A Simple Hyper-Geometric Approach for Discovering Putative Transcription Factor Binding Sites

Yoseph Barash, Gill Bejerano, and Nir Friedman

School of Computer Science & Engineering,
The Hebrew University, Jerusalem 91904, Israel
{hoan,jill,nir}@cs.huji.ac.il

Abstract. A central issue in molecular biology is understanding the regulatory mechanisms that control gene expression. The recent flood of genomic and post-genomic data opens the way for computational methods elucidating the key components that play a role in these mechanisms. One important consequence is the ability to recognize groups of genes that are co-expressed using microarray expression data. We then wish to identify *in-silico* putative transcription factor binding sites in the promoter regions of these gene, that might explain the co-regulation, and hint at possible regulators. In this paper we describe a simple and fast, yet powerful, two stages approach to this task. Using a rigorous hypergeometric statistical analysis and a straightforward computational procedure we find small conserved sequence kernels. These are then stochastically expanded into PSSMs using an EM-like procedure. We demonstrate the utility and speed of our methods by applying them to several data sets from recent literature. We also compare these results with those of MEME when run on the same sets.

1 Introduction

A central issue in molecular biology is understanding the regulatory mechanisms that control gene expression. The recent flood of genomic and post-genomic data, such as microarray expression measurements, opens the way for computational methods elucidating the key components that play a role in these mechanisms.

Much of the specificity in transcription regulation is achieved by *transcription factors*, which are largely responsible for the so called combinatorial aspects of the regulatory process (the number of possible behaviors being much larger than the number of factors). These are proteins that, when in the suitable state, can bind to specific DNA sequences. By binding to the chromosome in a location near the gene, these factors can either activate or repress the transcription of the gene. While there are many potential sites where these factors can bind, it is clear that much of the regulation occurs by factors that bind in the *promoter region* which is located upstream of the transcription start site.

Unlike DNA-DNA hybridization, the dynamics of protein-DNA recognition are not completely understood. Nonetheless, experimental results show that transcription factors have specific preference to particular DNA sequences. Somewhat generalizing, the affinity of most factors is determined to a large extent by one or more relatively short regions of 6-10bp. (One must bear in mind that DNA strands span a complete turn every 10 bases, thus geometric considerations make it unlikely that a single protein binds to a longer region, although counterexamples are known.) A common situation is the formation of *dimers* in which two DNA binding proteins form a complex. Each of the two proteins, binds to a short sequence, and together they bind to a sequence that can be 12-18bp long, with a short spacer separating the two regions. Common protein motifs such as the DNA binding Helix-Turn-Helix (HTH) motif also induce the same preference on the regulatory site.

The recent advances in microarray experiments allow to monitor the expression levels of genes in a genome-wide manner [8, 9, 14, 15, 22, 23]. An important aspect of these experiments is that they allow to find groups of genes that have similar expression patterns across a wide range of conditions [12]. Arguably, the simplest biological explanation of co-expression is co-regulation by the same transcription factors.[1]

This observation sparked several works on *in-silico* identification of putative transcription factor binding sites [4, 17, 19–21]. The general scheme that most of these papers take involves two phases. First, they perform, or assume, some clustering of genes based on gene expression measurements. Second, they search for short DNA patterns that appear in the promoter region of the genes in each particular cluster. These works are based to a large extent on methods that were developed to find common motifs in protein and DNA sequences. These include combinatorial methods [6, 19, 21, 24, 25], parameter optimization methods such as Expectation Maximization (EM) [1], and Markov Chain Monte Carlo (MCMC) simulations [18, 20]. See [19] for a review of these lines of work.

The use of expression profiles helps to select relatively "clean" clusters of genes (i.e., most of them are indeed co-regulated by the same factors). Our interest here lies with the second phase, and is thus not limited to gene expression analysis. Given high quality clusters of genes, suspected for any reason to be co-regulated, we address the hardness of the computational problem of finding putative binding sites in these clusters.

In this paper we describe a fast, simple, yet powerful, approach for finding putative binding sites with respect to a given cluster of genes. Like some of the other works we divide this phase into two stages. In the first stage we scan, in an exhaustive manner, for simple patterns from an enumerable class (such as all 7-mers). We use a straightforward, natural, and well understood statistical model for filtering significant patterns out of this class. Using the hyper-geometric distribution, we compute the probability that a subset of genes of the given size will have these many occurrences of the pattern we examine, when chosen randomly from the group of all known genes. In the second stage, we use the patterns

[1] Clearly this is not always the case. Co-regulation can be achieved by other means, and similar expression patterns can be a result of parallel pathways or a close serial relationship. Nonetheless, this is often the case, and a reasonable hypothesis to test.

that were chosen as seeds for training a more expressive *position specific scoring matrix* (PSSM) to model the putative binding site. These models are both more accurate representation of the binding site, and potentially capture much longer conserved regions.

By assuming that most binding sites do contain highly conserved short subsequences and by explicitly using our post-genomic knowledge of all known and putative genes to contrast clusters of genes against the genome background, we acquire quality seeds for the construction of PSSMs through a simplified hyper-geometric model. The seeds allow us to track down potential binding site locations through a specific relatively conserved region within them. We then use these short seeds to guide the construction of potentially much longer PSSMs encompassing more, or possibly the complete binding site. In particular, they allow us to align multiple sequences without resorting to an expensive search procedure (such as MCMC simulations).

Indeed, an important feature of our approach is the evaluation speed. Once we finish a pre-processing stage, we can evaluate clusters very efficiently. The preprocessing is genome-wide and not cluster specific. It can be done only once and stored for all future reference. This is important both for facilitating interactive analysis, and for serving as computationally-cheap quality starting points for other, more complex analysis tools (such as [2]) on top of our method.

In the next three sections we outline our algorithmic approach, discussing significance of events, seed finding, and seed expansion into PSSMs, respectively. In Section 5 we describe experimental and comparative results, and then conclude with a discussion.

2 Scoring Events for Significance

2.1 Preliminaries

Suppose we are given a set of genes $\mathcal{G}$. Ideally, these are all the known and putative genes in a genome. With each gene $g \in \mathcal{G}$ we associate a promoter sequence[2] s_g. For simplicity we assume that each of these sequences is of the same size, L.

Suppose we are now given a subset of genes $G \subset \mathcal{G}$ suspected to be coregulated by some transcription factor. (For example, based on clustering of genes by their expression patterns.) Our aim is to find patterns in the promoter region of these genes, that we will consider as putative binding sites. The assumption being that the co-regulation is mediated by factors that are present in most of the genes in group G, but overall rare in $\mathcal{G}$. Thus, a pattern is considered significant if it is characteristic of G compared to the background $\mathcal{G}$.

Before we discuss what constitutes a pattern in our context, we address the basic statistical definition of a characteristic property. Suppose we find a pattern that appears in the promoter sequences of several genes in G. How do

[2] Or an upstream region that best approximates it, when the transcription start site is unknown.

we measure the significance of these appearances with respect to $\mathcal{G}$? A related question one may ask, is whether the set G is significantly different, in terms of the composition of its upstream region, from $\mathcal{G}$.

For now, we concentrate on events occurring in the promoter region of a gene. We focus on *binary* events, such as "s_g contains the subsequence `ACGTTCG` or its reverse complement". Alternatively, one can consider *counting* the number of occurrences of an event in each promoter sequence, e.g., "the number of times the subsequence `ACGTTCG` appears in s_g". The analysis of such counting events, while attractive in our biological context, is more complex, in particular since multiple occurrences of an event in a sequence are not independent of each other. See [21, 24] for approximate solutions to this problem.

Formally, a binary event E is defined by a *characteristic function* $I_E : \{A, C, G, T\}^\star \rightarrow \{0, 1\}$, that determines whether that event occurred or not in any given nucleotide sequence. Given a set G, we define $\#_E(G) = \sum_{g \in G} I_E(s_g)$ to be the number of times E occurs in the promoter regions of group G. We want to assess the significance of observing E at least $\#_E(G)$ times in G, when taking the set of genes $\mathcal{G}$ as the background for our decision.

There are two general approaches for testing such significance. In both cases we compute *p-values*: the probability of the observations occurring under the *null-hypothesis*. This value serves as a measure of the significance of the pattern - the lower p-value is, the more plausible it is that an observation is significant, rather than a chance artifact. The two approaches differ, however, in the nature of each null-hypothesis.

2.2 Random Sequence Null Hypothesis

In this approach, the null hypothesis assumes that the sequences s_g for $g \in G$ are generated from a background sequence model $P_0(s)$. This background distribution attempts to model "prototypical" promoter regions, but does not include any group-specific motifs. Thus, if the event E detects such special motifs, then the probability of randomly sampling genes that satisfy E is small.

The background sequence model can be, for example, a Markov process of some order (say 2 or 3) estimated from the sequences in $\mathcal{G}$ (or, preferably, from $\mathcal{G} - G$). Using this background model we need to compute the probability $p_E = P_0(I_E(s) = 1)$ that a random sequence of Length L will match the event of interest. Now, if we also assume under the null hypothesis that the n sequences in G are independent of each other, then the number of matches to E in G is distributed $Bin(n, p_E)$. We can then compute the p-value of finding $\#_E(G)$ or more such random sequences by the tail weight of a Binomial distribution.

The key technical issue in this approach is computing p_E. This, of course, depends on the assumed form of the background distribution, and on the complexity of the event. However, even for the simple definition of a pattern as an exact subsequence (i.e., $I_E(s) = 1$ iff s contains a specific subsequence) and background probability of the form of an order 1 Markov chain, the required computation is not trivial. This forces the development of various approximations to p_E of varying accuracy and complexity [4, 7, 21].

2.3 Random Selection Null Hypothesis

Alternatively, in the approach we focus on here, one does not make any assumption about the distribution of promoter sequences. Instead, the null hypothesis is that G was selected at random from $\mathcal{G}$, in a manner that is independent of the contents of the genes' promoter regions.

Assume that $K = \#_E(\mathcal{G})$ out of $N = |\mathcal{G}|$ genes satisfy E. Thus, we require the number[3] of genes that satisfy E in $\mathcal{G}$. The probability of an observation under the null hypothesis is the probability of randomly choosing $n = |G|$ genes in such a way that $k = \#_E(G)$ of them include the event E. This is simply the *hyper-geometric* probability of finding k red-balls among n draws without replacement from an urn containing K red balls and $N - K$ black ones:

$$P_{\text{hyper}}(k \mid n, K, N) = \frac{\binom{K}{k}\binom{N-K}{n-k}}{\binom{N}{n}}$$

The p-value of the observation is the probability of drawing k or more genes that satisfy E in n draws. This requires summing the tail of the hyper-geometric distribution

$$p\text{-}value(E, G) = \sum_{k'=k}^{n} P_{\text{hyper}}(k' \mid n, K, N)$$

The main appeal of this approach lies in its simplicity, both computationally and statistically. This null hypothesis is particularly attractive in the post-genomic era, where nearly all promoter sequences are known. Under this assumption, irrelevant clustering selects genes in a manner that is independent of their promoter region.

2.4 Dealing with Multiple Hypotheses

We have just defined the significance of a single event E with respect to a group of genes G. But when we try many different events $E_1, \ldots, E_M$ over the same group of genes long enough, we will eventually stumble upon a surprising event even in a group of randomly selected sequences, chosen under the null hypothesis.

Judging the significance of findings in such repeated experiments is known as *multiple hypotheses testing*. More formally, in this situation we have computed a set of p-values $p_1, \ldots, p_M$, the smallest corresponding to the most surprising event. We now ask how significant are our findings considering that we have performed M experiments.

One approach is to find a value $q = q(M)$, such that the probability that any of the events (or the smallest one) has a p-value less than q is small. Using the union bound under the null hypothesis we get that

$$P(\min_m p_m \leq t) \leq \sum_m P(p_m \leq q) = M \cdot q$$

[3] But not the identity, simplifying the implied underlying *in-vitro* measurements.

Thus, if we want to ensure that this probability of a false recognition is less than 0.01 (i.e., 99% confidence), we need to set the *Bonfferoni* threshold $q = \frac{0.01}{M}$ (see, for example, [11]).

The Bonfferoni threshold is strict, as it ensures that each and every validated scoring event is not an artifact. Our aim, however, is a bit different. We want to retrieve a set of events, such that *most* of them are not artifacts. We are often willing to tolerate a certain fraction of artifacts among the events we return. A statistical method that addresses this kind of requirement is the *False Discovery Rate* (FDR) method of [3]. Roughly put, the intuition here is as follows. Under the null hypothesis, there is some probability that the best scoring event will have a small p-value. However, if the group was chosen by the null hypothesis, it can be shown that the p-values we compute are distributed uniformly. Thus, the p-value of the second best event is expected to be roughly twice as large as the p-value of the best event. Given this intuition, we should be less strict in rejecting the null hypothesis for the second best pattern and so on.

To carry out this idea, we sort the events by their observed p-values, so that $p_1 \leq p_2 \leq \ldots \leq p_M$. We then return the events $E_1, \ldots, E_k$ where $k \leq M$ is the maximal index such that $p_k \leq \frac{kq}{M}$ and q is the significance level we want to achieve in selecting. We have replaced a strict validation test of single events, with a more tolerable version validating a group of events. We may now detect significant patterns, weaker than the most prominent one, that were previously below the threshold computed for the later.

3 Finding Promising Seeds

3.1 Simple Events

We want to consider patterns over relatively short subsequences. We fix a parameter ℓ that determines the length of the sequences we are interested in. Events are then defined over the space of 4^ℓ ℓ-mers.

Arguably the simplest ℓ-mer pattern is a specific subsequence (or consensus). Thus, if σ is an ℓ-mer it defines the event "σ is a subsequence of s". A useful aspect of such events, is that they are *exhaustively enumerable* for the range of ℓ we are interested in. This suggests examining all ℓ-mer patterns in G and ranking them according to their significance.

However, known binding sites that are identified by biological assays, display variability in the binding sequence. Thus, we do not expect to see only exact matches to the ℓ-mer consensus. Instead, we want to allow approximate matches when we search G. To formalize, consider a distance measure between two ℓ-mers, $d(\sigma, \sigma')$. The simplest such function is the hamming distance. However, we may consider more realistic functions, such as distances that penalize changes in a position specific manner. (Biology suggests, for example, that central positions in short binding sites are more conserved.) For concreteness, we focus on the hamming distance measure in the reminder of the paper. However, we stress that the following discussion applies directly to any chosen distance measure.

Let σ be an ℓ-mer. We define a δ-ball centered around σ to be the set $\text{Ball}_\delta(\sigma)$ of ℓ-mers that are of distance at most δ from σ. Thus, in the hamming distance, example, $\text{Ball}_1(\texttt{AAA}) = \{\texttt{AAA}, \texttt{CAA}, \texttt{GAA}, \texttt{TAA}, \texttt{ACA}, \texttt{AGA}, \texttt{ATA}, \texttt{AAC}, \texttt{AAG}, \texttt{AAT}\}$. We match an event E with $\text{Ball}_\delta(\sigma)$ such that $I_E(s) = 1$ iff s or its reverse complementary contain an ℓ-mer $\in \text{Ball}_\delta(\sigma)$.

Given ℓ and δ we wish to examine all balls that have at least one occurrence in $\mathcal{G}$ (the rest will never appear in any sub group). Balls that occur in all genes in $\mathcal{G}$ are also discarded (as they occur in all genes of any sub group). We denote this set of non-trivial events with respect to $\mathcal{G}$ as $\mathcal{B}_{(\ell,\delta)}$. Note that for $\delta > 0$, it may include balls whose centers do not appear in any promoter region.

Finding the set $\mathcal{B}_{(\ell,\delta)}$ of balls, and annotating for each gene whether it matches each ball can be done in a straightforward manner. The time requirement then is $N \cdot L \cdot 4^\ell$, and the space requirement $N \cdot |\mathcal{B}_{(\ell,\delta)}|$.

This genome-wide pre-processing needs to be done only once. Storing its results we can rapidly compute p-values of all $\mathcal{B}_{(\ell,\delta)}$ events with respect to any proposed subset of genes. We simply look up which events occurred in the genes in the cluster, and then compute the hyper-geometric tail distribution. Furthermore, one may wish to increase, shrink, or shift the regions under consideration (e.g., from 1000bp to 2000bp upstream), or adjust the upstream regions of several genes (say, due to elucidation of exact transcription start site). While in general the pre-processing phase must be repeated, in practice, since it is mainly made up of counting events, we may efficiently subtract, and add, respectively the counts in the symmetrical difference between the old and new sets of strings, avoiding repeating the complete process over again. With many completely sequenced genomes and gene expression data of model organisms in various settings just beginning to accumulate, our division of labour is especially useful.

3.2 Reducing the Event Space

The definition of $\mathcal{B}_{(\ell,\delta)}$, holding all events we wish to examine, may include as many as $\min(4^\ell, LN)$ balls. We note however, that many of these balls overlap. Thus, if σ and σ' are two ℓ-mers that differ, in the hamming distance example[4], in exactly one letter, then the overlap between $\text{Ball}_\delta(\sigma)$ and $\text{Ball}_\delta(\sigma')$ is clearly substantial. Moreover, if we notice that most of the "mass" of these balls (in terms of the number of occurrences in genes in $\mathcal{G}$) lies in the intersection, we expect that the significance of the events defined by both of them will be similar, since they will be highly correlated.

A way to decrease the storage requirements, and thus extend the range of manageable ℓ's can be found by a guided choice of a representative subset of $\mathcal{B}_{(\ell,\delta)}$ during pre-processing. Based on the above intuitions we want a covering set of balls with maximal mass, to minimize the size of the subset, and minimal overlap, to diversify the events themselves. A heuristic solution can be offered in the form of a greedy algorithm. Starting from an empty subset we repeatedly choose balls of maximal mass that do not violate the minimal overlap demand,

[4] Analogous proximity thresholds can be defined for other distance measures.

until we can no longer continue. We now proceed to examine and store the results only for the events corresponding to the chosen balls.

We stress that since this sparsification is done during pre-processing, before we observe any group G, it should not alter the statistical significance of the results we observe when G is later given to us.

4 Learning Finer Representations

4.1 Position Specific Scoring Matrices

Using the methods of the previous section we can collect a set of promising patterns that are significant for G. These patterns are based on the notion of a δ-ball. Biological knowledge about transcription factor binding sites suggests that the definition of a binding site is in fact more subtle. Some positions are highly conserved, while others are less so. In the literature, there are two main representation of such sites. The first is the IUPAC consensus sequences. This approach determines the consensus string of the binding site using a 15 letter alphabet that describe which subset of $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ is possible at each position.

A *position specific scoring matrix* (PSSM) (see, e.g., [10]) offers a more refined representation. A PSSM of length ℓ is an object $\mathcal{P} = \{p_1, \ldots, p_\ell\}$, composed of ℓ column distributions over the alphabet $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$. The distribution p_i, specifies the probability of seeing each nucleotide at the i'th position in the pattern.

Once we have a PSSM $\mathcal{P}$, we can score each ℓ-mer σ by computing its combined probability given $\mathcal{P}$. A more common practice is to compute the log-odds between the PSSM probability and a background probability of nucleotides. Thus, if p_0 is assumed to be the nucleotide probability in promoter regions, then the score of an ℓ-mer σ is:

$$Score_{\mathcal{P}}(\sigma) = \sum_i \log \frac{p_i(\sigma[i])}{p_0(\sigma[i])}$$

If this score is positive σ is more probable according to $\mathcal{P}$ than it is according to the background probability. In practice we set a threshold α (replacing zero) for detecting a pattern. Thus, a pair $(\mathcal{P}, \alpha)$ defines an event $I_{(\mathcal{P},\alpha)}(s)$. This event occurs iff the best matching subsequence of length ℓ in s, or in its reverse complement, has a score higher than α. That is, if

$$\max_i(Score_{\mathcal{P}}(s[i, \ldots, i+\ell-1]), Score_{\mathcal{P}}(\overline{s[i, \ldots, i+\ell-1]}) > \alpha$$

4.2 Selecting a Threshold

Before we discuss how to learn the PSSM, we consider choosing a threshold α for a given PSSM $\mathcal{P}$. It is possible to set $\alpha = 0$, treating the background and the PSSM as equiprobable. However, since the pattern is a rarer event, we want a stricter threshold. Another potential approach tries to reduce the probability of false recognition. That is, to find an α such that the probability that a random background sequence σ will score higher than α is smaller than a pre-specified ϵ. Then, if we want to allow on average one false detection every k genes, we would set $\epsilon = \frac{1}{k*L}$. Unfortunately, we are not aware of an efficient computational procedure to find such thresholds.

Here we suggest a simple alternative. We search for a threshold α, such that the induced detections in the group G will be most significant. Thus, given a group G of genes, and a PSSM $\mathcal{P}$, we search for

$$\alpha^* = \arg\min_{\alpha} p\text{-}value(G, I_{(\mathcal{P},\alpha)})$$

That is, we adjust the threshold α so that the event defined by $(\mathcal{P}, \alpha)$ has the smallest p-value with respect to G. This *discriminative* choice of a threshold ensures that we adjust it to take into account the amount of "spurious" matches to the PSSM outside of G. Thus, we strive for a threshold that maximizes the number of matches within G and at the same time minimizes the number of matches outside G. The use of p-values provides a principled way of balancing these two requirements.

We can find this threshold quite efficiently. We compute the best score of the PSSM over each gene in $\mathcal{G}$, and sort this list of scores. We then evaluate only thresholds which are, say, half way between any two adjacent values in our list of sorted scores (each succeeding threshold admits another gene into the group of supposedly detected events). Using, for example, radix sort, this procedure takes time $O(NL)$.

4.3 Learning PSSMs

Learning PSSMs is composed of two tasks. Estimating the parameters of the PSSM given a set of training sequences that are examples of the pattern we want to match, and finding these sequences. The latter is clearly a harder problem and requires some care.

We start with the first task. Suppose we are given a collection $\sigma_1, \ldots, \sigma_n$ of ℓ-mers that correspond to *aligned* sites. We can easily estimate a PSSM $\mathcal{P}$ that corresponds to these sequences. For each position i, we count the number of occurrences of each nucleotide in that position. This results in a count $N(i, c) = \sum_j 1\{\sigma_j[i] = c\}$.

Given the counts we estimate the probabilities. To avoid entries with zero probability, we add *pseudo-counts* to each position. Thus, we assign

$$p_i(c) = \frac{N(i, c) + \gamma}{n + 4\gamma} \tag{1}$$

The key question is how to select the training sequences and how to align them. Our approach builds on our ability to find seeds of conserved sequences. Suppose that we find a significant δ-ball using the methods of the previous section. We can then use this as a *seed* for learning a PSSM. The simplest approach takes the ℓ-mers that match the ball within the promoter regions of

G as the training sequences for the PSSM. The learned PSSM then quantifies which differences are common among these sequences and which ones are rare. This gives a more refined view of the pattern that was captured by the δ-ball.

This simple approach learns an ℓ-PSSM from the δ-ball events found in the data. However, using PSSMs we can extend the pattern to a much longer one. We start by aligning not only the sequences that match the δ-ball, but also their flanking regions. These are aligned by virtue of the alignment of the core ℓ-mers. We can then learn a PSSM over a much wider region (say 20bp). If there are conserved positions outside the core positions, this approach will find them.[5]

Consider, for example, a HTH DNA binding motif, or a binding factor dimer, where each component matches 6-10bps with several unspecific gap positions between the two specific sites. If we find one of the two sites using the methods of the previous sections, then growing a PSSM on the flanking regions allows us to discover the other conserved positions.

Once we construct such an initial PSSM, we can improve it using a standard EM-like iterative procedure. This procedure consists of the following steps. Given a PSSM $\mathcal{P}_0$, we compute a threshold α_0 as described above. We then consider each position in the training sequences and compute the probability that the pattern appears at that position. Formally, we compute the likelihood ratio $(\mathcal{P}_0, \alpha_0)$ assigns to the appearance of the pattern at $s[i, \ldots, i+\ell-1]$. We then convert this ratio to a probability by computing

$$\rho_{s,i} = \mathrm{logit}(Score_{\mathcal{P}_0}(s[i, \ldots, i+\ell-1]) - \alpha_0)$$

where $\mathrm{logit}(x) = 1/(1+e^{-x})$ is the *logistic function*. We then re-scale these probabilities by dividing by a normalization factor Z_s so that the posterior probability of observing the pattern in s and its reverse complement sums to 1. Once we have computed these posterior probabilities, we can accumulate *expected counts*

$$N(i,c) = \sum_g \sum_j \frac{\rho_{s_g,j}}{Z_{s_g}} 1\{s_g[j+i] = c\}.$$

These represent the expected number of times that the i'th position in the PSSM takes the value c, based on the posterior probabilities.

Once we collected these expected counts, we re-estimate the weights of the PSSM using Eq. 1 to get a new a PSSM. We optimize the threshold of this PSSM, and repeat the process. Although this process does not guarantee improvement in the p-value of the learned PSSM, it is often the case that successive iterations do lead to significant such improvements. Note that our iterations are analogous to EM's hill-climbing behaviour, and differ from Gibbs samplers where one performs a stochastic random walk aimed at a beneficial equilibrium distribution.

[5] This assume that there are no variable lengths gaps inside the patterns. The structural constraints on transcription factors suggest that these are not common.

Table 1. Selected results on binding site regions of several yeast data sets, comparing our findings with those of MEME.

Source/ Cluster	Trans. Factor	Consensus	Seed rank	Seed p-value	PSSM rank	PSSM p-value	MEME ≤ 8 rank	MEME ≤ 8 e-value	MEME ≤ 50 rank	MEME ≤ 50 e-value
Spellman et al. [22]										
CLN2	MBF	ACGCGT	1	4e-26	1	3e-42	1	1e-18	1	7e-31
SIC1	SWI5p	CCAGCA	1	1e-07	1	1e-12	1	8e-00	8	5e+02
Tavazoie et al. [23]										
3	putative	GATGAG	2	9e-07	5	6e-09	4	1e+06	2	1e-14
	putative	GAAAAatT	3	4e-07	2	1e-11	23	8e+07	3	7e-10
8	STRE	aAGGgG	1	6e-07	3	4e-06	20	1e+08	–	–
14	putative	TTCGCGT	1	2e-09	2	7e-11	13	1e+07	–	–
	putative	TGTTTgTT	3	2e-07	–	–	–	–	13	4e+05
30	MET31/32p	gCCACAgT	1	2e-11	1	2e-11	2	5e+02	8	1e+03
Iyer et al. [16]										
MBF	MBF	ACGCGT	1	1e-12	1	3e-18	3	1e+04	19	1e-03
SBF	SBF	CGCGAAA	1	1e-32	1	1e-37	2	1e-17	–	–

5 Experimental Results

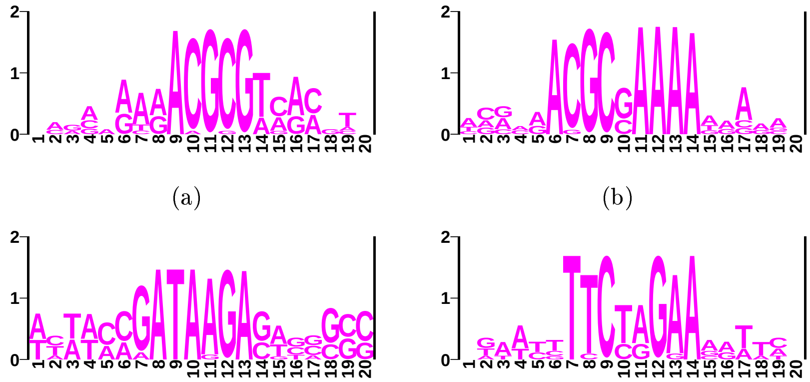
We performed several experiments on data from the yeast genome to evaluate the utility and limitations of the methods described above. Thus, we focused on several recent examples from the literature that report binding sites found either using computational tools or by biological verification. To better calibrate the results, we also applied MEME [1], one of the standard tools in this field, on the same examples.

In this first analysis we chose to use the simple hamming distance measure and treat the 1000bp sequence upstream of the ORF starting position as the promoter region. We note that the latter is a somewhat crude approximation, as this region also contains an untranslated region of the transcript.

We ran our method in two stages. In the first stage, we searched for patterns of length 6–8 with δ ranging between 0–2 mismatches, and an allowed ball overlap factor of 0–1. Generally speaking, in these runs the patterns found with no mismatches or ball overlaps had better p-values. This happens because we search for relatively short patterns, allowing for a non-trivial probability of a random match. For this reason we report below only results with exact matches and no overlap. We believe that higher values of both parameters will be useful for longer patterns (say of length 12 or 13). In the second stage we run the EM-like procedure described above on all the patterns that received significant scores. We chose to learn PSSMs of width 20 using 15 iterations of our procedure.

To compare the results of these two stages, we ran MEME (version 3.0.3) in two configurations. The first restricted MEME to retrieve only short patterns of width 6–8, corresponding to our ℓ-mers stage. The second configuration used MEME's own defaults for pattern retrieval resembling our end product PSSMs.

We applied our procedure to several data sets from the recent literature. Selected results are summarized in Table 1. In this table we rank the top results from the different runs of each procedure by their p-values (or e-values) reported by the programs after removing repeated patterns. We report the relative rank



(a) (b) (c) (d)

Fig. 1. Examples of PSSMs learned by our procedure. (a) CLN2 cluster. (b) SBF cluster. (c) Gasch *et al.* Cluster M. (d) Gasch *et al.* Cluster I/J.

of the patterns singled out in the literature and their significance scores. We discuss these results in order.

The first data set is by Spellman *et al.* [22]. They report several cell-cycle related clusters of genes. In a recent paper, Sinha and Tompa [21] report results of a systematic search for binding sites in these clusters of IUPAC consensus regions using a random sequence null hypothesis utilizing a Markov chain of order 3. The main technical developments in [21] are methods for approximating the p-value computation with respect to such a null-hypothesis.

We examined two clusters reported on by Sinha and Tompa. In the first one, CLN2, our method identifies the pattern `ACGCGT` and various expansions of it. This pattern was found using patterns of length 6, 7, and 8 with significant p-values. The PSSMs learned from these patterns were quite similar, all containing the above motif. Figure 1(a) shows an example. In the second cluster, SIC1, the signal appears with a marginal p-value (close to the Bonfferoni cutoff) already at $\ell = 6$. The trained PSSM recovers the longer pattern with a significant p-value. In both cases, the top ranking patterns correspond to the known binding site.

The second data set is by Tavazoie *et al.* [23]. That paper also examines cell-cycle related expression levels that were grouped using k-means clustering. They examined 30 clusters, and applied an MCMC-based procedure for finding PSSM patterns in the promoter regions of genes in each cluster. We examined the clusters they report as statistically significant, and were able to reproduce binding sites that are very close to the PSSMs they report; see Table 1.

In a recent paper, Iyer *at al.* [16] identify, using experimental methods, two groups of genes that are regulated by the MBF/SBF transcription factor. Here, again, we managed to recover the binding sites they discuss with high confidence. For example, we show one of our matching PSSMs in Figure 1(b).

Finally, we discuss the recent data set of yeast response to environmental stress by Gasch *et al.* [14]. We report on two clusters of genes "M", and "I/J". In cluster M the string `CACGTGA` is found in several of the highest scoring patterns. However, when we turned to grow PSSMs out of our seeds, a matrix of a lower ranking seed `GATAAGA` exceeded the rest, exemplifying that seed ordering is not necessarily maintained when the patterns are extended. The latter, more prominent PSSM is shown in Figure 1(c). In cluster I/J a significant short pattern rising above our threshold is not found. However when we extended the top most seed we obtained the PSSM of Figure 1(d) which both nearly crosses our significance threshold, and holds biological appeal, showing two conserved short regions flanking a less conserved 2-mer.

In general, the scores of the learned PSSMs vary. In some cases, the best seeds yield the best scoring PSSMs. More often, the best scoring PSSM corresponds to a seed lower in the list (we took into account only seeds that have p-value matching the FDR decision threshold). In most cases the PSSM learned to recognize regions flanking the seed sequence. In some cases more conserved regions were discovered. In general our approach manages to identify short patterns that are close to the pattern in the data. Moreover, using our PSSM learning procedure we are able to expand these into more expressive patterns.

We note that in most analysed cases MEME also identified the shorter patterns. However, there are two marked differences. First and foremost is run time. Compared on a 733 MHz Pentium III Linux machine our seed discovery programs ran between half a minute and an hour, exhaustively examining all possible patterns, while the EM-like PSSM growing iterations added a couple of minutes. The shortest MEME run on the same data sets took about an hour, while longer ones ran for days, when asked to return only the top thirty patterns. Second, MEME often gave top scores to spurious patterns that are clear artifacts of the sequence distributions in the promoter regions (such as poly A's). When using MEME one can try to avoid these problems by supplying a more detailed background model. This has the effect of removing most low complexity patterns from the top scoring ones. Our program avoids most of these pitfalls by performing its significance tests with respect to the genome background to begin with.

6 Discussion

In this paper we examined the problem of finding putative transcription factor binding sites with respect to a selected group of genes. We advocate significance calculations with respect to the *random selection* null hypothesis. We claim that this hypothesis is both simple and clear and is more suitable for gene expression experiments than the *random sequence* null hypothesis. We then use a simple hyper-geometric test in a framework for constructing models of binding sites. This framework starts by *systematically* scanning a family of simple "seed" patterns. These seeds are then used for building PSSMs. We describe how to construct statistical tests to select the most surprising threshold value for a

PSSM and combine this with an EM-like iterative procedure to improve it. We thus combine a first phase of kernel identification based on a rigorous statistical analysis of word over-representation, with a subsequent phase of optimization, leading to a PSSM, which can be used to scan sequences for new matches of the putative regulatory motif.

We showed that even before performing iterative optimization of the PSSMs, our method recovers highly selective seed patterns very rapidly. We reconstructed results from several recent papers that use more elaborate and computationally intensive tools for finding binding sites, as well as present novel binding sites.

A potential weakness of our model is the fact that we disregard multiple copies of a match in the same sequence (the restriction to binary events). Despite the fact that this phenomenon is known to happen in eukaryotic genes, we recall that a mathematical analysis of counting the number of occurrences in a single string is more elaborate, and computationally intensive. This may indeed lead in such cases to under-estimation, which is problematic mainly for small clusters of co-regulated genes. The recognition of two conserved patterns separated by a relatively long spacer (say of 10bp or more), resulting from a HTH motif or a dimer complex, can however be attacked by looking for proximity relationships between pairs of occurrences of different significant seeds.

As this field is showing an influx of interest, our work resembles several others in different aspects. We highlight only the most relevant ones.

The use of the hyper-geometric distribution in the context of finding binding sites is used by Jensen and Knudsen [17] to find short conserved subsequences of length 4–6 bp. They demonstrate the ability to reconstruct sequences, but suffer statistical problems when they consider longer ℓ-mers, due to the large number of competing hypotheses.

Already in Galas *et al.* [13], word statistics are used to detect over-represented motifs, and a definition of a general concept of "word neighborhood" is given similar to the ball definition we give here. However, the analysis there is restricted to over-representations at specific positions with respect to a common point of reference across all sequence, deeming it mostly appropriate for prokaryotic transcription or translation promoter region elucidation.

The general outline of our approach is similar to that of Wolferstetter *et al.* [27] and Vilo *et al.* [26]. Both search for over-represented words and try to extend them. Vilo *et al.* examine ℓ-mers of varying sizes that are identified by building a suffix tree for the promoter regions. Then, they use a binomial formula for evaluating significance. For the clustering they constructed, this resulted in a very large pool of sequences (over 1500). They use multiple alignment-like procedure for combining these ℓ-mers into longer consensus regions. Thus, to learn longer binding sites with variable position, they require overlapping subsequences to be present in the data. This is in contrast to our approach that uses PSSMs to extend the observed patterns, and so is more robust to highly variable positions that flank the conserved region.

Van Helden *et al.* [24] also use binomial approach. They try to take into consideration the presence of multiple copies of a motif in the same sequence, but suffer from resulting inaccuracies with respect to auto-correlating patterns. Our work can be seen as generalizing this approach in several respects, including the use of a hyper-geometric null model, the discussion of general distance functions and event space coarsening, and the iterative PSSM improvement phase.

There are several directions in which we can extend our approach, some of them embedding ideas from previous works into our context.

First, in order to estimate the sensitivity of our model it will be interesting to examine it on smaller, and known, gene families, as well as on synthetic data sets, as those advocated in [19]. Extending our empirical work beyond yeast should also provide new insights and challenges.

Our method treats the complete promoter region as a uniform whole. However, biological evidence suggests that the occurrence of binding sites can depend on the position within the promoter sequence [22]. We can easily augment our method by defining events on sub-regions within the promoter sequence. This will facilitate the discovery of subsequences specific to certain positions. Another biological insight already mentioned is the phenomena of two conserved patterns separated by a relatively long spacer. In the case of homeodimers we can easily expand our scope to handle events that require two appearances of the subsequence within the promoter region. Otherwise, we can try to extend our PSSMs further to flank the seed while weighting each column such as to allow for longer spacers between meaningful sub-patterns.

So far we have looked for contiguous conserved patterns within the binding site. More complex extensions involve defining new distance measures that incorporate preferences for more conserved positions in specific positions in the pattern, and random projection techniques, akin to [5], which will allow us to easily handle longer ℓ-mers. We can also further generalize our model by allowing ourselves to express our ℓ-mer centroids over the IUPAC alphabet. This allows both for a reduction of the event space and the natural incorporation of biological insight, as outlined above. Our current method for diluting the set of "covering" δ-balls is highly heuristic. Interesting theoretical issues include the formal criteria we should optimize in selecting this approximating set of δ-balls and how to efficiently optimize with respect to such a criterion. Finally, we intend to combine the putative sites we discover with learning methods that learn dependencies between different sites and between sites and other attributes such as expression levels and functional annotations [2].

Acknowledgments

The authors thank Zohar Yakhini for arousing our interest in this problem and for many useful discussions relating to this work, and the referees for pointing out further relevant literature. This work was supported in part by Israeli Ministry of Science grant 2008-1-99, and an Israel Science Foundation infrastructure grant. GB is also supported by a grant from the Ministry of Science, Israel. NF is also supported by an Alon Fellowship.

References

1. T.L. Bailey and Elkan C. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. In *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, volume 2, pages 28–36. 1994.
2. Y. Barash and N. Friedman. Context-specific Bayesian clustering for gene expression data. In *Proc. Ann. Int. Conf. Comput. Mol. Biol.*, volume 5, pages 12–21. 2001.
3. Y. Benjamini and Y. Hochberg. Controlling the False Discovery Rate: a practical and powerful approach to multiple testing. *J. Royal Statistical Society B*, 57:289–300, 1995.
4. A. Brazma, I. Jonassen, J. Vilo, and E. Ukkonen. Predicting gene regulatory elements in silico on a genomic scale. *Genome Res.*, 8:1202–15, 1998.
5. J. Buhler and M. Tompa. finding motifs using random projections. In *Proc. Ann. Int. Conf. Comput. Mol. Biol.*, volume 5, pages 69–76. 2001.
6. H. J. Bussemaker, H. Li, and E. D. Siggia. building a dictionary for genomes: identification of presumptive regulatory sites by statistical analysis. *PNAS*, 97(18):10096–100, 2000.
7. H. J. Bussemaker, H. Li, and E. D. Siggia. Regulatory element detection using a probabilistic segmentation model. In *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, volume 8, pages 67–74. 2000.
8. S. Chu, J. DeRisi, M. Eisen, J. Mullholland, D. Botstein, P. Brown, and I. Herskowitz. The transcriptional program of sporulation in budding yeast. *Science*, 282:699–705, 1998.
9. J. DeRisi., V. Iyer, and P. Brown. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, 282:699–705, 1997.
10. R. Durbin, S. Eddy, A. Krogh, and G. Mitchison. *Biological Sequence Analysis : Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1998.
11. R. Durrett. *Probablity Theory and Examples*. Wadsworth and Brooks, Cole, California, 1991.
12. M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *PNAS*, 95:14863–14868, 1998.
13. D. J. Galas, M. Eggert, and M. S. Waterman. Rigorous pattern-recognition methods for dna sequences: analysis of promoter sequences from *Escherichia coli*. *J. Mol. Biol.*, 186:117–28, 1985.
14. A. P. Gasch, P. T. Spellman, C. M. Kao, O. Carmel-Harel, M. B. Eisen, G. Storz, D. Botstein, and P. O. Brown. Genomic expression program in the response of yeast cells to environmental changes. *Mol. Bio. Cell*, 11:4241–4257, 2000.
15. T. R. Hughes, M. J. Marton, A. R. Jones, C. J. Roberts, R. Stoughton, C. D. Armour, H. A. Bennett, E. Coffey, H. Dai, Y. D. He, M. J. Kidd, A. M. King, M. R. Meyer, D. Slade, P. Y. Lum, S. B. Stepaniants, D. D. Shoemaker, D. Gachotte, K. Chakraburtty, J. Simon, M. Bard, and S. H. Friend. Functional discovery via a compendium of expression profiles. *Cell*, 102(1):109–26, 2000.
16. V. R. Iyer, C. E. Horak, C. S. Scafe, D. Botstein, M. Snyder, and P. O. Brown. Genomic binding sites of the yeast cell-cycle transcription factors sbf and mbf. *Nature*, 409:533 – 538, 2001.
17. L. J. Jensen and S. Knudsen. Automatic discovery of regulatory patterns in promoter regions based on whole cell expression data and functional annotation. *Bioinformatics*, 16:326–333, 2000.

18. C. E. Lawrence, S. F. Altschul, M. S. Boguski, J. S. Liu, R. F. Neuwald, and J. C. Wooton. Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. *Science*, 262:208–214, 1993.
19. P.A. Pevzner and S.H. Sze. Combinatorial approaches to finding subtle signals in dna sequences. In *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, volume 8, pages 269–78. 2000.
20. F.P. Roth, P.W. Hughes, J.D. Estep, and G.M. Church. Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation. *Nat. Biotechnol.*, 16:939–945, 1998.
21. S. Sinha and M. Tompa. A statistical method for finding transcription factor binding sites. In *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, volume 8, pages 344–54. 2000.
22. P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, and B. Futcher. Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Mol. Biol. Cell*, 9(12):3273–97, 1998.
23. S. Tavazoie, J. D. Hughes, M. J. Campbell, R. J. Cho, and G. M. Church. Systematic determination of genetic network architecture. *Nat Genet*, 22(3):281–5, 1999. Comment in: Nat Genet 1999 Jul;22(3):213-5.
24. J. van Helden, B. Andre, and J. Collado-Vides. Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. *J. Mol. Biol.*, 281(5):827–42, 1998.
25. J. van Helden, A. F. Rios, and J. Collado-Vides. discovering regulatory elements in non-coding sequences by analysis of spaced dyads. *Nucl. Acids Res.*, 28(8):1808–18, 2000.
26. J. Vilo, A. Brazma, I. Jonassen, A. Robinson, and E. Ukkonen. Mining for putative regulatory elements in the yeast genome using gene expression data. In *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, volume 8, pages 384–94. 2000.
27. F. Wolfertstetter, K. Frech, G. Herrmann, and T. Werner. Identification of functional elements in unaligned nucleic acid sequences by a novel tuple search algorithm. *Comput. Appl. Biosci.*, 12(1):71–80, 1996.