BIOINFORMATICS

Vol. 22 no. 14 2006, pages e384–e392
doi:10.1093/bioinformatics/btl251

Informative priors based on transcription factor structural class improve *de novo* motif discovery

Leelavati Narlikar[1,*], Raluca Gordân[1,*], Uwe Ohler[1,2,*] and Alexander J. Hartemink[1,2,*]

[1]Department of Computer Science, Duke University, Durham, NC 27708 and [2]Institute for Genome Sciences and Policy, Duke University, Durham, NC 27708.

ABSTRACT

Motivation: An important problem in molecular biology is to identify the locations at which a transcription factor (TF) binds to DNA, given a set of DNA sequences believed to be bound by that TF. In previous work, we showed that information in the DNA sequence of a binding site is sufficient to predict the structural class of the TF that binds it. In particular, this suggests that we can predict which locations in any DNA sequence are more likely to be bound by certain classes of TFs than others. Here, we argue that traditional methods for *de novo* motif finding can be significantly improved by adopting an informative prior probability that a TF binding site occurs at each sequence location. To demonstrate the utility of such an approach, we present PRIORITY, a powerful new *de novo* motif finding algorithm.

Results: Using data from TRANSFAC, we train three classifiers to recognize binding sites of basic leucine zipper, forkhead, and basic helix loop helix TFs. These classifiers are used to equip PRIORITY with three class-specific priors, in addition to a default prior to handle TFs of other classes. We apply PRIORITY and a number of popular motif finding programs to sets of yeast intergenic regions that are reported by ChIP-chip to be bound by particular TFs. PRIORITY identifies motifs the other methods fail to identify, and correctly predicts the structural class of the TF recognizing the identified binding sites.

Availability: Supplementary material and code can be found at http://www.cs.duke.edu/~amink/.

Contact: lee@cs.duke.edu, raluca@cs.duke.edu, uwe.ohler@duke.edu, amink@cs.duke.edu.

1 INTRODUCTION

Transcriptional regulation is governed in large part by interactions between DNA-binding proteins called transcription factors (TFs) and the corresponding sites on the DNA to which they bind. TF proteins have specific three-dimensional structures crucial for recognition of their binding sites. The binding affinity, and hence the transcription of the regulated gene, depends on both the TF's DNA-binding domain and the site it recognizes. A TF usually binds multiple sites sharing some common structure, which is typically represented using a statistical or word-based model.

An important problem in deciphering the gene regulatory code is to be able to find *de novo* binding sites for a TF given a collection of DNA sequences thought to be bound by that TF (Wasserman, 2004; Siggia, 2005). Recent advances in gene-expression arrays (Spellman *et al*., 1998; Kim *et al*., 2001, and many more), ChIP-chip experiments (Harbison *et al*., 2004; Liu *et al*., 2005), and *in vitro* DNA-binding arrays (Mukherjee *et al*., 2004) have resulted in an explosion of such data. Finding the most probable locations of binding sites hidden within the DNA sequences, and hence learning the motif best describing these binding sites, constitutes a problem of parameter estimation over an exponential search space.

Current motif finding algorithms commonly have difficulty when the motifs describing a set of binding sites are quite weak, in the sense that they are not especially over-represented relative to background. In such cases, additional information might be useful in guiding an algorithm to these weaker motifs, perhaps 'up-weighting' them relative to background so that they can be detected. This can be done using comparative genomic information, but even that information will not handle another common problem, illustrated by the following scenario. Imagine that TF_1 binds to a particular set of DNA sequences but that many of those same sequences are also bound by TF_2. If the motif of TF_2 is much stronger than that of TF_1, then the motif for TF_2 will be reported as the motif for both TFs, even if the TFs recognize and bind to DNA in quite different ways. In this paper, we present a way to overcome both of these problems.

Most eukaryotic TFs can be classified based on the structure of their DNA-binding domains. Due to the co-evolution of TFs with their binding sites, one might expect that just as TFs with a similar structure have similar DNA-binding mechanisms, there might be corresponding similarities within the DNA binding sites of TFs with similar DNA-binding mechanisms. Indeed, in a previous paper (Narlikar and Hartemink, 2006), we have shown that it is possible to predict the structural class of a TF using neither its amino acid sequence nor other protein structure information, but only the sequences of its DNA binding sites. Briefly, we built a multiclass classifier to distinguish between TFs of six different classes—Cys_2His_2 zinc fingers, Cys_4 zinc fingers, basic helix loop helix, basic leucine zippers, forkheads, and homeodomains—using only features of the sequences of their binding sites. We were able to correctly classify 87% of the TFs in a leave-one-out cross-validation procedure. Here, we build a set of binary classifiers which classify short DNA sequences as either binding sites of a particular structural class or not. We extract a large number of sequence features from these binding sites, and train a sparse Bayesian classifier based on logistic regression for this purpose. We adopt the output from three such classifiers as priors in Gibbs sampling to search for TF binding sites. The goal of these priors is for the search algorithm to be able to more rapidly and sensitively capture the ''true'' motif of the TF. This

*To whom correspondence should be addressed.

motif is expected to be based on the known binding properties of TFs sharing the same DNA-binding domain, and not just statistical over-representation relative to a background model of the sequence.

We show that our algorithm, called PRIORITY, is able to identify motifs that are not selected by popular motif finding algorithms. Along with the best motif, our algorithm outputs the most likely class to which the TF belongs. Also, when the class of the TF is known and a specific class prior can be applied by itself, we show that the resulting algorithm converges in significantly fewer iterations than when using a uniform prior. Our choice of Gibbs sampling over other search methods like expectation maximization (Dempster *et al*., 1977) is arbitrary; the concept of class-specific location priors can be applied in either context. Our choice of a position specific score matrix (PSSM), which stores the preference for each putative nucleotide at each position of the binding site (Staden, 1984), as a model for binding sites is also arbitrary; we use this model because it is widely used, and again, the concept of class-specific location priors can be incorporated with nearly any model of a TF binding site. The purpose of this paper is to show how using informative priors with respect to locations in the DNA sequences (here based on the TF structural class) improves motif discovery in general.

2 APPROACH

In this section we start with the description of the sequence model, go on to describe the generation of the class prior, and finally explain the Gibbs sampling strategy for the actual search.

2.1 Model framework

2.1.1 Sequence model Assume we have n DNA sequences $\boldsymbol{X}_1$ to $\boldsymbol{X}_n$ believed to be bound by the same TF. For simplicity, we assume that there is at most one instance of a binding site (or DNA motif) of that TF of length W hidden in each sequence (analogous to the zero or one occurrence per sequence model, or ZOOPS, in MEME (Bailey and Elkan, 1994)), though we can extend this approach to finding multiple instances of the binding site (analogous to the two component mixture model in MEME), as is implemented by Thijs *et al*. (2002). The motif follows a PSSM model while the rest of the sequence follows some pre-calculated background model $\boldsymbol{\phi}_0$. The PSSM can be described by a matrix $\boldsymbol{\phi}$ where $\phi_{a,b}$ is the probability of finding base b at location a within the binding site for $1 \leq b \leq 4$ and $1 \leq a \leq \mathrm{W}$. Let $\boldsymbol{Z}$ be a vector of size n denoting the starting location of the binding site in each sequence: $Z_i = j$ if there is a binding site starting at location j in $\boldsymbol{X}_i$ and we adopt the convention that $Z_i = 0$ if there is no binding site in $\boldsymbol{X}_i$. Thus if the sequence $\boldsymbol{X}_i$ is of length m_i and if $\boldsymbol{X}_i$ contains a binding site at location Z_i, we can compute the probability of the sequence given the model parameters as:

$$P(\boldsymbol{X}_i \mid \boldsymbol{\phi}, Z_i > 0, \boldsymbol{\phi}_0) = (X_{i,1}, X_{i,2}, \ldots, X_{i,Z_i-1} \mid \boldsymbol{\phi}_0) \times \prod_{k=Z_i}^{Z_i+W-1} \phi_{k-Z_i+1, X_{i,k}} \times P(X_{i,Z_i+W}, \ldots, X_{i,m_i} \mid \boldsymbol{\phi}_0)$$

and if it does not contain a binding site as:

$$P(\boldsymbol{X}_i \mid \boldsymbol{\phi}, Z_i = 0, \boldsymbol{\phi}_0) = P(X_{i,1}, X_{i,2} \ldots X_{i,m_i} \mid \boldsymbol{\phi}_0)$$

2.1.2 Objective function We wish to find $\boldsymbol{\phi}$ and $\mathbf{Z}$ to maximize the joint posterior distribution of all the unknowns given the data. Hence, the objective function is:

$$\arg\max_{\boldsymbol{\phi}, \boldsymbol{Z}} \; P(\boldsymbol{\phi}, \boldsymbol{Z} \mid \boldsymbol{X}, \boldsymbol{\phi}_0) \tag{1}$$

2.2 Calculation of the prior

Most motif discovery algorithms assume *a priori* that a binding site is uniformly likely to occur in all locations within each sequence. However, since we have demonstrated that certain sequences are more or less likely to be bound by various classes of TFs, we can build an informative prior to reflect such an *a priori* bias. To do so, we create three binary classifiers. The first one classifies a DNA subsequence as a binding site of a basic leucine zipper (bZip) TF or not a binding site of a bZip TF. The second distinguishes between forkhead binding sites and forkhead non-binding sites. The third distinguishes between basic helix loop helix (bHLH) binding sites and bHLH non-binding sites.

To build training sets for these classifiers, we use binding sites listed in TRANSFAC 9.4 (Wingender *et al*., 2001) that fall into one of these classes. We remove binding sites belonging to *Saccharomyces cerevisiae* from this set, since we intend to test the algorithm on yeast TFs. This leaves us with 1131 bZip, 466 forkhead, and 325 bHLH binding sites. For the training set of non-binding sites, we use a third-order Markov model from yeast intergenic regions and randomly sample subsequences of the same length distribution as the binding sites from that Markov model. We include three times as many non-binding sites as binding sites for each classifier to provide enough coverage.

For each sequence in the three training sets we construct a vector of length 1387 describing possibly relevant features of this sequence. These sequence features include:

(1) *Subsequence frequency features (1364)*: Integers representing counts of all subsequences of length 1 (i.e., each of the four nucleotides) to length 5 (i.e., each of the 4^5 possible nucleotide strings). These integers account for a total of 1364 entries in the vector, comprising the vast majority of possibly relevant features.

(2) *Ungapped palindrome features (8)*: Binary indicator variables denoting whether the sequence contains palindromic[1] subsequences of half-length 3, 4, 5, or 6 that span the entire site (i.e., end to end), as well as those that do not span the entire site (i.e., are somewhere in the middle of the site).

(3) *Gapped palindrome features (8)*: Binary indicator variables denoting whether the sequence contains gapped palindromic subsequences of half-length 3, 4, 5, or 6 that span the entire site (i.e., end to end), as well as those that do not span the entire site (i.e., are somewhere in the middle of the site). A gapped palindromic subsequence is one in which some non-palindromic nucleotides are inserted exactly in the middle of two otherwise palindromic halves.

(4) *Special features (7)*: Binary indicator variables that denote the presence or absence of features that have been identified in the literature to be over-represented in the binding sites of certain classes of TFs.

[1]Throughout, we mean palindromic in the reverse complement sense.

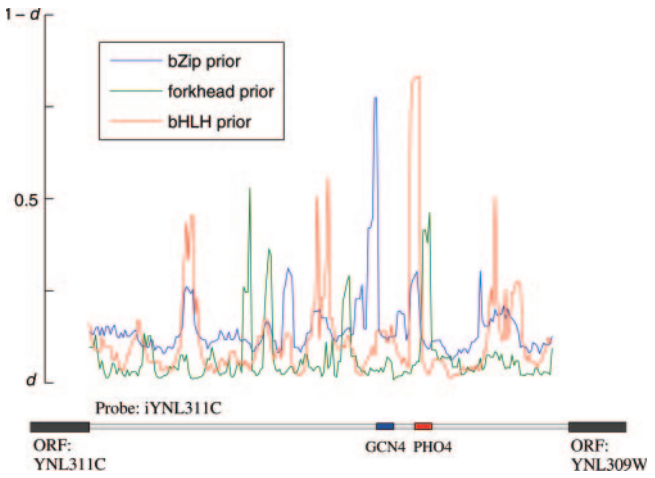

Fig. 1. Prior distributions for three classes on intergenic region iYNL311C in yeast. The *Y*-axis shows the C_{ijk} value ranging from d to $1 - d$ (see text) for each of the three classes: bZip, forkhead, and bHLH where $\boldsymbol{X}_i$ is the sequence of the probe corresponding to iYNL311C. The blue and red boxes are putative motifs for Gcn4 and Pho4, respectively, predicted by Harbison *et al*. (2004) with the criterion of a probe for an intergenic region being bound with *p*-value < 0.001. Gcn4 is a bZip protein and Pho4 is a bHLH protein. As can be seen, the probabilities at the starting locations of these motifs are higher for the respective priors.

The classifiers are learned using Bayesian sparse multinomial logistic regression (SMLR), which is designed to select a small set of features relevant for classification (Krishnapuram *et al*., 2005). The fact that features in binding sites can be used to predict the structure of the DNA-binding domain of a TF has been shown by Narlikar and Hartemink (2006) where a six-way classifier was built based on the same DNA sequence features to distinguish between TFs belonging to one of six different structural classes. We estimate the generalization accuracy using 10-fold cross-validation and achieve 89.6%, 95.2%, and 95.1% for the bZip, forkhead, and bHLH binary classifiers respectively.

Each binary classifier, being based on logistic regression, outputs the probability of the input sequence being a binding site of the respective class. Since the classifiers have a nonzero misclassification rate, instead of using the probabilities reported by the classifier directly, we linearly scale them to lie in the interval $[d, 1 - d]$, where $0 \leq d \leq 0.5$ is a tunable parameter. One can think of this transformation as a result of mixing with a uniform prior to dilute the effect of the classifier-based prior to a certain extent. Setting d to zero would be a special case in which the probabilities from the classifier are used as they are setting d to 0.5 would be a special case in which the probabilities from the classifier are ignored and a uniform prior is used instead. In all our analyses, we arbitrarily set d to 0.3.

In the general case in which r structural classes are modeled, the transformed output of the r classifiers is stored as a three dimensional vector $\boldsymbol{C}$ where C_{ijk} is the probability of the subsequence of length W starting at location j in sequence $\boldsymbol{X}_i$ being a binding site of class k and $(1 - C_{ijk})$ is the probability of it not being a binding site of that class. For C_{ij0} (the probability of the subsequence being a binding site of a TF which is not a member of the r classes for which we have built classifiers), we use a uniform probability which can be an input from the user. In all our analyses, we arbitrarily set it to 0.4.

As an illustration, Figure 1 shows the values of C_{ijk} for the classes bZip, forkhead, and bHLH ($r = 3$), where $\boldsymbol{X}_i$ is the intergenic region iYNL311C in yeast. Also shown are the putative binding sites predicted by Harbison *et al*. (2004) when they use that region as a probe. As is evident from the figure, certain positions in the sequence are *a priori* more likely to contain a binding site of a particular class than others. The idea is to have such a prior distribution over locations in each sequence in $\boldsymbol{X}$ to aid motif discovery.

We now introduce $\boldsymbol{c}$, a vector of length n, where each c_i is a hidden variable representing the class of the TF that recognizes the binding site starting at Z_i in sequence $\boldsymbol{X}_i$. Each c_i can take a value from 1 to r representing the r classes or 0 to handle the possibility that the binding site belongs to none of the r classes. This allows us to robustly find motifs of TFs with totally different DNA-binding domains from those we model. We use another parameter $\boldsymbol{\gamma}$, a vector of length $r + 1$ to define the multinomial parameters of $\boldsymbol{c}$.

Using $\boldsymbol{C}$ and $\boldsymbol{c}$, the prior probability on $\boldsymbol{Z}$ can be calculated as:

$$P(Z_i = 0 \mid c_i = k) \propto \prod_{j=1}^{m_i} (1 - C_{ijk}) \tag{2}$$

and for $u > 0$ as

$$P(Z_i = u \mid c_i = k) \propto C_{iuk} \prod_{\substack{j=1 \\ j \neq u}}^{m_i} (1 - C_{ijk}) \tag{3}$$

$P(Z_i \mid c_i)$ is normalized assuming the same proportionality constant in equations (2) and (3), so that under the assumptions of the model, we have

$$\sum_{j=0}^{m_i} P(Z_i = j \mid c_i = k) = 1 \quad \text{for } 0 \leq k \leq r$$

The inclusion of parameters $\boldsymbol{c}$ and $\boldsymbol{\gamma}$ changes the objective function in equation (1) to:

$$\arg\max_{\boldsymbol{\phi}, \boldsymbol{Z}, \boldsymbol{\gamma}, \boldsymbol{c}} P(\boldsymbol{\phi}, \boldsymbol{Z}, \boldsymbol{\gamma}, \boldsymbol{c} \mid \boldsymbol{X}, \boldsymbol{\phi}_0) \tag{4}$$

2.3 Gibbs sampling

Gibbs sampling is a Markov chain Monte Carlo (MCMC) method that approximates sampling from a joint posterior distribution by sampling iteratively from individual conditional distributions (Gelfand and Smith, 1990). Let J_v denote the distribution function of parameter v conditional on the current values of all other parameters and data. We thus need to iteratively sample v from J_v for all unknown parameters v.

Applying the collapsed Gibbs sampling strategy developed by Liu (1994) for a faster convergence, we can integrate out both the $\boldsymbol{\phi}$ and $\boldsymbol{\gamma}$ and sample only the Z_i and c_i.

The expression for sampling $\boldsymbol{Z}$ from its conditional distribution is:

$$\begin{aligned} J_Z &= P(\boldsymbol{Z} \mid \boldsymbol{c}, \boldsymbol{X}, \boldsymbol{\phi}_0) \\ &\propto P(\boldsymbol{Z}, \boldsymbol{c}, \boldsymbol{X} \mid \boldsymbol{\phi}_0) \\ &= \int_{\boldsymbol{\gamma}, \boldsymbol{\phi}} P(\boldsymbol{\phi}, \boldsymbol{Z}, \boldsymbol{\gamma}, \boldsymbol{c}, \boldsymbol{X} \mid \boldsymbol{\phi}_0) \mathrm{d}\boldsymbol{\phi} \mathrm{d}\boldsymbol{\gamma} \\ &\propto P(\boldsymbol{Z} \mid \boldsymbol{c}) \int_{\phi} P(\boldsymbol{X} \mid \boldsymbol{\phi}, \boldsymbol{Z}, \boldsymbol{\phi}_0) P(\boldsymbol{\phi}) \mathrm{d}\boldsymbol{\phi} \end{aligned} \tag{5}$$

We get the above simplification since $\boldsymbol{Z}$ is independent of $\boldsymbol{\gamma}$ conditional on $\boldsymbol{c}$. By definition, the prior on $\boldsymbol{Z}$ is also independent of $\boldsymbol{\phi}$.

Similarly, $\boldsymbol{c}$ is independent of $\boldsymbol{\phi}$ and $\boldsymbol{\phi}_0$ conditional on $\boldsymbol{Z}$. We thus get an expression for sampling $\boldsymbol{c}$ from its conditional distribution:

$$\begin{aligned} J_{\boldsymbol{c}} &= P(\boldsymbol{c} \mid \boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{\phi}_0) \\ &\propto P(\boldsymbol{Z}, \boldsymbol{c}, \boldsymbol{X} \mid \boldsymbol{\phi}_0) \\ &= \int_{\boldsymbol{\gamma}, \boldsymbol{\phi}} P(\boldsymbol{\phi}, \boldsymbol{Z}, \boldsymbol{\gamma}, \boldsymbol{c}, \boldsymbol{X} \mid \boldsymbol{\phi}_0) \mathrm{d}\boldsymbol{\phi} \mathrm{d}\boldsymbol{\gamma} \\ &= \int_{\boldsymbol{\gamma}} P(\boldsymbol{Z}, \boldsymbol{\gamma}, \boldsymbol{c}) \mathrm{d}\boldsymbol{\gamma} \int_{\boldsymbol{\phi}} P(\boldsymbol{X} \mid \boldsymbol{\phi}, \boldsymbol{Z}, \boldsymbol{\phi}_0) \, P(\boldsymbol{\phi}) \mathrm{d}\boldsymbol{\phi} \\ &\propto P(\boldsymbol{Z} \mid \boldsymbol{c}) \int_{\boldsymbol{\gamma}} P(\boldsymbol{c} \mid \boldsymbol{\gamma}) \, P(\boldsymbol{\gamma}) \mathrm{d}\boldsymbol{\gamma} \end{aligned} \tag{6}$$

Proceeding analogously to the derivation of Liu (1994), we can simplify the integrals using Dirichlet priors on both $\boldsymbol{\phi}$ and $\boldsymbol{\gamma}$. We derive the sampling distribution for Z_i, i.e J_{Z_i}, by computing $J_Z / J_{Z_{[-i]}}$ using equation (5), where $\boldsymbol{Z}_{[-i]}$ is the vector $\boldsymbol{Z}$ without Z_i. We further simplify the result by dividing it by $P(Z_i = 0, \boldsymbol{X}_i \mid c_i, \boldsymbol{\phi}_0)$ which is a constant at a particular sampling step. We thus have a sampling distribution for Z_i similar to the predictive update formula as described in Liu *et al*. (1995), but with the inclusion of the class prior:

$$J_{[Z_i = j]} = \frac{P(Z_i = j \mid c_i) \times \left(\prod_{a=1}^{W} \phi_{a, X_{i, j+a-1}} \right)}{P(Z_i = 0 \mid c_i) \times P(X_{i,j}, \ldots, X_{i, j+W-1} \mid \boldsymbol{\phi}_0)}$$

for $j > 0$, and

$$J_{[Z_i = j]} = 1$$

for $j = 0$ where $\boldsymbol{\phi}$ is calculated from the counts of the sites contributing to the current alignment $\boldsymbol{Z}_{[-i]}$ and the pseudocounts as determined by the Dirichlet prior.

Similarly, we get a sampling distribution for c_i:

$$\begin{aligned} J_{c_i} &= P(c_i \mid \mathbf{Z}, \mathbf{c}_{[-i]}) \\ &\propto P(Z_i \mid c_i = k) \times \gamma_k \quad \text{for } 0 \leq k \leq r \end{aligned}$$

where $\boldsymbol{\gamma}$ is calculated from the counts for each class from the current $\boldsymbol{c}_{[-i]}$ and the pseudocounts from the respective Dirichlet prior for $\boldsymbol{\gamma}$, where $\boldsymbol{c}_{[-i]}$ is the vector $\boldsymbol{c}$ without c_i.

We also provide the option of searching in the reverse complement of each sequence. This does not make a difference to any of the derivations. We simply concatenate the reverse complement of each $\boldsymbol{X}_i$ at the end of the original $\boldsymbol{X}_i$, and now the algorithm searches for zero or one occurrence of the motif in this longer sequence. Special care is taken to ensure that invalid locations (such as those spanning the concatenation boundary) have zero probability density during the sampling.

2.4 Scoring scheme

The joint posterior distribution function after each iteration can be calculated as:

$$\begin{aligned} P(\boldsymbol{\phi}, \boldsymbol{Z}, \boldsymbol{\gamma}, \boldsymbol{c} \mid \boldsymbol{X}, \boldsymbol{\phi}_0) \propto P(\boldsymbol{X} \mid \boldsymbol{\phi}, \boldsymbol{Z}, \boldsymbol{\phi}_0) \times P(\boldsymbol{Z} \mid \boldsymbol{c}) \\ \times P(\boldsymbol{c} \mid \boldsymbol{\gamma}) \times P(\boldsymbol{\phi}) \times P(\boldsymbol{\gamma}) \end{aligned} \tag{7}$$

To simplify the computation, we divide equation (7) by the constant probability $P(\boldsymbol{X} \mid \boldsymbol{Z} = \mathbf{0}, \boldsymbol{\phi}_0)$ and use the logarithm of the resulting function to score a motif.

In order to maximize the objective function and hence the score, we run the Gibbs sampler for a predetermined number of iterations after apparent convergence to the joint posterior, and output the highest scoring PSSM at the end.

3 RESULTS

We examined the ChIP-chip data published by Harbison *et al*. (2004) where the intergenic binding locations of TFs in yeast are profiled under various environmental conditions. We study the set of intergenic regions (or probes) that are bound with p-value < 0.001 by TFs belonging to one of the three classes for which we have built binary classifiers. There are a total of 24 TFs which qualify according to classification information in TRANSFAC, with a distribution of fourteen bZip, three forkhead, and seven bHLH proteins. We also use six more TFs whose binding sites have been well characterized in the literature, but do not fall in any of the three classes. This set is used to determine if our algorithm correctly learns motifs belonging to TFs in other structural classes for which we have not designed a specific binary classifier.

We compare the motifs found by our method to those found by Harbison *et al*. (2004). Harbison *et al*. use six different popular motif discovery programs: AlignACE (Roth *et al*., 1998), MEME (Bailey and Elkan, 1994), MDscan (Liu *et al*., 2002), a method by Kellis *et al*. (2003), a new conservation-based method by Harbison *et al*. (2004) called CONVERGE, and a modified MEME which was fed conservation information across *sensu stricto Saccharomyces* species. In the main text of this paper we consider only the three programs which do not use conservation information, namely AlignACE, MEME, and MDscan; the supplementary material contains a comparison with all six programs for the TFs considered in this paper, and profiled in all reported environmental conditions. Harbison *et al*. (2004) also do a post-processing step of clustering results from all these programs using cutoffs for significance by various criteria to reach a single motif (if it meets their significance criteria, none otherwise) per TF. Here we compare our results with the raw output from each of the three programs as well as the post-processed single motif derived from all six programs. Thus, our method is competing with six state-of-the-art motif finding algorithms, and also their combination.

There are various differences in the inherent properties of these programs as well as the way in which they are run. AlignACE is based on Gibbs sampling, but uses only single nucleotide frequency to model the background. It was run with the default settings ten times. MEME was run with a fifth order Markov background model using the ZOOPS option and allowed to look for motifs of width 7 to 18 nucleotides. MDscan was also run repeatedly, once with each width in the range 8 to 15 nucleotides.

3.1 Performance of PRIORITY

We set the Dirichlet prior parameters for $\boldsymbol{\phi}$ to 0.5 for all four bases. We gave 3 pseudocounts to γ_k when k is the class of the TF and 1 otherwise. We searched for motifs in the reverse complement of each sequence just as all other programs used for comparison do. With these parameter settings, we applied PRIORITY on each probeset

corresponding to all the 30 TFs profiled under various environmental conditions. Our algorithm was applied for a fixed window size of length 8, so in general it was at a disadvantage with respect to the other programs where the width is varied. We restarted our program 10 times to prevent local optima and report the motif with the highest score.

Table 1 illustrates the results for TFs under the environmental condition considered by Harbison *et al*. (2004) in reporting their final motif. For TFs where they do not report a final motif, we use the probeset resulting from the environmental condition that produces the largest number of bound sequences.

We believe, as is also argued by Liu *et al*. (2002), that a motif finding algorithm should be evaluated based on whether its top motif is correct or not. Each algorithm can use whatever method or score it chooses to rank the motifs and report a top motif. Thus in Table 1, we list the top motif from each of the four algorithms: AlignACE, MEME, MDscan, and PRIORITY according to their respective scoring systems. We also list the final motif reported by Harbison *et al*., but it is important to note that this final motif is produced after considerable human and computational efforts. The post-processing steps include testing multiple motifs from each of the six programs for significance by AUC scores as well as enrichment scores, and then clustering them to produce one motif.

Looking at the table, it is clear that the top motifs from AlignACE rarely match the true motifs from the literature. We believe this happens because AlignACE uses such a simple model to capture features in the background sequence. It has been shown previously that having a higher order Markov model to model the background sequence helps in motif discovery (Liu *et al*., 2001; Thijs *et al*., 2001). The other programs are not disadvantaged by a simple background model as is AlignACE, but in all cases, are outperformed by PRIORITY, as discussed in the remainder of this section.

For more clarity, we categorize the TFs listed in Table 1 into three groups:

- Group I: Literature consensus motif exists, and PRIORITY fails to find such a motif.
- Group II: Literature consensus motif exists and PRIORITY succeeds in finding such a motif.
- Group III: No literature consensus motif exists.

We now discuss TFs falling into these groups in detail.

Group I: This group includes only four TFs: Arr1, Yap3, Yap5, and Yap6. These are all bZip proteins and members of the Yap family (Arr1 is also called Yap8). No program finds motifs matching the literature for any of these four. Thus when PRIORITY fails, the other programs also fail. However, in the case of Arr1, Yap5, and Yap6, PRIORITY predicts a class other than bZip. This is a clue to the fact that the motif the algorithm converges to in these cases may not be a true motif of the TF that was profiled. While we still consider these three cases as failures of our algorithm, at least the algorithm provides some diagnostic information.

Group II: This group includes a total of 20 TFs: Cad1, Cin5, Gcn4, Hac1, Sko1, Yap1, Yap7, Fkh1, Fkh2, Cbf1, Ino2, Ino4, Pho4, Tye7, Leu3, Nrg1, Rap1, Reb1, Ste12, and Ume6. Among the 20 motifs correctly identified by our program, AlignACE finds 2, MEME finds 13, and MDscan finds 17. None of the three other programs finds the true motif for bZip Sko1. While MDscan finds the true motif for Hac1, it does not appear as the post-processed final motif reported by Harbison *et al*.

Along with the correct motif, PRIORITY consistently predicts the true class for TFs in the three classes (100% accuracy). It also correctly assigns the ''other'' class to five of the six TFs not belonging to the three classes explicitly modeled; although PRIORITY learns the true motif of Ste12, it assigns the wrong class. We believe this case is an instance of the algorithm getting stuck in a local maximum or a misclassification by the forkhead binary classifier.

Judging by the performance of PRIORITY on these TFs, we see that despite the computationally expensive steps of Harbison *et al*. in calculating the final motif, our program directly reports better results than the post-processed combination of all six programs.

Group III: Here we consider the remaining six TFs (Cst6, Met28, Met4, Fhl1, Phd1, Sok2) for which there is no known consensus in the literature. For the bZips Cst6 and Met28, without experimental verification, there is no way of knowing for sure if the motifs found by our method are indeed true.

For Met4, Harbison *et al*. find a motif using their algorithm CONVERGE (which exploits cross-species sequence conservation information). This long motif is present in only eight of the 37 bound probes, hence it is no surprise that programs that do not use conservation information are not able to find it. However, we do not know if it is a true motif; in fact, in the literature search that we conducted, we did not find any evidence of Met4 binding DNA directly. Our algorithm finds a different motif for this set of bound intergenic regions which is present in 29 of the 37 sequences and assigns it a bHLH class. This leads us to conclude that this motif could belong to a bHLH protein which is either a cofactor (binds to the same set of sequences separately) or forms a complex with Met4 and binds DNA. Subsequent literature search proves the latter to be true: Met4 forms a complex with Cbf1 and Met28, and it is Cbf1 (a bHLH class protein) which makes contact with DNA at TCACGTG (Kuras *et al*., 1997). PRIORITY does not find the same motif for Met28. In addition to being part of this complex, Met28 is part of other complexes which bind DNA (Blaiseau and Thomas, 1998) and is also capable of binding DNA by itself with low affinity (Kuras *et al*., 1997). We believe these different binding modes dilute the binding site signal.

For forkhead Fhl1, all programs find the same motif (see reverse complement for MEME). This motif is an exact match to the Rap1 binding site. Rap1 does not fall into any of the three classes, and PRIORITY diagnoses this by reporting the class associated with the motif to be ''other'', suggesting that the motif is most likely not a motif for Fhl1. More than half of the probes bound by Rap1 appear in the set bound by Fhl1. Indeed, these TFs are known to be cofactors for some ribosomal protein genes and bind cooperatively (Schawalder *et al*., 2004). We could not find any definitive evidence in the literature either of Fhl1 binding DNA directly, or via a complex with Rap1 or some other TF. However, if Fhl1 does bind DNA directly, and the motif learned is its true motif, one would expect to find multiple copies of the motif (since both Rap1 and Fhl1 need a site on the same probe to which to bind). Harbison *et al*. attempted to determine which TFs tended to use repetitive motifs, but Rap1 does not seem to fall into this category (nor does Fhl1). This makes us believe that the motif learned is bound exclusively by Rap1.

Table 1. Motif comparison for 30 TFs with four different programs. Table shows the motifs learned by various algorithms used by Harbison *et al*. and those learned by our algorithm. For comparison, we use the motifs with the top MAP score for AlignACE, MEME, and MDscan, as well as the final motif reported by Harbison *et al*. after clustering results from these three and three other motif finding programs which use conservation information. In the fifth column we report the top motif according to our score. We also report the predicted class and the percentage of entries in ***c*** contributing to that class. The last column is the literature consensus as used by Harbison *et al*. collected from YPD, SCPD, and TRANSFAC databases at the time their paper was published. The bold sections in the motifs indicate either a match with the literature consensus in the final column or to a motif we found in the literature search we conducted. In cases where the match is not obvious, it is probably because the reverse complement of the sequence matches the literature consensus. Lower case letters in the motifs indicate a weaker preference (less information content at that position). Ambiguity codes: S=C/G, W=A/T, R=A/G, Y=C/T, M=A/C, K=G/T, and '.'= A/C/G/T.

TF	Harbison *et al.*				PRIORITY		Literature
	AlignACE Top MAP	MEME Top MAP	MDscan Top MAP	Post-processed motif from all six programs	Top MAP	Predicted class	
Basic Leucine Zipper TFs							
Arr1	R.AmA.a.A.A.AmA.A	cAmAcACmcAmAmayrcA	CACACACAC	—	YAAACAca	fork 78%	TTACTAA
Cad1	GtGTGTGkGTGTG	GC**TKACTAA**T.	GKGTGTGK	m**TTAsTmA**kC	GC**TTACTA**	bZip 73%	TTACTAA
Cin5	AARAAAA.AA.A	TTTYyttytTy.ytyYYK	.GSGssgG	**TTAygTAA**	**TTAyGTAA**	bZip 94%	TTAC[r]TAA*
Cst6	A.A..rAAA.A..A..a.A	rmAtk.mAwrCRAAaA	AgTY.AsT	—	.ACTGGAC	bZip 80%	—
Gcn4	rAAAAARAAa	yTyTyyTyyYTyTTTc	**TGAsTCAt**	**TGAsTCa**	**ATGACTCA**	bZip 96%	ArTGACTCw
Hac1	A..rAA..MAAARA	TrCSTSkccwywtmM	**TAcGTGkC**	—	**ACGTGTCA**	bZip 76%	kGmCA[G]CGTGTC**
Met28	a.A.A.....AAAA.AAA	TkyTTTTkskssskcTTw	ATrTAyAT	—	SKAAACYG	bZip 75%	—
Met4	AA.AR.RARAAA	AArAAMmmRmAA	TATATATAT	RMmAwsTGKSgyGsc	TGTCACGt	bHLH 80%	—
Sko1	mAAA.RARr.AA	TkTTkyyykTTTkyKKCk	sSgtacSs	—	.t**ACGTCA**	bZip 72%	ACGTCA
Yap1	AwMArrAAr..A	ssTTTyCrT	**TTA.TAA**k	**TTaGTmA**Gc	**TKACTAA**w	bZip 87%	TTAsTAA
Yap3	A.A...A.A.A.A.A.A.Amr	T.kyttcTT.mTTkTT	CACACACAC	—	.CTAAaTS	bZip 65%	TTACTAA
Yap5	GTGTG.GTGTG	GTCGAgsAgAAcsAgGAt	CACACmCAC	—	CGTGKGYG	bHLH 93%	TTACTAA
Yap6	RArAArAAAA	magAAA.rrrAArArR	smYGCAks	—	..CGTGG.	bHLH 91%	TTACTAA
Yap7	AAAARRAAAR	m**TTAsTmA**kc..	**TKAsTMA**k	m**TkAsTmA**k	**TkASTAA**K	bZip 82%	TTACTAA
Forkhead TFs							
Fhl1	RTGTayGGrTG	.t.taCAyCCrTACAYyy	TGyryGGr	rTGTayGGrtg	TGTAYGGR	other 94%	—
Fkh1	RAR.ARA.RAA.A	aaa**.rtAAACAa**..r.a	**tTGTTTAC**	**tTgTTTac**	**GTAAACAA**	fork 95%	GGTAAACAA
Fkh2	rRaAAR.AAA.R	AArrr.rAAAaa.r.AAA	**.GtAAACAA**	aaa**.GTAAACAa**	**GTAAACAA**	fork 94%	GGTAAACAA
Basic Helix Loop Helix TFs							
Cbf1	rAAAAARAAR	**RTCACGTGm**	**kCACGTGm**	**tCACGTG**	**rTCACGTG**	bHLH 97%	rTCACrTGA
Ino2	rARARARR.AA	s**CAYsTGMw**.a	k**CAsrTG**c	**CAcaTG**c	**TCACAT**GC	bHLH 86%	ATTTCACATC
Ino4	A...A.AARrArAR	t**TTYCACATG**s	**CAygTGma**	**CATGTGaaa**a	**CATGTGAA**	bHLH 85%	CATGTGAAAT
Phd1	rRAAARrRAA	rAaA.grAaA.RrRaA	.SSSsSSS	sc.GC.gg	kCGTGsc.	bHLH 95%	—
Pho4	AAAAArAAA..A	**sCACGTg**	**sCACGTGs**	**CACGTGs**	**CACGTG**cs	bHLH 81%	wcacgtk.g
Sok2	ARAArRAAA.R	ArrM..AAAmr.RrAA	SSss.sSG	tGCAg..a	.sCGTG..	bHLH 93%	—
Tye7	rRAARAraAAAYs	rY**CACsTG**AYg	T**CACGTG**A	t**CACGTG**Ay	**CAsGTG**AT	bHLH 92%	CA..TG
TFs belonging to structural classes other than the three modeled explicitly							
Leu3	aA.AAAaAAA...A	**gCCsGtacCGS**wc	**CGgtacCG**	**cCGgtacCGG**	**GgtACyGG**	other 80%	yGCCGGTACCGGyk
Nrg1	rAAAAArAAR	srAarmSrAAA	gGA**CCCT**k	GGa**CCCT**	.GGA**CCCT**	other 89%	CCCT
Rap1	**rTGYayGGrTg**	**..grTGyayGGrTGyr**	**yCCRtrCM**	**tGyayGGrtg**	**ACCCRTAC**	other 90%	wrmACCCATACAyy
Reb1	ksCGGGTAAy	...k**sCGGGTAA**y.	r**TTACCCG**	**CGGGTAA**	m**TTACCCG**	other 93%	TTACCCGG
Ste12	AAAArRAAA..R	gAaACa..t.**TgAaACa**	**tGTTTCA**.	**tgAAAC**	tt**TGAAAC**	fork 95%	ATGAAAC
Ume6	**TsGGCGGCT**A	ww**TAGCCGCCsA**.s	T**AGCCGCC**	t**aGCCGCCs**w	**AGCCGCCs**	other 86%	wGCCGCCGw

* The motif with the inserted r was experimentally confirmed by Harbison *et al*. after they conducted a gel-shift assay to verify the authenticity of the motif they obtained by their *in silico* analysis for Cin5.

** Harbison *et al*. report the longer motif with the central G as literature consensus, but in a literature search we conducted, we found that a new binding site TYACGTGTYM without the central G has been experimentally confirmed by Patil *et al*. (2004) using gel-retardation assays.

For the two bHLH TFs Phd1 and Sok2, the final motifs reported by Harbison *et al*. are both matches to the zinc-coordinating Sut1 TF which does not belong to any of the three classes we studied. Looking at the bound probes, Harbison *et al*. conclude that both pairs Sut1/Phd1 and Sut1/Sok2 are highly co-occurring regulator pairs. This, we believe is a case similar to that of Fhl1, where a strong motif of a different co-occurring TF is learned by regular motif discovery algorithms. The difference is that our algorithm does not find the strong Sut1 motif like it finds Rap1 for Fhl1. Instead, it finds motifs of the bHLH class for both TFs. We thus think these motifs could be true motifs of the two bHLH TFs.

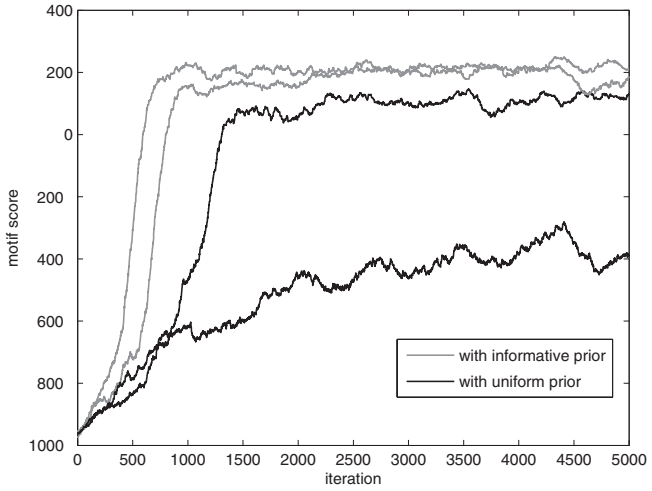

Fig. 2. Motif scores for two Gibbs samplers searching for a Gcn4 motif, one with and the other without the informative prior, over 5000 iterations. Both programs were run five times from different starting locations. The two black plots are the best and worst runs for the program with the uniform prior. The two grey plots are the best and worst runs for the program with the informative prior. Although the absolute values of the scores are not comparable (due to an arbitrary constant value assigned to the uniform prior), it is clear that the number of iterations taken to converge for the algorithm with the informative prior is almost half. Also, each of the five runs converges to a similar final motif in the case of the program incorporating the informative prior. On the other hand, during the worst of the five runs for the program with the uniform prior, the sampler gets stuck in a local maximum that corresponds to a suboptimal motif.

Partitioning the TFs in this manner enables us to draw some important conclusions about the performance of PRIORITY. Simply looking at the results of Group I and Group II, we see that our algorithm finds the correct motif whenever at least one of the other programs finds it and sometimes when none do. From results on TFs in Group III, we see that our program learns motifs of co-occurring TFs and predicts the true class of the co-occurring TF. When the class of the co-occurring TF is different from the profiled TF, our program may help to diagnose the existence of this co-occurring TF.

3.2 Performance of single-class PRIORITY

Sometimes, we know in advance the structural class of the TF which is binding a set of DNA sequences. In such a case, we can fix the class parameter $\boldsymbol{c}$ in advance and not sample from it. We applied this single-class version of PRIORITY on the same ChIP-chip data by setting the class parameter to the respective class of the TF.

Here we do not list the results obtained by using the ''true'' class prior on each of the 30 TFs. The final motifs are not very different, but we notice a big difference in the running times of the sampler when using a single-class informative prior versus using a uniform prior (as is done in most programs). As just one example, we concentrate on Gcn4, a bZip protein, which seems to have a strong motif. Our version of the simple Gibbs sampler with a uniform prior (which is similar to AlignACE with a higher order background model) also finds it.

Figure 2 is a graph of the score of the sampled motif at each iteration (explained in Section 2.4) versus the number of iterations. We ran the sampler with and without the informative prior five times for 5000 iterations and recorded the score of the motif at the end of each iteration. The final motif at the end of each run is simply the motif that scored the best at some point during the run. We have shown the best and the worst scoring runs with and without the informative prior. Although both methods have respective maximum scores at the same values of $\mathbf{Z}$, the sampler with the informative prior converges much sooner than the one with the uniform prior. In fact, in one of the runs, the sampler with the uniform prior gets stuck in a local maximum and remains stuck for all 5000 iterations. With the single-class informative prior, the sampler is less likely to suffer this fate.

4 DISCUSSION

We demonstrate the benefits of using class-specific priors in *de novo* motif discovery problems. More generally, we show how the presence of an informative prior over sequence locations makes it possible to learn the correct motif where conventional methods that use a uniform prior fail.

A novel feature of our method is its ability to output the probable class of the TF binding the motif along with the motif. This gives users more confidence in the learned motif being a description of ''true'' binding sites in cases where the structural class of TF is known. In cases where the TF is not known, the predicted class can be used to limit the possible TFs to be further investigated. For instance, in the case of searching for binding sites in the upstream regions of a set of coexpressed genes, an indication of the class may provide a clue as to which TF could be regulating the set.

In cases where a strong motif of a different TF exists in the same probeset (e.g., Met4, Fhl1), PRIORITY correctly finds this strong motif. In addition, by predicting the class of this motif as the true class which is different from the class of the profiled TF, the program is able to diagnose the presence of the co-occurring TF.

Throughout the paper, we have used PSSMs to model motifs. The PSSM model inherently assumes two things: 1) the binding sites recognized by a particular TF are of fixed length, and 2) position-specific nucleotide preferences exhibit independence between positions. However, experimental and computational studies over the past few years have shown that positions within binding sites are not always independent. Bulyk *et al*. (2002) showed experimentally that for the zinc finger Zif268, there is significant interdependence between the nucleotides of its binding sites. To have a more flexible model for binding sites, Agarwal and Bafna (1998) proposed using Bayesian networks. Since learning general Bayesian networks is an NP-hard problem (Chickering, 1995), Agarwal and Bafna (1998) relaxed their model to trees, and Barash *et al*. (2003) extended this to mixtures of trees and mixtures of PSSMs. Their work showed that these more expressive models indeed yielded better likelihood scores. However, incorporation of a more expressive model into the *de novo* motif finding problem makes the search more complex when no additional information is used. In such cases, when learning a more complex model, an informative prior will prove even more useful in focusing the search significantly.

Our method assigns a prior on the locations within each sequence X_i and not on any specific form of the motif model. Thus in principle, we can incorporate our prior into any general motif finding algorithm and any motif model. Adding a prior on the motif model is orthogonal to our methodology, and can be used when required.

We are the first to propose an informative prior over sequence locations, but others have used structural information to add a prior over motif models (in each case, a PSSM). Sandelin and Wasserman (2004) use JASPAR (Sandelin *et al*., 2004) PSSMs to build a single familial binding profile for each TF family and use that as a prior over PSSMs. However, their work is on narrower domain classes, each not containing more than 10 members. Also, they need to know what family the TF belongs to beforehand. Macisaac *et al*. (2006) extend this concept of DNA-binding profiles to include more families and more variations within families. They generate hypotheses from the profiles and test each one on ChIP-chip data in a classifier-based approach. Xing and Karp (2004) propose a new Bayesian model to capture structural properties typical of particular families of motifs. They learn expressive profiles from PSSMs specific to different classes of TFs. They have results only on simulated data and unfortunately we could not find the code for comparison. Slightly different, but based on the same idea of using prior knowledge related to PSSM models is the SOMBRERO algorithm by Mahony *et al*. (2005). They cluster known PSSMs using self organizing maps (SOMs) and use these clusters as prior knowledge for their search. All these approaches generate a prior over PSSMs and thus apply it on PSSMs directly. Sandelin and Wasserman use pseudocounts to initialize the PSSM they intend to learn, Macisaac *et al*. use their profiles as priors on PSSMs during EM, Xing and Karp use the parameters learned from their profile model as a prior on PSSMs, and Mahony *et al*. use clusters learned from known PSSMs as a starting point for their SOM algorithm which has PSSMs as nodes. Thus these methods can be used only if the motif model to be learned is a matrix based model like a PSSM.

Since we include various features from raw binding sites in our classifiers, we believe we are able to capture inter-position dependencies and structures like palindromes where these other methods cannot. Also, since Sandelin and Wasserman (2004) and Xing and Karp (2004) consider only PSSMs, they lose information about binding sites which were not used to form the PSSM, either because they were of a different size or they just did not contribute to a high scoring PSSM.

Kaplan *et al*. (2005) devise a structure-based approach to predict binding sites from the Cys_2His_2 zinc finger protein family. Their approach is the reverse of ours in the sense that they predict DNA-binding preferences from the zinc finger residue information of the TF and then scan the genome for putative binding sites with those preferences. It is not possible for us to compare our results with theirs due to the difference in the classes under consideration.

Thus far, we have considered only three classes of TFs in yeast. We are in the process of expanding our work to include other big classes like Cys_2His_2, homeodomains, etc. The problem with increasing the number of classes is not only with finding a good binary classifier for each new class, but also the increased computational time required for the Gibbs sampler to converge to sampling from the posterior and visit good optima. For up to two classes, the computational time is fine. In fact, as described in Section 3.2, the sampler reaches its maximum faster with a single-class informative prior than with a uniform prior. However for more than two class-specific priors, we notice the sampler begins to get stuck in local maxima more often. Multiple restarts solves the problem for three classes (the results of which are described in this paper) but it is open at this point how well this will scale to an even larger number of classes. There is a huge body of literature on convergence in Gibbs samplers and other MCMC methods, and we are in the process of exploring other search techniques which may yield faster convergence.

One current disadvantage of our method and all the methods considered by Harbison *et al*. is that none of them provide a significance score to the discovered motif. As a result, the user is left having to calculate various significance scores after the fact based on enrichment, AUC scores, or some other metric as Harbison *et al*. do in their paper. Having multiple priors with different distribution values makes it more tricky. In the case of the single-class version of PRIORITY, a *p*-value can be calculated using random sequence sets of similar length distribution (see supplementary material).

The goal of this study is to demonstrate the significant benefits of informative priors over sequence locations; we have not yet incorporated additional features like learning the optimal width of the motif, searching for multiple copies, etc. We note, however, that these features are useful and will only further improve the performance of the algorithm.

In closing, we believe that using algorithms based only on statistical over-representation will fall short when searching for motifs in more complex organisms having genomes with large intergenic regions. Using informative priors over sequence locations—constructed on the basis of conservation among species (Kellis *et al*., 2003), class-specific DNA binding preferences as presented here, or information like nucleosome occupancy (Lee *et al*., 2004)—will benefit motif finding algorithms as they are applied to more complex organisms.

ACKNOWLEDGMENTS

The authors would like to gratefully acknowledge that the research presented here was supported in part by an Alfred P. Sloan Fellowship to U.O., and by a National Science Foundation CAREER award and an Alfred P. Sloan Fellowship to A.J.H.

REFERENCES

Agarwal,P. and Bafna,V. (1998) Detecting non-adjacent correlations within signals in DNA, *RECOMB '98*

Bailey,T. and Elkan,C. (1994) Fitting a mixture model by expectation maximization to discover motifs in biopolymers, *ISMB '94*, AAAI Press, Menlo Park, California, pp. 28–36.

Barash,Y., Elidan,G., Friedman,N., and Kaplan,T. (2003) Modeling dependencies in protein-DNA binding sites, *RECOMB '03*.

Blaiseau,P. and Thomas,D. (1998) Multiple transcriptional activation complexes tether the yeast activator Met4 to DNA, *The EMBO Journal*, 17:6327–6336.

Bulyk,M., Johnson,P., and Church,G. (2002) Nucleotides of transcription factor binding sites exert interdependent effects on the binding affinities of transcription factors, *Nucleic Acids Research*, 30:1255–1261.

Chickering,D. (1995) Learning Bayesian networks is NP complete, In Fisher,D. and Lenz,H., eds., *Learning from Data: Artificial Intelligence and Statistics V*, pp. 121–130.

Dempster,A., Laird,N., and Rubin,D. (1977) Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society B*, 39:1–38.

Gelfand,A. and Smith,A. (1990) Sampling based approaches to calculating marginal densities, *Journal of the American Statistical Association*, 85:398–409.

Harbison,C., Gordon,D., Lee,T., Rinaldi,N., Macisaac,K., Danford,T., Hannett,N., Tagne,J., Reynolds,D., Yoo,J., Jennings,E., Zeitlinger,J., Pokholok,D., Kellis,M., Rolfe,P., Takusagawa,K., Lander,E., Gifford,D., Fraenkel,E., Young,R. (2004) Transcriptional regulatory code of a eukaryotic genome, *Nature*, 431:99–104.

Kaplan,T., Friedman,N., and Margalit,H. (2005) Ab initio prediction of transcription factor targets using structural knowledge, *PLoS Computational Biology*, 1(1):e1.

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*, 432:241–254.

Kim,S., Lund,J., Kiraly,M., Duke,K., Jiang,M., Stuart,J., Eizinger,A., Wylie,B., and Davidson,G. (2001) A gene expression map for *Caenorhabditis elegans*, *Science*, 293:2087–2092.

Krishnapuram,B., Figueiredo,M., Carin,L., and Hartemink,A. (2005) Sparse multinomial logistic regression: Fast algorithms and generalization bounds, *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 27:957–968.

Kuras,L., Barbey,R., and Thomas,D. (1997) Assembly of a bZIP-bHLH transcription activation complex: Formation of the yeast Cbf1-Met4-Met28 complex is regulated through Met28 stimulation of Cbf1 DNA binding, *The EMBO Journal*, 16(9):2441–51.

Lee,C., Shibata,Y., Rao,B., Strahl,B., Lieb,J. (2004) Evidence for nucleosome depletion at active regulatory regions genome-wide, *Nature Genetics*, 36(8):900–905.

Liu,J. (1994) The collapsed Gibbs sampler with applications to a gene regulation problem, *Journal of the American Statistical Association*, 89:958–966.

Liu,J., Neuwald,A., and Lawrence,C. (1995) Bayesian models for multiple local sequence alignment and Gibbs sampling strategies, *Journal of the American Statistical Association*, 90:1156–1170.

Liu,X., Brutlag,D., and Liu,J. (2001) BioProspector: Discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes, *Pacific Symposium on Biocomputing '01*, World Scientific, New Jersey, pp. 127–138.

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

Liu,X., Noll,D., Lieb,J., and Clarke,N. (2005) DIP-chip: Rapid and accurate determination of DNA binding specificity, *Genome Research*, 15(3):421–427.

Macisaac,K., Gordon,D., Nekludova,L., Odom,D., Schreiber,J., Gifford,D., Young,R., Fraenkel,E. (2006) A hypothesis-based approach for identifying the binding specificity of regulatory proteins from chromatin immunoprecipitation data, *Bioinformatics*, 22:423-429.

Mahony,S., Golden,A., Smith,T., and Benos,P. (2005) Improved detection of DNA motifs using a self-organized clustering of familial binding profiles, *Bioinformatics*, 21 (Supp 1):i283–i291.

Mukherjee,S., Berger,M., Jona,G., Wang,X., Muzzey,D., Snyder,M., Young,R., and Bulyk,M. (2004) Rapid analysis of the DNA binding specificities of transcription factors with DNA microarrays, *Nature Genetics*, 36(12):1331–1339.

Narlikar,L. and Hartemink,A. (2006) Sequence features of DNA binding sites reveal structural class of associated transcription factor, *Bioinformatics*, 22:157–163.

Patil,C., Li,H., and Walter,P. (2004) Gcn4p and novel upstream activating sequences regulate targets of the unfolded protein response, *PLoS Biology*, 2(8):E246.

Roth,F., Hughes,J., Estep,P., and Church,G. (1998) Finding DNA regulatory motifs within unaligned non-coding sequences clustered by whole-genome mRNA quantitation, *Nature Biotechnology*, 16:939–945.

Sandelin,A., Alkema,W., Engström,P., Wasserman,W., and Lenhard,B. (2004) JASPAR: An open access database for eukaryotic transcription factor binding profiles, *Nucleic Acids Research*, 32(1) Datebase Issue.

Sandelin,A. and Wasserman,W. (2004) Constrained binding site diversity within families of transcription factors enhances pattern discovery bioinformatics, *Journal of Molecular Biology*, 338(2):207–215.

Schawalder,S., Kabani,M., Howald,I., Choudhury,U., Werner,M., and Shore,D. (2004) Growth-regulated recruitment of the essential yeast ribosomal protein gene activator Ifh1, *Nature*, 432:1958–1061.

Siggia,E. (2005) Computational methods for transcriptional regulation, *Current Opinion in Genetics and Development*, 15:214–221.

Spellman,P., Sherlock,G., Zhang,M., Iyer,V., Anders,K., Eisen,M., Brown,P., Botstein,D., and Futcher,B. (1998) Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization, *Molecular Biology of the Cell*, 9:3273–3297.

Staden,R. (1984) Computer methods to locate signals in nucleic acid sequences, *Nucleic Acids Research*, 12:505–519.

Thijs,G., Lescot,M., Marchal,K., Rombauts,S., De Moor,B., Rouze,P., and Moreau,Y. (2002) A higher-order background model improves the detection of potential promoter regulatory elements by Gibbs sampling, *Bioinformatics*, 17:1113–1122.

Thijs,G., Marchal,K., Lescot,M., Rombauts,S., De Moor,B., Rouze,P., and Moreau,Y. (2002) A Gibbs sampling method to detect over-represented motifs in the upstream regions of coexpressed genes, *Journal of Computational Biology*, 9:447–464.

Wasserman,W. and Sandelin,A. (2004) Applied bioinformatics for the identification of regulatory elements, *Nature Reviews Genetics*, 5(4):276–287.

Wingender,E., Chen,X., Fricke,E., Geffers,R., Hehl,R., Liebich,I., Krull,M., Matys,V., Michael,H., Ohnhauser,R., Pruss,M., Schacherer,F., Thiele,S., Urbach,S. (2001) The TRANSFAC system on gene expression regulation, *Nucleic Acids Research*, 29:281–283.

Xing,E. and Karp,R. (2004) MotifPrototyper: A Bayesian profile model for motif families, *Proc. Natl. Acad. Sci.*,101:10523–10528.