $\boldsymbol{\tau}$. The distance between neighboring TFBSs in a CRM, $d_{ij} = a_{i,j+1} - a_{i,j}$, is assumed to follow $Q(\ ;\lambda, w)$, a geometric distribution truncated at w, i.e. $Q(d;\lambda, w) = (1-\lambda)^{d-w}\lambda \quad (d = w, w+1, \cdots)$. The distribution of nucleotides in the *background* sequence is a multinomial distribution with unknown parameter $\boldsymbol{\rho} = (\rho_A, \cdots, \rho_T)$.

Next, we let $\boldsymbol{u}$ be a binary vector indicating which motifs are included in the module, i.e. $\boldsymbol{u} = (u_1, \cdots, u_D)^T$, where $u_j = 1(0)$ if the j th motif type is present (absent) in the module. By construction, $|\boldsymbol{u}| = K$. Thus, the information regarding K is completely encoded by $\boldsymbol{u}$. In light of this notation, the set of PWMs for the CRM is defined as $\boldsymbol{\Theta} = \{\Theta_j :\ u_j = 1\}$. Since now we restrict our inference of CRM to a subset of $\mathcal{D}$, the probability model for the observed sequence data can be written as:

$$P(\boldsymbol{S}\,|\,\mathcal{D},\boldsymbol{\tau},\boldsymbol{u},\lambda,\boldsymbol{\rho}) = \sum_{\boldsymbol{a}}\sum_{\boldsymbol{T}} P(\boldsymbol{S}\,|\,\boldsymbol{a},\boldsymbol{T},\mathcal{D},\boldsymbol{\tau},\boldsymbol{u},\lambda,\,\boldsymbol{\rho})P(\boldsymbol{a}\,|\,\lambda)P(\boldsymbol{T}\,|\,\boldsymbol{a},\boldsymbol{\tau}).$$

From the above likelihood formulation, we need to simultaneously estimate the optimal $\boldsymbol{u}$ and the parameters $(\mathcal{D}, \boldsymbol{\tau}, \lambda, \boldsymbol{\rho})$. To achieve this, we first prescribe a prior distribution on the parameters and missing data:

$$P(\mathcal{D}, \boldsymbol{\tau}, \boldsymbol{u}, \lambda, \boldsymbol{\rho}) = f_1(\mathcal{D} \mid \boldsymbol{u}) f_2(\boldsymbol{\tau} \mid \boldsymbol{u}) f_3(\boldsymbol{\rho}) g_1(\boldsymbol{u}) g_2(\lambda).$$

Here the $f_i(\cdot)$'s are (product) Dirichlet distributions. Assuming each u_i takes the value 1 with a prior probability of π (i.e. π is the prior probability of including a motif in the module), $g_1(\boldsymbol{u})$ represents a product of D Bernoulli(π) distributions; and $g_2(\lambda)$, a generally flat Beta distribution. More precisely, we assume *a priori* that $\Theta_i \sim \prod_{j=1}^{w} \text{Dirichlet}(\boldsymbol{\beta}_{ij})$ (for $i = 1, \cdots, D$); $\boldsymbol{\rho} \sim \text{Dirichlet}(\boldsymbol{\beta}_0)$; $\lambda \sim \text{Beta}(a, b)$. Given $\boldsymbol{u}$ (with $|\boldsymbol{u}| = K$), each row of $\boldsymbol{\tau}$ is assumed to follow an independent Dirichlet. Let the i th row $v_i|\boldsymbol{u} \sim \text{Dirichlet}(\boldsymbol{\alpha}_i)$, where $i = 1, \cdots, K$.

Let $\Omega = (\mathcal{D},\boldsymbol{\tau},\lambda,\boldsymbol{\rho})$ denote the full parameter set. Then the posterior distribution of Ω has the form

$$P(\Omega, \boldsymbol{u} \,|\, \boldsymbol{S}) \propto P(\boldsymbol{S} \,|\, \boldsymbol{u},\Omega) f_1(\mathcal{D} \,|\, \boldsymbol{u}) f_2(\boldsymbol{\tau} \,|\, \boldsymbol{u}) f_3(\boldsymbol{\rho}) g_1(\boldsymbol{u}) g_2(\lambda). \tag{3.1}$$

Gibbs sampling approaches were developed to infer the CRM from a special case of the posterior distribution (3.1) with fixed $\boldsymbol{u}$ [44, 51]. Given the flexibility of the model and the size of the parameter space for an unknown $\boldsymbol{u}$, it is unlikely that a standard MCMC approach can converge to a good solution in a reasonable amount of time. If we ignore the ordering of sites $\boldsymbol{T}$ and assume components of $\boldsymbol{a}$ to be independent, this model is reduced to the original motif model in Section 2 which can be updated through the previous Gibbs or DA procedure.

3.1 A hybrid EMC-DA approach: EMCmodule

With a starting set of putative binding motifs $\mathcal{D}$, an alternative approach was proposed by Gupta and Liu [17], which involves simultaneously modifying the motifs and estimating the posterior probability for each of them to be included in the CRM. This was acheived through iterations of the following Monte Carlo sampling steps: (i) Given the current collection of motif PWMs (or sites), sample motifs into the CRM by evolutionary Monte Carlo (EMC); (ii) Given the CRM configuration and the PWMs, update the motif site locations through DA; and (iii) Given motif site locations, update all parameters including PWMs.

3.1.1 Evolutionary Monte Carlo for module selection

It has been demonstrated that the EMC method is effective for sampling and optimization with functions of binary variables [26]. Conceptually, we should be able to apply EMC directly to select motifs comprising the CRM, but a complication here is that there are many continuous parameters such as the Θ_j's, λ, and $\boldsymbol{\tau}$ that vary in dimensionality when a putative motif in $\mathcal{D}$ is included or excluded from the CRM. We therefore integrate out the continuous parameters analytically and condition on variables $\boldsymbol{a}$ and $\boldsymbol{T}$ when updating the CRM composition. Let $\Omega^{(u)} = (\boldsymbol{\Theta}, \boldsymbol{\rho}, \boldsymbol{\tau}, \lambda)$ denote the set of all parameters in the model, for a fixed $\boldsymbol{u}$. Then, the marginalized conditional posterior probability for a module configuration $\boldsymbol{u}$ is:

$$P(\boldsymbol{u} \,|\, \boldsymbol{a},\boldsymbol{T},\boldsymbol{S}) \propto \pi^{|\boldsymbol{u}|}(1-\pi)^{D-|\boldsymbol{u}|} \int P(S \,|\, \boldsymbol{a},\boldsymbol{T},\Omega^{(u)}) P(\Omega^{(u)} \mid \boldsymbol{u}) d\Omega^{(u)}, \tag{3.2}$$

where only $\boldsymbol{\Theta}$ and $\boldsymbol{\tau}$ are dependent on $\boldsymbol{u}$; and $\boldsymbol{a}$ and $\boldsymbol{T}$ are the sets of locations and types, respectively, of all putative motif sites (for all the D motifs in $\mathcal{D}$). Thus, only when the indicator u_i for the weight matrix Θ_i is 1, do its site locations and types contribute to the computation of (3.2). When we modify the current $\boldsymbol{u}$ by excluding a motif type, its site locations and corresponding motif type indicators are removed from the computation of (3.2).

For EMC, we need to prescribe a set of temperatures, $t_1 > t_2 > \cdots > t_M = 1$, one for each member in the population. Then, we define $\phi_i(\boldsymbol{u}_i) \propto \exp[\log P(\boldsymbol{u}_i \mid \boldsymbol{a}, \boldsymbol{T}, \boldsymbol{S})/t_i]$, and $\phi(\boldsymbol{U}) \propto \prod_{i=1}^{M} \phi_i(u_i)$. The "population" $\boldsymbol{U} = (\boldsymbol{u}_1, \cdots, \boldsymbol{u}_M)$ is then updated iteratively using two types of moves: *mutation* and *crossover*.

In the mutation operation, a unit $\boldsymbol{u}_k$ is randomly selected from the current population and mutated to a new vector $\boldsymbol{v}_k$ by changing the values of some of its bits chosen at random. The new member $\boldsymbol{v}_k$ is accepted to replace $\boldsymbol{u}_k$ with probability $\min(1, r_m)$, where $r_m = \phi_k(\boldsymbol{v}_k)/\phi_k(\boldsymbol{u}_k)$.

In the crossover step, two individuals, $\boldsymbol{u}_j$ and $\boldsymbol{u}_k$, are chosen at random from the population. A crossover point x is chosen randomly over the positions 1 to D, and two new units $\boldsymbol{v}_j$ and $\boldsymbol{v}_k$ are formed by switching between the two individuals the segments on the right side of the crossover point. The two "children" are accepted into the population to replace their parents $\boldsymbol{u}_j$ and $\boldsymbol{u}_k$ with probability $\min(1, r_c)$, where $r_c = \frac{\phi_j(\boldsymbol{v}_j)\phi_k(\boldsymbol{v}_k)}{\phi_j(\boldsymbol{u}_j)\phi_k(\boldsymbol{u}_k)}$. If rejected, the parents are kept unchanged. On convergence, the samples of $\boldsymbol{u}_M$ (for temperature $t_M = 1$) follow the target distribution (3.2).

3.1.2 Sampling motif sites $\boldsymbol{A}$ through recursive DA

The second part of the algorithm consists of updating the motif sites conditional on a CRM configuration (i.e., with $\boldsymbol{u}$ fixed). For simplicity, we describe the method for a single sequence $S = (s_1, \cdots, s_L)$– the same procedure is repeated for all sequences in the data set. For simplicity of notation, we assume that all motifs are of width w. For fixed $\boldsymbol{u}$, let $F(i, j, k, \boldsymbol{u}) = P(s_{[i,j,k]} \mid \Omega^{(u)}, \boldsymbol{u})$ denote the probability of observing the part of the sequence S from position i to j, with a motif of type k $\{k \in \mathcal{D} : u_k = 1\}$ occupying positions from $j - w + 1$ to j ($k = 0$ denotes the background). Let $K = \sum_{k=1}^{D} u_k$ denote the number of motif types in the module. For notational simplicity, let us assume that $\boldsymbol{u}$ represents the set of the first K motifs, indexed 1 through K. Since the motif site updating step is *conditional* given $\boldsymbol{u}$, we drop the subscript $\boldsymbol{u}$ from $F(i, j, k, \boldsymbol{u})$ in the remaining part of the section.

In the *forward summation* step, we recursively calculate the probability of different motif types ending at a position j of the sequence:

$$\begin{aligned} F(1,j,k) = & \left[\sum_{i<j}\sum_{l=1}^{K} F(1,i,l)\tau_{l,k}\, Q(j-i-w;\lambda,w) + P(s_{[1,j-w,0]}|\boldsymbol{\rho})\right] \\ & \times F(j-w+1,j,k). \end{aligned}$$

By convention, the initial conditions are: $F(0,0,k) = 1$, $(k = 0, 1, \cdots, K)$, and $F(i,j,k) = 0$ for $j < i$ and $k > 0$. In the *backward sampling* step, we use Bayes theorem to calculate the probability of motif occurrence at each position, starting from the end of the sequence. If a motif of type k ends at position i in the sequence, the probability that the next motif further ahead in the sequence spans position $(i' - w + 1)$ to i', $(i' \leqslant i - w)$, and is of type k', is:

$$
\begin{aligned}
&P(A_{\cdot,i'-w+1,k'} = 1 \mid \boldsymbol{S}, \Omega, A_{\cdot,i-w+1,k} = 1) \\
&\quad = \frac{F(1,i',k')\; P(s_{[i'+1,i-w,0]}|\boldsymbol{\rho})\; F(i-w+1,i,k)\; Q(i-i'-w;\lambda,w)\; \tau_{k',k}}{F(1,i,k)}.
\end{aligned}
$$

The required expressions have all been calculated in the forward sum.

Finally, given the motif type indicator $\boldsymbol{u}$ and the motif position and type vectors $\boldsymbol{a}$ and $\boldsymbol{T}$, we now update the parameters $\Omega = (\boldsymbol{\Theta}, \boldsymbol{\rho}, \boldsymbol{\tau}, \lambda)$ by a random sample from their joint conditional distribution. Since conjugate priors have been assumed for all parameters, their conditional posterior distributions are also of the same form and are straightforward to simulate from. For example, the posterior of Θ_i will be $\prod_{j=1}^{w} \text{Dirichlet}(\boldsymbol{\beta}_{ij} + \boldsymbol{n}_{ij})$, where $\boldsymbol{n}_{ij}$ is a vector containing the counts of the 4 nucleotides at the jth position of all the sites corresponding to motif type i. For those motifs that have not been selected by the module (i.e., with $u_i = 0$), the corresponding Θ's still follow their prior distribution. The posterior distributions of the other parameters can be similarly calculated using conjugate prior distributions.

3.2 A case-study

We compared the performance of EMCmodule with EM- and Gibbs sampling-based methods in an analysis of mammalian skeletal muscle regulatory sequences [44]. The raw data consist of upstream sequences of lengths up to 5000 bp each corresponding to 24 orthologous pairs of genes in the human and mouse genomes– each of the sequences being known to contain at least one experimentally reported transcription-factor binding site corresponding to one of 5 motif types: MEF, MYF2, SRF, SP1 and TEF. Following the procedure of Thompson et al. [44], we aligned the sequences for each orthologous pair (human and mouse) and retained only the parts that shared a percent identity greater than 65%, cutting down the sequence search space to about 40% of the original sequences.

Using BioProspector and EM (MEME), we obtained initial sets of 100 motifs including redundant ones. The top-scoring 10 motifs from BioProspector and MEME respectively contained 2 and 3 matches to the true motif set (of 5). The Gibbs sampler under a module model [44] found 2 matches, but could find 2 others with a more detailed and precise prior input (the number of sites per motif and motif abundance per sequence), which may not be available in real applications. The best scoring module configuration from EMCmodule contained 3 of the true 5, MYF, MEF2, and SP1, and two uncharacterized motifs. There are few TEF sites matching the reported consensus in these sequences, which may explain why they were not found. The relative error rates for the algorithms were compared using knowledge of the 154 experimentally determined TFBSs [44]. Table 1 shows that EMCmodule significantly cuts down the percentage of false positives in the output, compared to the methods that do not adjust for positional clustering of motifs.

4 Motif discovery in multiple species

Modeling CRMs enhances the performance of *de novo* motif discovery because it allows the use of information encoded by the spatial correlation among TFBS's in the same module. Likewise, the use of multiple genomes enhances motif prediction because it allows the use of information from the evolutionary conservation of TFBS's in related species. Several recent methods employ such information to enhance the power of *cis*-regulatroy analysis. PhyloCon [45] builds multiple alignments among orthologs and extends these alignments to identify motif profiles. CompareProspector [31] biases motif search to more conserved regions based on conservation scores. With a given alignment of orthologs and a phylogenetic tree, EMnEM [34], PhyME [40], and PhyloGibbs [39] detect motifs based on more comprehensive evolutionary models for TFBS's. When evolutionary distances among the genomes are too large for the orthologous sequences to be reliably aligned, Li and Wong [25] proposed an ortholog sampler that finds motifs in multiple species independent of ortholog alignments. Jensen *et al.* [19] used a Bayesian clustering approach to combine TF binding motifs from promoters of multiple orthologs.

Table 1: Error rates for module prediction methods.

Method	MEF	MYF	SP1	SRF	Total	SENS	SPEC	TSpec
EM	0	1	21	0	161	0.14	0.14	0.20
BioProspector	6	1	8	1	155	0.10	0.10	0.36
GS	6	6	2	1	84	0.10	0.25	0.44
GS^{p*}	14	14	4	6	162	0.25	0.23	0.60
EMCmodule	12	12	5	7	180	0.23	0.20	0.67
True	**32**	**50**	**44**	**28**	**154**	–	–	–

SENS (sensitivity) ≡ (# predicted true positives)/(# true positives); SPEC (specificity) ≡ (# predicted true positives)/(# predicted sites). TSpec: Total specificity– the fraction of the predicted motif *types* that "correspond" to known motifs (match in at least 80% of all positions). The Gibbs sampler (GS) requires the total number of motif types to be specified (here = 5). GS^{p*} denotes the GS using a strong informative prior.

In this section, we review in details the coupled hidden Markov model (c-HMM) developed by Zhou and Wong [52] (ZW hereafter) as an example for motif discovery that utilizes information from cis-regulatory modules and multiple genomes. The authors use a hidden Markov model (HMM) to capture the colocalization tendency of multiple TFBS's within each species, and then couple the hidden states (which indicate the locations of modules and TFBS's within the modules) of these HMMs through multiple-species alignment. They developed evolutionary models separately for background nucleotides and for motif binding sites, in order to capture the different degrees of conservation among the background and among the binding sites. A Markov chain Monte Carlo algorithm is devised for sampling CRMs and their component motifs simultaneously from their joint posterior distribution.

4.1 The coupled hidden Markov model

The input data consist of upstream or regulatory sequences of n (co-regulated) genes from N species, i.e., a total of $n \times N$ sequences. Assuming these genes are regulated by CRMs composed of binding sites of K TFs, one wants to find these TFBS's and their motifs (PWMs). Assume that the N species are closely related in the sense that their orthologous TFs share the same binding motif, which applies to groups of species within mammals, or within Drosophila, etc.

Let us first focus on the module structure in one sequence. Assume that the sequence is composed of two types of regions, modules and background. A module contains multiple TFBS's separated by background nucleotides, while background regions contain only background nucleotides. Accordingly, we assume that the sequence is generated from a hidden Markov model with two states, a module state (M) and a background state (B). In a module state, the HMM either emits a nucleotide from the background model (of nucleotide preference) θ_0, or it emits a binding site of one of the K motifs (PWMs) $\Theta_1, \Theta_2, \cdots, \Theta_K$. The probability for emission from θ_0 and $\Theta_k (k = 1, 2, \cdots, K)$ is denoted by q_0 and q_k, respectively ($\sum_{k=0}^{K} q_k = 1$) (Figure 4.1A). Note that a module state can be further decomposed to $K + 1$ states, corresponding to within-module background (M_0) and K motif binding sites (M_1 to M_K), i.e. $M = \{M_0, M_1, \cdots, M_K\}$. Assuming that the width of motif k is w_k, a binding site of this motif, a piece of sequence of length w_k, is treated as one state of M_k as a whole ($k = 1, 2, \cdots, K$). The transition probability from a background to a module state is r, i.e., the chance of initiating a new module is r. The transition probability from a module state to a background state is t, i.e., the expected length of a module is $1/t$. Denote the transition matrix by

$$T = \begin{bmatrix} T(B,B) & T(B,M) \\ T(M,B) & T(M,M) \end{bmatrix} = \begin{bmatrix} 1-r & r \\ t & 1-t \end{bmatrix}. \tag{4.1}$$

This model can be viewed as a stochastic version of the hierarchical mixture model defined in [51].

The HMMs in different orthologs are coupled through multiple alignment, so that the hidden states of aligned bases in different species are collapsed into a common state (Figure B). For instance, the nucleotides of state 4 in the three orthologs are aligned in Figure B. Thus these three states are collapsed into one state, which determines whether these aligned nucleotides are background or binding sites of a motif. (Note that these aligned nucleotides in different orthologs are not necessarily identical.) Here hidden states refer to the decomposed states, i.e. B and M_0 to M_K, which specify the locations of modules and motif sites. This coupled hidden Markov model (c-HMM hereafter) has a natural graphical model representation (lower panel of Figure B), in which each state is represented by a node in the graph and the arrows specify the dependence among them. The transition (conditional) probabilities for nodes with a single parental node are defined by the same T in equation 4.1. We define the conditional probability for a node with multiple parents as follows: If node Y has m parents, each in state

Y_i $(i = 1, 2, \cdots, m)$, then we have

$$P(Y|Y_1, \cdots, Y_m) = \frac{C_B}{m} T(B, Y) + \frac{C_M}{m} T(M, Y), \tag{4.2}$$

where C_B and C_M are the numbers of the parents in states B and M, respectively $(m = C_B + C_M)$. This equation shows that the transition probability to a node with multiple parents is defined as the weighted average from the parental nodes in background states and module states. The same emission model described in the previous paragraph is used for unaligned states. For aligned (coupled) states, ZW assume star-topology evolutionary models with one common ancestor. The c-HMM first emits (hidden) ancestral nucleotides by the emission model defined in Figure 2A given the coupled hidden states. Then, different models are used for the evolution from the ancestral to descendant nucleotides depending on whether they are background or TFBS's.

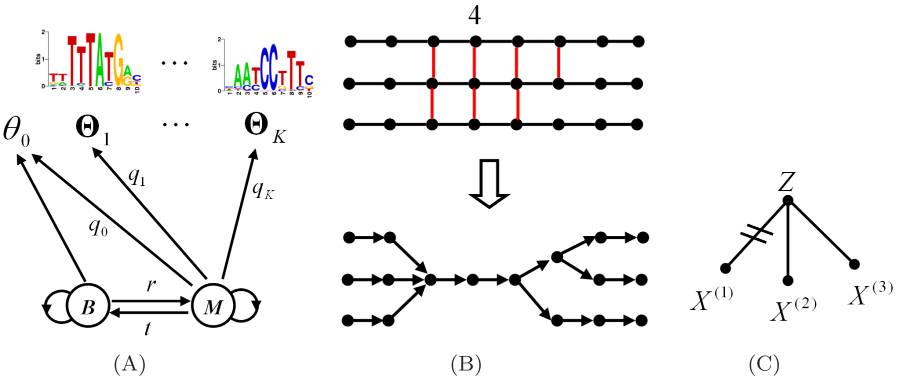

Figure 2: The coupled hidden Markov model (c-HMM). (A) The HMM for module structure in one sequence. (B) Multiple alignment of three orthologous sequences (upper panel) and its corresponding graphical model representation of the c-HMM (lower panel). The nodes represent the hidden states. The vertical bars in the upper panel indicate that the nucleotides emitted from these states are aligned and thus collapsed in the lower panel. Note that a node will emit w_k nucleotides if the corresponding state is M_k $(k = 1, \cdots, K)$. (C) The evolutionary model for motifs using one base of a motif as an illustration. The hidden ancestral base is Z, which evolves to three descendant bases $X^{(1)}$, $X^{(2)}$, and $X^{(3)}$. Here the evolutionary bond between $X^{(1)}$ and Z is broken, implying that $X^{(1)}$ is independent of Z. The bond between $X^{(2)}$ and Z and that between $X^{(3)}$ and Z are connected, which means that $X^{(2)} = X^{(3)} = Z$.

A neutral substitution matrix is used for the evolution of aligned background nucleotides, both within and outside of modules, with a transition rate of α and a transversion rate of β:

$$\Phi = \begin{bmatrix} 1-\mu_b & \beta & \alpha & \beta \\ \beta & 1-\mu_b & \beta & \alpha \\ \alpha & \beta & 1-\mu_b & \beta \\ \beta & \alpha & \beta & 1-\mu_b \end{bmatrix}, \tag{4.3}$$

where the rows and columns are ordered as A, C, G, and T, and $\mu_b = \alpha + 2\beta$ is defined as the background mutation rate. ZW assume an independent evolution for each position (column) of a motif under the nucleotide substitution model of Felsenstein [13]. Suppose the weight vector of a particular position in the motif is θ. The ancestral nucleotide, denoted by Z, is assumed to follow a discrete distribution with the probability vector θ on $\{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$. If X is a corresponding nucleotide in a descendant species, then either X inherits Z directly (with probability μ_f) or it is generated independently from the same weight vector θ (with probability $1-\mu_f$). The parameter μ_f, which is identical for all the positions within a motif, reflects the mutation rate of the TFBS's. This model takes PWM into account in the binding site evolution, which agrees with the non-neutral constraint of TFBS's that they are recognized by the same protein (TF). It is obvious that under this model, the marginal distribution of any motif column is identical in all the species. This evolutionary model introduces another hidden variable which indicates whether X is identical to or independent of Z for each base of an aligned TFBS. These indicators are called evolutionary bonds between ancestral and descendent bases (Figure 4.1C). If $X = Z$, we say that the bond is connected; If X is independent of Z, we say that the bond is broken.

4.2 Gibbs sampling and Bayesian inference

The full model involves the following parameters: the transition matrix T defined in equation 4.1, the mixture emission probabilities $q_0, q_1, \cdots, q_K$, the motif widths $w_1, \cdots, w_K$, the PWMs $\Theta_1, \cdots, \Theta_K$, the background models for ancestral nucleotides and all current species, and the evolutionary parameters α, β, and μ_f. The number of TFs, K, and the expected module length, L, are taken as input, and the transition probability t is fixed to $t = 1/L$ in T. Compared to the HMx model in [51], this model has three extra free parameters, α, β, and μ_f, related to the evolutionary models. Independent Poisson priors are put on motif widths and flat Dirichlet distributions are used as priors for all the other parameters. With a given alignment for each ortholog group, one may treat as missing data the locations of modules and motifs (i.e. the hidden states), the ancestral sequences, and the evolutionary bonds. ZW develop a Gibbs sampler (called MultiModule, hereafter) to sample from the joint posterior distribution of all the unknown parameters and missing data. To consider the uncertainty in multiple alignment, they adopt an HMM-based multiple alignment [3, 22] conditional on the current parameter values. This is achieved by adding a Metropolis-Hastings step in the Gibbs sampler to update these alignments dynamically according to the current sampled parameters, especially the background substitution matrix Φ (equation 4.3). In summary, the input data of MultiModule are groups of orthologous sequences, and the program builds an initial alignment of each ortholog group by a standard HMM-based multiple alignment algorithm. Then each iteration of MultiModule is composed of three steps: (1) Given alignments and all the other missing data, update motif widths and other parameters by their conditional posterior distributions; (2) Given current parameters, with probability u, update the alignment of each ortholog group; (3) Given alignments and parameters, a dynamic programming approach is used to sample module and motif locations, ancestral sequences, and evolutionary bonds. The probability u is typically chosen in the range $[0.1, 0.3]$. (See ref[52] for the details of the Gibbs sampling of MultiModule.)

Motif and module predictions are based on their marginal posterior distributions constructed by the samples generated by MultiModule after some burn-in period (usually the first 50% of iterations). The width of each motif is estimated by its rounded posterior mean. MultiModule records the following posterior probabilities for each sequence position in all the species: (1) P_k, the probability that the position is within a site for motif k, i.e., the hidden state is M_k ($k = 1, 2, \cdots, K$); (2) P_m, the probability that the position is within a module, i.e., the hidden state is M; (3) P_a, the probability that the position is aligned. All the contiguous segments with $P_k > 0.5$ are aligned (and extended if necessary) to generate predicted sites of motif k given the estimated width w_k. The corresponding average P_a over the bases of a predicted site is reported as a measure of its conservation. All the contiguous regions with $P_m > 0.5$ are collected as candidates for modules, and a module is predicted if the region contains at least two predicted motif binding sites. The boundary of a predicted module is defined by the first and last predicted binding sites it contains.

Under the c-HMM, if one fixes $r = 1 - t = 1$ in the transition matrix T (equation 4.1), then MultiModule reduces to a motif discovery method, assuming the existence of K motifs in the sequences. This setting is useful when the motifs do not form modules, and it is defined as the motif mode of MultiModule in [52].

4.3 Simulation studies

Here we present the simulation studies conducted in [52] to illustrate the use of MultiModule. The authors used the following model to simulate data sets in this study: They generated 20 hypothetical ancestral sequences, each of length 1000 bps. Twenty modules, each of 100 bps and containing one binding site of each of the three TFs, were randomly placed in these sequences. TFBS's were simulated from their known weight matrices with logo plots [38] shown in Figure 3. Then based on the choices of the background mutation rate μ_b (with $\alpha = 3\beta$ in equation 4.3) and the motif mutation rate μ_f, they generated sequences of three descendant species according to the evolutionary models in section 4.1. The indel (insertion-deletion) rate was fixed to $0.1\mu_b$. After the ancestral sequences were removed, each data set finally contains 60 sequences from three species. The simulation study was composed of two groups of data sets, and in both groups they set $\mu_f = 0.2\mu_b$ but varied the value of μ_b. In the first group, they set $\mu_b = 0.1$ to mimic the case where species are evolutionarily close. In the second group, they set $\mu_b = 0.4$ to study the situation for remotely related species. For each group 10 data sets were generated independently.

MultiModule was applied to these data sets under three different sets of program parameters: (A) Module mode, $L = 100, u = 0.2$; (B) Motif mode, $u = 0.2$; (C) Motif mode, $u = 0$. For each set of parameters, they ran MultiModule for 2,000 iterations with $K = 3$, searching both strands of the sequences. Initial alignments were built by ordinary HMM based multiple alignment methods. If $u = 0$, these initial alignments were effectively fixed along the iterations.

The results are summarized in Table 2, which includes the sensitivity, the

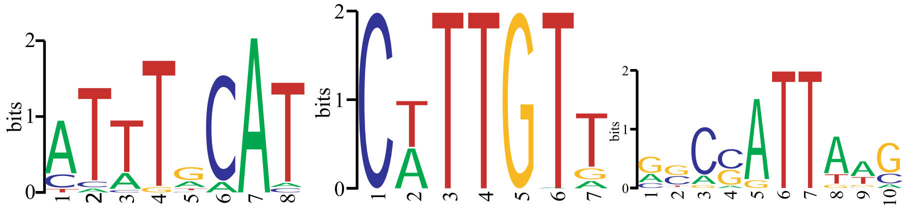

Figure 3: Logo plots for the motifs in the simulated studies: (A) Oct4, (B) Sox2, and (C) Nanog

specificity, and an overall measurement score of the performance, defined as the geometric average of the sensitivity and specificity. One sees that updating alignments improves the performance for both $\mu_b = 0.1$ and 0.4, and the improvement is more significant for the latter setting (compare results of B and C in Table 2). The reason is that the uncertainty in alignments for the cases with $\mu_b = 0.4$ is higher than that for $\mu_b = 0.1$, and thus updating alignments, which aims to average over different possible alignments, has a greater positive effect. Considering module structure shows an obvious improvement for $\mu_b = 0.1$, but it is only slightly better than running the motif mode for $\mu_b = 0.4$ (compare A and B in Table 2). For $\mu_b = 0.4$, MultiModule found all the three motifs under both parameter settings (A and B) for five data sets, and the predictions in A with an overall score of 70% definitely outperformed that in B with an overall score of 58%. For the other five data sets, no motifs were identified in setting A, but in setting B (motif mode) subsets of the motifs were still identified for some of the data sets. This may be caused by the slower convergence of MultiModule in setting A, because of its higher model complexity, especially when the species are farther apart. One possible quick remedy of this is to use the output from setting B as initial values for setting A, which will be a much better starting point for the posterior sampling.

Table 2: Results for the simulation study

	Oct4 (60)	Sox2 (60)	Nanog (60)	Three motifs in total		
	N_2/N_1	N_2/N_1	N_2/N_1	Sen	Spe	Overall
(A)	38.7/57.4	51.6/66.0	40.8/46.3	73%	77%	75%
(B)	27.7/45.3	52.2/91.3	27.6/37.8	60%	62%	61%
(C)	22.2/36.7	42.4/89.6	23.6/39.0	49%	53%	51%
(A)	18.8/24.8	22.7/30.6	21.0/33.9	35%	70%	49%
(B)	9.3/29.4	34.0/51.1	21.7/31.6	36%	58%	46%
(C)	5.1/8.1	14.2/18.2	8.3/14.2	15%	68%	32%

N_2 and N_1 refer to the numbers of correct and total predictions for each motif, respectively. TF names are followed by the numbers of true sites in parentheses. The upper and lower halves refer to the average results over 10 independently generated data sets with $\mu_b = 0.1$ and 0.4, respectively. "Overall" is the geometric average of sensitivity ("Sen") and specificity ("Spe"). For each data set, the optimal results (in terms of overall score) among three independent runs under the same parameters were used for the calculation of averages. Parameter sets (A), (B), (C) are defined as: (A) Module mode, $L = 100, u = 0.2$; (B) Motif mode, $u = 0.2$; (C) Motif mode, $u = 0$.

MultiModule has also been tested on two well-annotated data sets from the human and the Drosophila genomes, and the results were compared to experimental validations. Please see [52] for more details.

5 Motif learning on ChIP-chip data

In recent years, a number of computational approaches have been developed to combine motif discovery with gene expression or ChIP-chip data, e.g., [7, 9, 10, 20]. These approaches identify a group of motifs, and then correlate expression values (or ChIP-intensity) to the identified motifs via linear or other regression techniques.

The use of ChIP-chip data has great advantage in understanding TF-DNA binding: Such data not only provide hundreds or even thousands of high resolution TF binding regions, but also give quantitative measures of the binding activity (ChIP-enrichment) for such regions. In this section, we introduce a new approach developed by Zhou and Liu [50] (ZL hereafter) for motif learning from ChIP-chip data, to illustrate the general framework of this type of methods. In contrast to many approaches that directly build generative statistical models in the sequence space such as those discussed in the previous sections, ZL map each ChIP-chip binding region into a feature space composed of generic features, background frequencies, and a set of motif scores derived from both known motifs documented in biological databases and motifs discovered *de novo*. Then, they apply the Bayesian additive regression trees (BARTs) [8] to learn the relationship between ChIP-intensity and these sequence features. As the sum of a set of trees, the BART model is flexible enough to approximate almost any complex relationship between responses and covariates. With carefully designed priors, each tree is constrained to be a weak learner, only contributing a small amount to the full model, which effectively prevents the BART model from overfitting. The learning of the model is carried out by Markov chain Monte Carlo sampling of the posterior BART distribution that lives in the additive tree space, which serves to average over different BART models. These posterior draws of BARTs also provide a natural way to rank the importance of each sequence feature in explaining ChIP-intensity.

Compared to other motif learning approaches with auxiliary data, there are at least two unique features of the Zhou-Liu method [50]. First, the features (or covariates) used in the method contain not only the discovered motifs, but also known motifs, background word frequencies, and other features such as the GC content and cross-species conservation. Second, the additive regression tree model is more flexible and robust than the regression methods used in the other approaches. These advantages will be illustrated in the application of this method to a recently published genome-wide human ChIP-chip data set.

Consider a set of n DNA sequences $\{\mathbf{S}_1, \mathbf{S}_2, \cdots, \mathbf{S}_n\}$, each with a ChIP-chip intensity y_i that measures the level of enrichment of that segment (fold changes) compared to the normal genomic background. In principle, the y_i serves as a surrogate of the binding affinity of the TF to the corresponding DNA segment in the genome. We write $\{(y_i, \mathbf{S}_i)$, for $i = 1, 2, \cdots, n\}$. For each $\mathbf{S}_i$, we extract p numerical features $\mathbf{x}_i = [x_{i1}, \cdots, x_{ip}]$, and transform the dataset to $\{(y_i, \mathbf{x}_i)\}_{i=1}^n$, on which we "learn" a relationship between y_i and $\mathbf{x}_i$ using the BART model. The details on how to extract features from each sequence $\mathbf{S}_i$ will be described in section 5.1. In comparison to the standard statistical learning problem, two novel features of the problem described here are that (a) the response variable y_i is continuous instead of categorical; and (b) the features are not given *a priori*, but need to be produced from the sequences by the researcher.

5.1 Feature extraction

Zhou and Liu [50] extract three categories of sequence features from a repeat-masked DNA sequence, the generic, the background, and the motif features. The generic features include the length, the GC content, and the average conservation of the sequence. For background features, they compute the number of occurrences of each k-mer (only for $k = 2$ and 3 in this paper) in the sequence. They count both forward and backward strands of the DNA sequence, and merge the counts of each k-mer and its reverse complement. For each value of k, if there are C_k distinct words after merging reverse complements, only the frequencies of $(C_k - 1)$ of them will be included in the feature vector since the last one is uniquely determined by the others. Note that the zeroth order frequency ($k = 1$) is equivalent to the GC content.

The motif features are extracted from a compiled set of motifs, each paramet rized by a PWM. The compiled set includes known motifs from TF databases such as TRANSFAC [46] or JASPAR [37], and *new* motifs found from the positive ChIP sequences in the data set of interest using a *de novo* motif search tool. ZL fit a segment-wise homogeneous first-order Markov chain as the background sequence model [27], which helps to account for the heterogeneous nature of genomic sequences such as regions of low complexities (eg. GC/AT rich). Intuitively, this model assumes that the sequence in consideration can be segmented into an unknown number of pieces and within each piece the nucleotides follow a homogeneous first-order Markov chain. Using a Bayesian formulation and Markov chain Monte Carlo, one estimates the background transition probability of each nucleotide. Suppose the current sequence is $\mathbf{S} = R_1 R_2 \cdots R_L$, the PWM of a motif of width w is $\mathbf{\Theta} = \Theta_i(j)$ $(i = 1, \cdots, w, j = \mathrm{A, C, G, T})$, and the background transition probability of R_n given R_{n-1} is $\theta_0(R_n | R_{n-1})$ $(1 \leqslant n \leqslant L)$. For each w-mer in $\mathbf{S}$, say $R_n \cdots R_{n+w-1}$, we calculate a probability ratio

$$r = \prod_{i=1}^{w} \frac{\Theta_i(R_{n+i-1})}{\theta_0(R_{n+i-1} | R_{n+i-2})}. \tag{5.1}$$

Considering both strands of the sequence, we have $2 \times (L - w + 1)$ such ratios. Then the motif score for this sequence is defined as $\log(\sum_{k=1}^m r_{(k)}/L)$, where $r_{(k)}$ is the kth ratio in descending order. In [50], the authors take $m = 25$, i.e., the top 25 ratios are included.

5.2 Bayesian additive regression trees

Here we give a brief review of the Bayesian additive regression tree (BART) model developed in [8]. Let Y be the response variable and $\mathbf{X} = [X_1, \cdots, X_p]$, the feature vector. Let T denote a binary tree with a set of interior and terminal nodes. Each interior node is associated with a binary decision rule based on only one feature, typically of the form $\{X_j \leqslant a\}$ or $\{X_j > a\}$, for $1 \leqslant j \leqslant p$. Suppose the number of terminal nodes of T is B. Then, the tree partitions the feature space into B disjoint regions, each associated with a parameter μ_b $(b = 1, \cdots, B)$ (see Figure 4 for an illustration). Consequently, the relationship between Y and $\mathbf{X}$ is approximated by a piece-wise constant function with B distinct pieces. Let $M = [\mu_1, \cdots, \mu_B]$, and we denote this tree-based piece-wise constant function by $g(\mathbf{X}, T, M)$. The additive regression tree model is simply a sum of N such piece-wise constant functions:

$$Y = \sum_{m=1}^{N} g(\mathbf{X}, T_m, M_m) + \epsilon, \quad \epsilon \sim N(0, \sigma^2), \tag{5.2}$$

in which each tree T_m is associated with a parameter vector M_m $(m = 1, \cdots, N)$. The number of trees N is usually large (100 to 200), which makes the model flexible enough to approximate a complex relationship between Y and $\mathbf{X}$. We assume that each observation $\{(y_i, \mathbf{x}_i)\}$, $i = 1, \cdots, n$, follows Eq. (5.2) and is independent of each other.

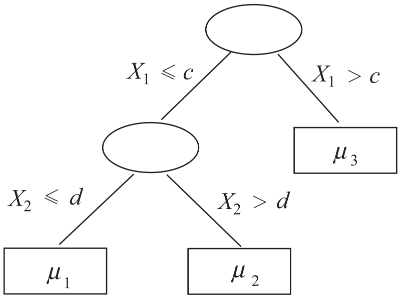

Figure 4: A regression tree with two interior and three terminal nodes. The decision rules partition the feature space into three disjoint regions: $\{X_1 \leqslant c, X_2 \leqslant d\}, \{X_1 \leqslant c, X_2 > d\}$, and $\{X_1 > c\}$. The mean parameters attached to these regions are μ_1, μ_2, and μ_3, respectively

To complete a Bayesian inference based on model (5.2), one needs to prescribe prior distributions for both the tree structures and the associated parameters, M_m and σ^2. The prior distribution for the tree structure is specified conservatively in [8] so that the size of each tree is kept small, which forces it to be a weak learner. The priors on M_m and σ^2 also contribute to preventing from overfitting. In particular, the prior probability for a tree with 1, 2, 3, 4, and $\geqslant 5$ terminal nodes is 0.05, 0.55, 0.28, 0.09, and 0.03, respectively.

Chipman *et al.* [8] developed a Markov chain Monte Carlo approach (BART

MCMC) to sample from the posterior distribution

$$P(\{(T_m, M_m)\}_{m=1}^N, \sigma^2 \mid \{(y_i, \mathbf{x}_i)\}_{i=1}^n). \tag{5.3}$$

Note that the tree structures are also updated along with MCMC iterations. Thus, the BART MCMC generates a large number of samples of additive trees, which form an ensemble of models. Now given a new feature vector $\mathbf{x}^*$, instead of predicting its response y^* based on the "best" model, BART predicts y^* by the average response of all sampled additive trees. More specifically, suppose one runs BART MCMC for J iterations after the burin-in period, which generates J sets of additive trees. For each of them, BART has one prediction: $y^{*(j)} = \sum_{m=1}^N g(\mathbf{x}^*, T_m^{(j)}, M_m^{(j)})$ $(j = 1, \cdots, J)$. These J predicted responses may be used to construct a point estimate of y^* by the plain average, as used in the following applications, or an interval estimate by the quantiles. Thus, BART has the nature of Bayesian model average.

5.3 Application to human ChIP-chip data

Zhou and Liu [50] applied BART to two recently published ChIP-chip data sets of the TFs Oct4 and Sox2 in human embryonic stem (ES) cells [5]. The performance of BART was compared with those of linear regressions [9], MARS [14, 10], and neural networks, respectively, based on ten-fold cross validations. The DNA microarray used in [5] covers -8 kb to $+2$kb of $\sim$17,000 annotated human genes. A Sox-Oct composite motif (Figure 8) was identified consistently in both sets of positive ChIP-regions using *de novo* motif discovery tools (e.g., [23]). This motif is known to be recognized by the protein complex of Oct4 and Sox2, the target TFs of the ChIP-chip experiments. Combined with all the 219 known high-quality PWMs from TRANSFAC and the PWMs of 4 TFs with known functions in ES cells from the literature, a final list of 224 PWMs were compiled for motif feature extraction. Here we present their cross-validation results on the Oct4 ChIP-chip data as a comparative study of several competing motif learning approaches.

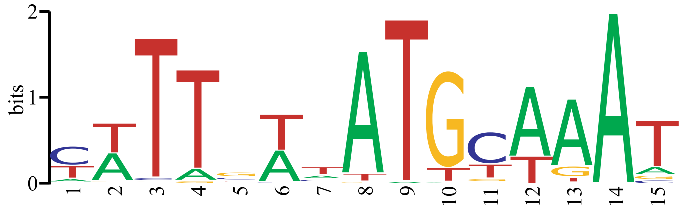

Figure 5: The Sox-Oct composite motif discovered in the Oct4 positive ChIP-regions

Boyer *et al.* [5] reported 603 Oct4-ChIP enriched regions (positives) in human ES cells. ZL randomly selected another 603 regions with the same length distribution from the genomic regions targeted by the DNA microarray (negatives). Note that each such region usually contains two or more neighboring probes on the array. A ChIP-intensity measure, which is defined as the average array-intensity ratio of ChIP samples over control samples, is attached to each of the 1206 ChIP-regions. We treat the logarithm of the ChIP-intensity measure as the response variable, and those features extracted from the genomic sequences as explanatory variables. There are a total of 1206 observations with $224 + 45 = 269$ features (explanatory variables) for this Oct4 data set.

ZL used the following methods to perform statistical learning on this data set: (1) LR-SO, linear regression using the Sox-Oct composite motif only; (2) LR-Full, linear regression using all the 269 features; (3) Step-SO, stepwise linear regression starting from LR-SO; (4) Step-Full, stepwise linear regression starting from LR-Full; (5) NN-SO, neural networks with the Sox-Oct composite motif feature as input; (6) NN-Full, neural networks with all the features as input; (7) MARS, multivariate adaptive regression splines using all the features with different tuning parameters; (8) BART with different number N of trees ranging from 20 to 200.

In Step-SO, one started from the LR-SO model, and used the stepwise method (with both forward and backward steps) to add or delete features in the linear regression model based the AIC criterion (see R function "step"). The Step-Full was performed similarly, but starting from the LR-Full model. For neural networks, ZL used the R package "nnet" with different combinations of the number of hidden nodes (2, 5, 10, 20, 30) and weight decay (0, 0.5, 1.0, 2.0). For MARS, they used the function "mars" in the R package "mda" made by Hastie and Tibshirani, with up to two-way interactions and a wide range of penalty terms. For BART, they ran 20,000 iterations after a burn-in period of 2,000 iterations, and used the default settings in the R package "BayesTree" for all other parameters.

The ten-fold cross validation procedure in [50] was conducted as follows. They first divided the observations into ten subgroups of equal sizes at random. Each time, one subgroup (called "the test sample") was left out and the remaining nine subgroups (called "the training sample") were used to train a model using the stated method. Then, they predicted the responses for the test sample based on the trained model and compared them with the observed responses. This process was continued until every subgroup had served as the test sample once. In [50], the authors used the correlation coefficient between the predicted and observed responses as a measure of the goodness of a model's performance. This measure is invariant under linear transformation, and can be intuitively understood as the fraction of variation in the response variable that can be explained by the features (covariates). We call this measure the CV-correlation (or CV-cor) henceforth.

The cross validation results are given in Table 3. The average CV-correlation (over 10 cross validations) of LR-SO is 0.446, which is the lowest among all the linear regression methods. Since all the other methods use more features, this shows that sequence features other than the target motif itself indeed contribute to the prediction of ChIP-intensity. Among all the linear regression methods, Step-SO achieves the highest CV-cor of 0.535. Only the optimal performance among all the combinations of parameters were reported for the neural networks. However, even these optimal results are not satisfactory. The NN-SO showed a slight improvement in CV-cor over that of LR-SO. For different parameters (the number of hidden nodes and weight decay), NN-SO showed roughly the same performance except for those with 20 or more hidden nodes and weight decay =

0, which overfitted the training data. The neural network with all the features as input encountered a severe overfitting problem, resulting in CV-cor's < 0.38, even worse than that of LR-SO. In order to relieve the overfitting problem for NNs, ZL reduced the input independent variables to those selected by the stepwise regression (about 45), and employed a weight decay of 1.0 with 2, 5, 10, 20, or 30 hidden nodes. More specifically, for each training data set, they performed Step-SO followed by NNs with features selected by the Step-SO as input. Then they calculated the CV-cor's for the test data. We call this approach Step+NN, and it reached an optimal CV-cor of 0.463 with 2 hidden nodes.

ZL applied MARS to this data set under two settings: the one with no interaction terms ($d = 1$) and the one considering two-way interactions ($d = 2$). For each setting, they chose different values of the penalty λ, which specifies the cost per degree of freedom. In the first setting ($d = 1$), they set the penalty $\lambda = 1, 2, \cdots, 10$, and observed that the CV-cor reaches its maximum of 0.580 when $\lambda = 6$. Although this optimal result is only slightly worse than that of BART (Table 3), we note that the performance of MARS was very sensitive to the choice of λ. With $\lambda = 2$ or 1, MARS greatly overfitted the training data, and the CV-cor's dropped to 0.459 and 0.283, respectively, which are almost the same or even worse than that of LR-SO. MARS with two-way interactions ($d = 2$) showed unsatisfactory performance for $\lambda \leqslant 5$ (i.e., CV-cor < 0.360). They then tested λ in the range of $[10, 50]$ and found the optimal CV-cor of 0.561 when $\lambda = 20$.

Table 3: Ten-fold cross validations for log-ChIP-intensity prediction on the Oct4 ChIP-chip data

Method	Cor	Imprv	Method	Cor	Imprv
LR-SO	0.446	0%	LR-Full	0.491	10%
Step-SO	0.535	20%	Step-Full	0.513	15%
NN-SO	0.468	5%	Step+NN	0.463	4%
MARS1,6	0.580	30%	MARS1,1	0.283	−37%
MARS2,20	0.561	26%	MARS2,4	0.337	−24%
BART20	0.592	33%	BART40	0.599	34%
BART60	0.596	34%	BART80	0.597	34%
BART100	0.600	35%	BART120	0.599	34%
BART140	0.599	34%	BART160	0.594	33%
BART180	0.595	33%	BART200	0.593	33%
Step-M	0.456	2%	BART-M	0.510	14%
MARS1,6-M	0.511	15%	MARS2,20-M	0.478	7%

Reported here are the average CV-correlations (Cor). LR-SO, LR-Full, Step-SO, Step-Full, NN-SO, and Step+NN are defined in the text. MARSa,b refers to the MARS with $d = a$ and $\lambda = b$. BARTm is the BART with m trees. Step-M, MARSa,b-M, and BART-M are Step-SO, the optimal MARS, and BART100 with only motif features as input. The improvement ("Imprv") is calculated by Cor/Cor(LR-SO)-1.

Notably, BARTs with different number of trees outperformed all the other methods uniformly. BARTs reached a CV-cor of about 0.6, indicating a greater than 30% of improvement over that of LR-SO and the optimal NN, and more than 10% of improvement over the best performance of the stepwise regression method. In addition, the performance of BART was very robust for different choices of the number of trees included. This is a great advantage over MARS, whose performances depended strongly on the choice of the penalty parameter λ, which is typically difficult for the user to set *a priori*. Compared to NNs, BART is much less prune to overfitting, which may be attributable to its Bayesian model averaging nature with various conservative prior specifications.

To further illustrate the effect of non-motif features, ZL did the following comparison. They excluded non-motif features from the input, and applied BART with 100 trees, MARS ($d = 1, \lambda = 6$), MARS ($d = 2, \lambda = 20$), and Step-SO to the resulting data set to perform ten-fold cross validations. In other words, the feature vectors contained only the 224 motif features. The CV-correlations of these approaches are given in Table 3, denoted by BART-M, MARS1,6-M, MARS2,20-M and Step-M, respectively. It is observed that the CV-correlations decreased substantially (about 12% to 15%) compared to the corresponding methods with all the features. One almost obtains no improvement (2%) in predictive power by taking more motif features in the linear regression. However, if the background and other generic features are incorporated, the stepwise regression improves dramatically (20%) in its prediction. This does not mean that the motif features are not useful, but their effects need to be considered in conjunction with background frequencies.

Step-M is equivalent to MotifRegressor [9] and MARS-M is equivalent to MARSMotif [10] with all the known and discovered (Sox-Oct) motifs as input. Thus, this study implies that BART with all three categories of features outperformed MotifRegressor and MARSMotif by 32% and 17% in CV-correlation, respectively.

6 Using nucleosome positioning information in motif discovery

Generally TF-DNA binding is represented as a one-dimensional process; however, in reality, binding occurs in three dimensional space. Biological evidence [32] shows that much of DNA consists of repeats of regions of about 147 bp wrapped around nucleosomes, separated by stretches of DNA called *linkers*. Recent techniques [47] based on high density genome tiling arrays have been used to experimentally measure genomic positions of nucleosomes, in which the measurement "intensities" indicate how likely that locus is to be nucleosome-bound. These studies suggest that nucleosome-free regions highly correlate with the location of functional TFBSs, and hence can lead to significant improvement in motif prediction, if considered.

Genome tiling arrays pose considerable challenges for data analysis. These arrays involve short overlapping probes covering the genome, which induces a spatial data structure. Although hidden Markov models or HMMs [35] may be used to accommodate such spatial structure, they induce an exponentially decaying distribution of state lengths, and are not directly appropriate for assessing structural features such as nucleosomes that have restrictions in physical dimension.

For instance, in Yuan et al. [47], the tiling array consisted of overlapping 50-mer oligonucleotide probes tiled every 20 base pairs. The nucleosomal state can thus be assumed to be represented by about 6 to 8 probes, while the linker states had no physical restriction. Since the experiment did not succeed in achieving a perfect synchronization of cells, additionally a third "delocalized nucleosomal" state was modeled, which had intensities more variable in length and measurement magnitude than expected for nucleosomal states.

Here, we describe a general framework for determining chromatin features from tiling array data and using this information to improve *de novo* motif prediction in eukaryotes [18].

6.1 A hierarchical generalized HMM (HGHMM)

Assume that the model consists of K $(= 3)$ states. The possible length duration in state k, $(k = 1, \cdots, K)$ is given by the set $D_k = \{r_k, \cdots, s_k\} \subset \mathbb{N}$ (i.e. $\mathbb{N}$ denotes the set of positive integers).

The generative model for the data is now described. The initial distribution of states is characterized by the probability vector $\boldsymbol{\pi} = (\pi_1, \cdots, \pi_K)$. The probability of spending time d in state k is given by the distribution $p_k(d|\boldsymbol{\phi})$, $d \in D_k$ $(1 \leqslant k \leqslant K)$, characterized by the parameter $\boldsymbol{\phi} = (\phi_1, \cdots, \phi_K)$. For the motivating application, $p_k(d)$ is chosen to be a truncated negative binomial distribution, between the range specified by each D_k. The latent state for probe i is denoted by the variable Z_i $(i = 1, \cdots, N)$. Logarithms of spot measurement ratios are denoted by y_{ij} $(1 \leqslant i \leqslant N;\ 1 \leqslant j \leqslant r)$ for N spots and r replicates each. Assume that given the (unobservable) state Z_i, y_{ij}'s are independent, with $y_{ij}|Z_i = k \sim g_k(\ \cdot\ ; \xi_{ik}, \sigma^2_{ik})$. For specifying g_k, a hierarchical model is developed that allows robust estimation of the parameters. Let $\boldsymbol{\mu} = (\mu_1, \cdots, \mu_K)$ and $\boldsymbol{\Sigma} = \{\sigma^2_{ik};\ 1 \leqslant i \leqslant N;\ 1 \leqslant k \leqslant K\}$. Assume $y_{ij}|Z_i = k, \xi_{ik}, \sigma^2_{ik} \sim N(\xi_{ik}, \sigma^2_{ik})$, $\xi_{ik}|\mu_k, \sigma^2_{ik} \sim N(\mu_k, \tau_0\sigma^2_{ik})$, $\sigma^2_{ik} \sim$ *Inv-Gamma*(ρ_k, α_k), where at the top level, $\mu_k \propto$ *constant*, and ρ_k, α_k, and τ_0 are hyperparameters. Finally, the transition probabilities between the states are given by the matrix $\boldsymbol{\tau} = (\tau_{jk})$, $(1 \leqslant j, k \leqslant K)$. Assume a Dirichlet prior for state transition probabilities, i.e. $\tau_{k1}, \cdots, \tau_{k,k-1}, \tau_{k,k+1}, \cdots, \tau_{k,K} \sim Dirichlet(\boldsymbol{\eta})$, where $\boldsymbol{\eta} = (\eta_1, \cdots, \eta_{k-1}, \eta_{k+1}, \cdots, \eta_K)$. Since the duration in a state is being modeled explicitly, no transition back to the same state can occur, i.e. there is a restriction $\tau_{kk} = 0$ for all $1 \leqslant k \leqslant K$.

6.2 Model fitting and parameter estimation

For notational simplicity, assume $\boldsymbol{Y} = \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_N\}$, is a single long sequence of length N, with r replicate observations for each $\boldsymbol{y}_i = (y_{i1}, \cdots y_{ir})'$. Let the set of all parameters be denoted by $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\tau}, \boldsymbol{\phi}, \boldsymbol{\pi}, \boldsymbol{\Sigma})$, and let $\boldsymbol{Z} = (Z_1, \cdots, Z_N)$ and $\boldsymbol{L} = (L_1, \cdots, L_N)$ be latent variables denoting the state identity and state lengths. L_i is a non-zero number denoting the state length if it is a point where a run of states ends, i.e.

$$L_i = \begin{cases} l & \text{if } Z_j = k,\ (i-l+1) \leqslant j \leqslant i\ ;\ Z_{i+1}, Z_{i-l} \neq k\ ;\ 1 \leqslant k \leqslant K \\ 0 & \text{otherwise.} \end{cases}$$

The observed data likelihood then may be written as:

$$L(\boldsymbol{\theta};\boldsymbol{Y})=\sum_{\boldsymbol{Z}}\sum_{\boldsymbol{L}}p(\boldsymbol{Y}|\boldsymbol{Z},\boldsymbol{L},\boldsymbol{\theta})P(\boldsymbol{L}|\boldsymbol{Z},\boldsymbol{\theta})P(\boldsymbol{Z}|\boldsymbol{\theta}). \tag{6.1}$$

The likelihood computation (6.1) is analytically intractable, involving a sum over all possible partitions of the sequence $\boldsymbol{Y}$ with different state conformations, and different state lengths (under the state restrictions). However, one can formulate a data augmentation algorithm which utilizes a recursive technique to efficiently sample from the posterior distributions of interest, as shown below. The key is to update the states and state length durations in an recursive manner, after calculating the required probability expressions through a *forward summation* step.

Let an indicator variable I_t take the value 1 if a segment boundary is present at position t of the sequence, meaning that a state run ends at t ($I_t=1\Leftrightarrow L_t\neq 0$). In the following, the notation $\boldsymbol{y}_{[1:t]}$ is used to denote the vector $\{y_1,y_2,\cdots,y_t\}$. Define the partial likelihood of the first t probes, with the state $Z_t=k$ ending at t after a state run length of $L_t=l$, by the "forward" probability:

$$\alpha_t(k,l)=P(Z_t=k,L_t=l,I_t=1,\boldsymbol{y}_{[1:t]}).$$

Also, let the state probability marginalized over all state lengths be given by $\beta_t(k)=\sum_{l=r_k}^{s_k}\alpha_t(k,l)$. Let $d_{(1)}=\min\{D_1,\cdots,D_K\}$ and $d_{(K)}=\max\{D_1,\cdots,D_K\}$. Then, assuming that the length spent in a state and the transition to that state are independent, i.e. $P(l,k|l',k')=P(L_t=l|Z_t=k)\tau_{k'k}=p_k(l)\tau_{k'k}$, it can be shown that

$$\alpha_t(k,l)=P(\boldsymbol{y}_{[t-l+1:t]}|Z_t=k)p_k(l)\sum_{k'\neq k}\tau_{k'k}\beta_{t-l}(k'), \tag{6.2}$$

for $2\leqslant t\leqslant N;\ 1\leqslant k\leqslant K;\ l\in\{d_{(1)},d_{(1)}+1,\cdots,\min[d_{(K)},t]\}$. The boundary conditions are: $\alpha_t(k,l)=0$ for $t<l<d_{(1)}$, and $\alpha_l(k,l)=\pi_kP(\boldsymbol{y}_{[1:l]}|Z_l=k)p_k(l)$ for $d_{(1)}\leqslant l\leqslant d_{(K)},\ k=1,\cdots,K$. $p_k(\cdot)$ denotes the k-th truncated negative binomial distribution.

The states and state duration lengths (Z_t,L_t) $(1\leqslant t\leqslant N)$ can now be updated, for current values of the parameters $\boldsymbol{\theta}=(\boldsymbol{\mu},\boldsymbol{\tau},\boldsymbol{\phi},\boldsymbol{\pi},\boldsymbol{\Sigma})$, using a *backward sampling*-based imputation step:

1. Set $i=N$. Update $Z_N|\boldsymbol{y},\boldsymbol{\theta}$ using $P(Z_N=k|\boldsymbol{y},\boldsymbol{\theta})=\frac{\beta_N(k)}{\sum_k\beta_N(k)}$.
2. Next, update $L_N|Z_N=k,\boldsymbol{y},\boldsymbol{\theta}$ using
$$P(L_N=l|Z_N=k,\boldsymbol{y},\boldsymbol{\theta})=\frac{P(L_N=l,Z_N=k|\boldsymbol{y},\boldsymbol{\theta})}{P(Z_N=k|\boldsymbol{y},\boldsymbol{\theta})}=\frac{\alpha_N(k,l)}{\beta_N(k)}.$$
3. Next, set $i=i-L_N$, and let $LS(i)=L_N$. Let $D_{(2)}$ be the second smallest value in the set $\{D_1,\cdots,D_K\}$. While $i>D_{(2)}$, repeat the following steps:
   - Draw $Z_i|\boldsymbol{y},\boldsymbol{\theta},Z_{i+LS(i)},L_{i+LS(i)}$ using
   $$P(Z_i=k|\boldsymbol{y},\boldsymbol{\theta},Z_{i+LS(i)},L_{i+LS(i)})=\frac{\beta_i(k)\tau_{kZ_{i+LS(i)}}}{\sum_k\beta_i(k)\tau_{kZ_{i+LS(i)}}},$$
   where $k\in\{1,\cdots,K\}\setminus Z_{i+LS(i)}$.

- Draw $L_i | Z_i, \boldsymbol{y}, \boldsymbol{\theta}$ using $P(L_i = l | Z_i, \boldsymbol{y}, \boldsymbol{\theta}) = \frac{\alpha_i(Z_i, l)}{\beta_i(Z_i)}$.
- Set $LS(i - L_i) = L_i,\ i = i - L_i$.

6.3 Application to a yeast data set

The HGHMM algorithm was applied to the normalized data from the longest contiguous mapped region, corresponding to about 61 Kbp (chromosomal coordinates 12921 to 73970), of yeast chromosome III [47]. The length ranges for the three states were: (1) linker: $D_1 = \{1, 2, 3, \cdots\}$, (2) delocalized nucleosome: $D_2 = \{9, \cdots, 30\}$, and (3) well-positioned nucleosome: $D_3 = \{6, 7, 8\}$.

It is of interest to examine whether nucleosome-free state predictions correlate with the location of TFBSs. Harbison et al. (2004) used genomewide location analysis (ChIP-chip) to determine occupancy of DNA-binding transcription regulators under a variety of conditions. The ChIP-chip data give locations of binding sites to only a 1Kb resolution, making further analysis necessary to determine the location of binding sites at a single nucleotide level. For the HGHMM algorithm, the probabilities of state membership for each probe were estimated from the posterior frequencies of visiting each state in M iterations (excluding burn-in). Each region was assigned to the occupancy state k, for which the estimated posterior state probability $\widehat{P}(Z_i = k | \boldsymbol{Y}) = \sum_{j=1}^{M} I(Z_i^{(j)} = k)/M$ was maximum. For all probes, this probability ranged from 0.5 to 0.9.

Two motif discovery methods SDDA [16] and BioProspector [30] were used to analyze the sequences for motif lengths of 8 to 10 and a maximum of 20 motifs per set. Motif searches were run separately on linker (L), nucleosomal (N) and delocalized nucleosomal (D) regions predicted by the HGHMM procedure. The highest specificity (proportion of regions containing motif sites corresponding to high binding propensities in the Harbison et al. (2004) data) was for the linker regions predicted by HGHMM: 61% by SDDA and 40% by BP (Table 4). Sensitivity is defined as the proportion of highly TF-bound regions found when regions were classified according to specific state predictions. The highest overall specificity and sensitivity was observed for the linker regions predicted with HGHMM, indicating nucleosome positioning information may aid significantly in motif discovery when other information is not known.

Table 4: Specificity (Spec) and Sensitivity (Sens) of motif predictions compared to data from Harbison et al

	SDDA		BP	
	Spec	Sens	Spec	Sens
Linker	0.61	0.7	0.40	0.87
Deloc Nucl	0.19	0.8	0.15	0.63
Nucleosomal	0.16	0.5	0.09	0.43

7 Conclusion

In this article we have tried to present an overview of statistical methods related to the computational discovery of transcription factor binding sites in genomic DNA sequences, ranging from the initial simple probabilistic models to more recently developed tools that attempt to use auxiliary information from experiments, evolutionary conservation, and chromatin structure for more accurate motif prediction. The field of motif discovery is a very active and rapidly expanding area, and our aim was to provide the reader a snapshot of some of the major challenges and possibilities that exist in the field, rather than give an exhaustive listing of work that has been published (which would in any case be almost an impossible task in the available space). With the advent of new genomic technologies and rapid increases in the volume, diversity, and resolution of available data, it seems that in spite of the considerable challenges that lie ahead, there is strong promise that many exciting discoveries in this field will continue to be made in the near future.

References

[1] Auger, I. E. and Lawrence, C. E. (1989). Algorithms for the optimal identification of segment neighborhoods. *Bull. Math. Biol.*, 51(1):39–54.

[2] Bailey, T. and Elkan, C. (1994). Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, 28–36.

[3] Baldi, P., Chauvin, Y., Hunkapiller, T., and McClure, M.A. (1994). Hidden Markov models of biological primary sequence information. *Proc. Natl. Acad. Sci. USA,* 91, 1059–1063.

[4] Barash, Y., Elidan, G., Friedman, N., and Kaplan, T. (2003). Modeling dependencies in protein-DNA binding sites. In *RECOMB proceedings*, 28–37.

[5] Boyer, L.A., Lee, T.I., Cole, M.F., Johnstone, S.E., Levine, S.S., Zucker, J.P., et al. (2005). Core transcriptional regulatory circuitry in human embryonic stem cells. *Cell,* 122, 947–956.

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

[7] Bussemaker, H.J., Li, H., and Siggia, E.D. (2001). Regulatory element detection using correlation with expression, *Nat. Genet.*, 27, 167–171.

[8] Chipman, H.A., George, E.I., and McCulloch, R.E. (2006). BART: Bayesian additive regression trees, *Technical Report*, Univ. of Chicago.

[9] 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.

[10] Das, D., Banerjee, N., and Zhang, M.Q. (2004). Interacting models of cooperative gene regulation, *Proc. Natl. Acad. Sci. USA*, 101, 16234–16239.

[11] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Stat. Soc. B*, 39(1):1–38.

[12] Durbin, R., Eddy, S., Krogh, A., and Mitchison, G. (1998). *Biological sequence analysis.* Cambridge University Press.

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

[14] Friedman, J.H. (1991). Multivariate adaptive regression splines, *Ann. Statist.*, 19, 1–67.

[15] Green, P. J. (1995). Reversible jump MCMC and Bayesian model determination. *Biometrika*, 82,711–732.

[16] Gupta, M. and Liu, J. S. (2003). Discovery of conserved sequence patterns using a stochastic dictionary model. *J. Am. Stat. Assoc.*, 98(461):55–66.

[17] Gupta, M. and Liu, J. S. (2005). De-novo cis-regulatory module elicitation for eukaryotic genomes. *Proc. Nat. Acad. Sci. USA*, 102(20):7079–7084.

[18] Gupta, M. (2007). Generalized hierarchical Markov models for discovery of length-constrained sequence features from genome tiling arrays. *Biometrics*, in press.

[19] Jensen, S.T., Shen, L., and Liu, J.S. (2006). Combining phylogenetic motif discovery and motif clustering to predict co-regulated genes. *Bioinformatics*, 21, 3832–3839.

[20] Keles, S. *et al.*, van der Laan, M., and Eisen, M.B. (2002). Identification of regulatory elements using a feature selection method, *Bioinformatics*, **18**, 1167–1175.

[21] Kellis, M., Patterson, N., Endrizzi, M., Birren, B., and Lander, E. (2003). Sequencing and comparison of yeast species to identify genes and regulatory elements. *Nature*, 423,241–254.

[22] Krogh, A., Brown, M., Mian, L.S., Sjöander, K., and Haussler, D. (1994). Hidden Markov models in computational biology: Applications to protein modeling. *J. Mol. Biol.,* 235, 1501–1531.

[23] Lawrence, C.E., Altschul, S.F., Boguski, M.S., Liu, J.S., Neuwald, A.F., and Wooton, J.C. (1993). Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. *Science,* 262, 208–214.

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

[25] Li, X., and Wong, W.H. (2005). Sampling motifs on phylogenetic trees. *Proc. Natl. Acad. Sci. USA,* 102, 9481–9486.

[26] Liang, F. and Wong, W. H. (2000). Evolutionary Monte Carlo: applications to c_p model sampling and change point problem. *Statistica Sinica*, 10,317–342.

[27] Liu, J.S. and Lawrence, C. E. (1999). Bayesian inference on biopolymer models, *Bioinformatics*, 15, 38–52.

[28] Liu, J. S., Neuwald, A. F., and Lawrence, C. E. (1995). Bayesian models for multiple local sequence alignment and Gibbs sampling strategies. *J. Am. Stat. Assoc.*, 90,1156–1170.

[29] Liu, J. S., Wong, W. H., and Kong, A. (1994). Covariance structure of the Gibbs sampler with applications to the comparisons of estimators and augmentation schemes. *Biometrika*, 81,27–40.

[30] Liu, X., Brutlag, D. L., and Liu, J. S. (2001). Bioprospector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. *Pacific Symposium on Biocomputing*, 127–138.

[31] Liu, Y., Liu, X.S., Wei, L., Altman, R.B., and Batzoglou, S. (2004). Eukaryotic regulatory element conservation analysis and identification using com-

parative genomics. *Genome Res.* 14, 451–458.

[32] Luger, K. (2006). Dynamic nucleosomes. *Chromosome Res,* 14, 5–16.

[33] Neuwald, A. F., Liu, J. S., and Lawrence, C. E. (1995). Gibbs Motif Sampling: detection of bacterial outer membrane protein repeats. *Protein Science*, 4,1618–1632.

[34] Moses, A.M., Chiang, D.Y., and Eisen, M.B. (2004). Phylogenetic motif detection by expectation-maximization on evolutionary mixtures. *Pac. Smp. Biocomput.,* 9, 324–335.

[35] Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE,* 77, 257–286.

[36] Sabatti, C. and Lange, K. (2002). Genomewide motif identification using a dictionary model. *IEEE Proceedings*, 90,1803–1810.

[37] Sandelin, A., Alkema, W., Engstrom, P., Wasserman, W.W., and Lenhard, B. (2004). JASPAR: an open-access database for eukaryotic transcription factor binding profiles, *Nucleic Acids Res.*, 32, D91–D94.

[38] Schneider, T.D. and Stephens, R.M. (1990). Sequence logos: a new way to display consensus sequences. *Nucleic Acids Res.,* 18, 6097–6100.

[39] Siddharthan, R., Siggia, E.D., and van Nimwegen, E. (2005). PhyloGibbs: A Gibbs sampling motif finder that incorporates phylogeny. *PLoS Comput. Biol.,* 1, e67.

[40] Sinha, S., Blanchette, M., and Tompa, M. (2004). PhyME: A probabilistic algorithm for finding motifs in sets of orthologous sequences. *BMC Bioinformatics,* 5, 170.

[41] Sinha, S. and Tompa, M. (2002). Discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic Acids Research*, 30,5549–5560.

[42] Stormo, G. D. and Hartzell, G. W. (1989). Identifying protein-binding sites from unaligned DNA fragments. *Proc. Natl. Acad. Sci. USA*, 86,1183–1187.

[43] Tanner, M. and Wong, W. H. (1987). The calculation of posterior distributions by data augmentation. *J. Am. Stat. Assoc.*, 82,528–550.

[44] Thompson, W., Palumbo, M. J., Wasserman, W. W., Liu, J. S., and Lawrence, C. E. (2004). Decoding human regulatory circuits. *Genome Research*, 10,1967–1974.

[45] Wang, T. and Stormo, G.D. (2003). Combining phylogenetic data with coregulated genes to identify regulatory motifs. *Bioinformatics,* 19, 2369–2380.

[46] Wingender, E., Chen, X., Hehl, R., Karas, H., Liebich, I., Matys, V., Meinhardt, T., Pruss, M., Reuter, I., and Schacherer, F. (2000). TRANSFAC: An integrated system for gene expression regulation. *Nucleic Acids Res.,* 28, 316–319.

[47] Yuan, G.-C., Liu, Y.-J., Dion, M. F., Slack, M. D., Wu, L. F., Altschuler, S. J. and Rando, O. J. (2005). Genome-scale identification of nucleosome positions in S. cerevisiae. *Science,* 309, 626–630.

[48] Zhao, X., Huang, H., and Speed, T. P. (2004). Finding short DNA motifs using permuted markov models. In *RECOMB proceedings*, 68–75.

[49] Zhou, Q. and Liu, J. S. (2004). Modeling within-motif dependence for transcription factor binding site predictions. *Bioinformatics*, 20(6):909–916.

[50] Zhou, Q. and Liu, J.S. (2008). Extracting sequence features to predict protein-DNA interactions: a comparative study. Nucleic Acids Research, in press.

[51] Zhou, Q. and Wong, W.H. (2004). CisModule: *De novo* discovery of cis-regulatory modules by hierarchical mixture modeling. *Proc. Natl. Acad. Sci. USA,* 101, 12114–12119.

[52] Zhou, Q. and Wong, W.H. (2007). Coupling hidden Markov models for the discovery of cis-regulatory modules in multiple species. *Ann. Appl. Statist.* to appear.