



Separating real motifs from their artifacts

Mathieu Blanchette¹ and Saurabh Sinha^{1,‡}

¹Department of Computer Science and Engineering, University of Washington, Box 352350, Seattle, WA 98195-2350, U.S.A.

ABSTRACT

The typical output of many computational methods to identify binding sites is a long list of motifs containing some real motifs (those most likely to correspond to the actual binding sites) along with a large number of random variations of these. We present a statistical method to separate real motifs from their artifacts. This produces a short list of high quality motifs that is sufficient to explain the over-representation of all motifs in the given sequences. Using synthetic data sets, we show that the output of our method is very accurate. On various sets of upstream sequences in *S. cerevisiae*, our program identifies several known binding sites, as well as a number of significant novel motifs.

Contact: {blanchem,saurabh}@cs.washington.edu

INTRODUCTION

There has been much recent work on computational methods to identify putative binding sites for regulatory factors in DNA sequences. One popular way to do so is to search for motifs (features that occur surprisingly often in a given set of sequences). Numerous tools have been developed for doing this. (See, for example, Lawrence et al. (1993), Bailey and Elkan (1995), van Helden et al. (1998), and Sinha and Tompa (2000).) The putative sites thereby found can then be verified for function, for example by mutagenesis experiments.

A set of sequences having binding sites for a few different factors typically contains hundreds of statistically over-represented motifs, most of them being minor variations of the true binding sites. How does one extract these few “real” motifs from the vast number that are simply artifacts of these few? For example, suppose a factor binds to TCACGCT in a set of sequences, causing this motif to be over-represented. Many of its variations, e.g. CACGCTT or TCACGCW, are also likely to be over-represented, simply because each has its number of occurrences artificially increased by the presence of TCACGCT. These variations are probably not accurate descriptions of the binding site and we would like to separate these “artifact” motifs from the “real” ones. Notice that if we took into account the fact that TCACGCT is over-represented, the high counts of

CACGCTT or TCACGCW might not be surprising anymore. Based on this intuition, we formulate the above-mentioned problem of extracting real motifs as follows:

BEST EXPLANATORS PROBLEM

Given: A set of sequences S , and a set X of motifs in S .

Find: The smallest subset E (the real motifs, called the *explanators*) of X such that, if we take into account the occurrences of motifs in E , the occurrences of motifs in X are *no longer surprising*[†].

This is the main problem we address in this paper. This is not a motif-finding problem *per se*, but rather a post-processing step that will improve the accuracy of the motifs reported.

We begin by illustrating the importance of this problem for motif-finding applications. Any motif-finding technique has to make a trade-off between good soundness (reporting few or no motifs that are not actual binding sites) and good completeness (missing few or no binding sites). Some motif-finding tools (e.g. van Helden et al. (1998), Sinha and Tompa (2000)), with an emphasis on completeness, produce a long list of motifs that are statistically over-represented. Since this list mostly contains artifact motifs, it has low soundness. This issue is especially relevant in the case of the motif-finding program YMF (Sinha and Tompa (2000)) where, in a list of motifs ordered by statistical significance, the first hundreds of positions may be occupied by artifacts of a single strong motif, while a second real motif is ranked below these in the list. In fact, this problem arises with any algorithm that is geared towards high completeness. Some programs (e.g. Vilo et al. (2000)) deal with this problem by clustering together motifs with high sequence similarity, thereby improving the readability of the output. Sequence similarity may, however, not be a reliable criterion for clustering motifs. For example, the factor MCB binds to WCGCGW (Zhu and Zhang (1999)) which has a five residue overlap with CNCGAAA, the binding site for SCB. Sequence similarity may wrongly cluster these together. One way to achieve high soundness, used for example by Rocke and Tompa (1998), is to iteratively find the most significant motif and mask its occurrences in the sequences, so that none of its variations

[‡]The two authors contributed equally to the paper

[†]This notion of surprise will be formalized later

will be found in future iterations. Clearly, this technique forbids finding any overlapping motif that is significant in its own right. For example, the factor Gal4p binds to CGGNNNNNNNNNNCCG, a strong motif in the GAL family. Moreover, a significant fraction of its binding sites also have the motif CGGNNNCTS, aligned with the canonical motif. The masking strategy would prevent us from finding this second motif.

We believe that a correct way to improve the soundness of a list of motifs, without losing on completeness, is by solving the BEST EXPLANATORS problem. This separates real motifs (the explanators) from their artifacts. The explanators thus found are likely to be accurate descriptions of the actual binding sites. In this paper, we present a statistical framework for reasoning about motif similarities and propose an algorithm to solve the BEST EXPLANATORS problem. First, we introduce and quantify the notion of *motif explanation*, which is the cornerstone of our approach. We then present our algorithm and validate our methodology using synthetic data sets. Finally, we present previously identified as well as novel motifs found in various sets of yeast genes.

MOTIFS AND EXPLANATIONS

In this section, we describe a statistical framework that formalizes the BEST EXPLANATORS problem, and derive a quantitative measure of how well motifs explain each other's occurrences. A motif m is a string over Σ , the alphabet of the fifteen IUPAC ambiguity codes for nucleotides. This model, while being less expressive than the alternate weight matrix model, is more amenable to exact algorithms, for example through enumeration (see van Helden et al. (1998), Tompa (1999), and Sinha and Tompa (2000)). In this model, the task of motif-finding is to find all motifs whose *count* (number of occurrences) in the input sequences is significantly greater than that expected if the input sequences were random. Such motifs are said to be *over-represented*.

We shall now examine the occurrences of two (or more) motifs, in relation to each other. Suppose we have one input sequence s and consider two motifs e and m . We say that e *explains* m if the count of m is *not* significantly larger than that expected when we factor in our knowledge of where e occurs in s . Thus, if e explains m , the occurrences of m can be interpreted as mere chance occurrences. If e is output as a putative binding site, reporting m would be unnecessary, and possibly misleading. Consider, for example, a sequence where the motif ACGCCW occurs 80 times, though it was expected to occur only 20 times. Consider also ACGCCA, expected to occur about 10 times, and occurring 40 times. Treated separately, both of these might be considered to be over-represented. However, if we factor in the fact that

ACGCCW occurs 80 times, we shall expect ACGCCA to occur about 40 times, and so the former explains the latter. Notice that the explanation relation is not symmetric. The notion of one motif explaining another is the key idea developed and exploited in this paper. The rest of this section shall formalize this notion.

Conditional probabilities of occurrences

The goal of this section is to derive formulas to compute the conditional expectation and variance of the count of one motif in a random sequence, given the occurrences of other motifs. These statistics will be used later to obtain a measure of explanation. For simplicity, we shall present the calculations in terms of occurrences on one strand only, though it is easy to generalize them to occurrences on both strands, and our implementation considers this general case. We shall assume that a random sequence is one that is generated using a k^{th} -order Markov model, with a transition matrix M chosen to mimic the non-coding sequence of the organism considered.

For a motif m and an input sequence s of length l , we define a binary vector E^m , of length l , to summarize our knowledge about the occurrences of m in s .

$$\forall i = 1 \dots l, \quad E_i^m = \begin{cases} 1 & \text{if motif } m \text{ starts at position } i \text{ in } s \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

We first show how to compute the probability of some motif m occurring at position i , given a set of positions where motifs e_1, \dots, e_c occur:

$$Pr[E_i^m = 1 \mid \bigwedge_{j=1 \dots l} \bigwedge_{k: E_j^{e_k} = 1} E_j^{e_k} = 1] = \frac{Pr[E_i^m = 1 \wedge (\bigwedge_{j=1 \dots l} \bigwedge_{k: E_j^{e_k} = 1} E_j^{e_k} = 1)]}{Pr[\bigwedge_{j=1 \dots l} \bigwedge_{k: E_j^{e_k} = 1} E_j^{e_k} = 1]} \quad (2)$$

Both the numerator and the denominator of (2) are easy to compute, as both are of the form $Pr[E_{i_1}^{m_1} = 1 \wedge E_{i_2}^{m_2} = 1 \wedge \dots \wedge E_{i_r}^{m_r} = 1]$. This probability is computed by building a sequence T of length l as follows: Start with $T = \text{NNN} \dots \text{NNN}$. Then, for $p = 1 \dots r$, write motif m_p at position i_p in T by specializing each symbol in T to the corresponding symbol in m_p . (The specialization of two IUPAC symbols is the most general symbol that is included in both of them.) If, for some motif m_p , there is no specialization possible, the resulting probability is zero. Otherwise, simply compute the probability of the resulting sequence T using the Markov chain M .

We now want to compute, for each position i , the value $p_i = Pr[E_i^m = 1 \mid E^{e_1}, E^{e_2}, \dots, E^{e_c}]$, that is, the probability that motif m occurs in sequence s at position i , given the positions of occurrences *and non-occurrences*

of motifs e_1, e_2, \dots, e_ζ . We have

$$p_i = Pr[E_i^m = 1 | (\bigwedge_{\substack{j=1 \dots l \\ k: E_j^{e_k} = 1}} E_j^{e_k} = 1) \wedge (\bigwedge_{\substack{j=1 \dots l \\ k: E_j^{e_k} = 0}} E_j^{e_k} = 0)] \quad (3)$$

Notice that if two positions a and b are sufficiently far apart ($|a-b| > c_1$, for some appropriately chosen c_1), then the fact that some motif e_i occurs at position a has very little influence on the probability that m occurs at position b . Similarly, if $|a-b| > c_2$, for some suitable c_2 , then the knowledge of the *non*-occurrence of e_i at position a has very little effect on the probability that motif m occurs at position b . Notice that the influence of non-occurrence dies much faster than that of a motif's occurrence, and thus for a same degree of approximation, we can choose $c_2 < c_1$. Thus, we have

$$p_i \approx Pr[E_i^m = 1 | (\bigwedge_{\substack{j: |j-i| \leq c_1 \\ k: E_j^{e_k} = 1}} E_j^{e_k} = 1) \wedge (\bigwedge_{\substack{j: |j-i| \leq c_2 \\ k: E_j^{e_k} = 0}} E_j^{e_k} = 0)] \quad (4)$$

Using Bayes' theorem a couple of times, we can write (4) as

$$\frac{1 - Pr[\bigvee_{\substack{j: |j-i| \leq c_2 \\ k: E_j^{e_k} = 0}} E_j^{e_k} = 1 | (\bigwedge_{\substack{j: |j-i| \leq c_1 \\ k: E_j^{e_k} = 1}} E_j^{e_k} = 1) \wedge (E_i^m = 1)]}{1 - Pr[\bigvee_{\substack{j: |j-i| \leq c_2 \\ k: E_j^{e_k} = 0}} E_j^{e_k} = 1 | (\bigwedge_{\substack{j: |j-i| \leq c_1 \\ k: E_j^{e_k} = 1}} E_j^{e_k} = 1)]} \times Pr[E_i^m = 1 | (\bigwedge_{\substack{j: |j-i| \leq c_1 \\ k: E_j^{e_k} = 1}} E_j^{e_k} = 1)]$$

Finally, by a simple application of the Inclusion-Exclusion principle, we can write, for any condition C ,

$$Pr[\bigvee_{\substack{j: |j-i| \leq c_2 \\ k: E_j^{e_k} = 0}} E_j^{e_k} = 1 | C] = \sum_{X \in \mathcal{2}^{\{(j,k): |i-j| \leq c_2 \wedge E_j^{e_k} = 0\}}} (-1)^{|X|+1} Pr[\bigwedge_{(j,k) \in X} (E_j^{e_k} = 1) | C] \quad (5)$$

We have now reduced p_i to an expression where each term is computable using Equation 2. Equation 6 contains sums over a potentially exponential-sized set of elements. However, for most choices of X , we have $Pr[\bigwedge_{(j,k) \in X} (E_j^{e_k} = 1) | C] = 0$, because it leads to illegal overlaps. For most sets of motifs, the number of ways to overlap k motifs in a region of size $2c_2 + 1$ is usually quite small (c_2 is typically set to 1 or 2), and thus the sum generally contains few non-zero terms. Moreover, it is easy to efficiently generate those legal overlaps.

We can now readily compute the conditional expectation of N_m , the number of occurrences of some motif m , given the occurrences of e_1, \dots, e_ζ .

$$\mu_m = \mathbb{E}[N_m | E^{e_1}, \dots, E^{e_\zeta}] = \sum_{i=1 \dots l} p_i$$

We can also compute the conditional variance of N_m :

$$\begin{aligned} \sigma_m^2 &= Var[N_m | E^{e_1}, \dots, E^{e_\zeta}] \\ &= \mathbb{E}[N_m^2 | E^{e_1}, \dots, E^{e_\zeta}] - \mathbb{E}[N_m | E^{e_1}, \dots, E^{e_\zeta}]^2 \\ &= \sum_{i=1 \dots l} (\sum_{j: |j-i| \leq c_3} Pr[E_i^m = 1 \wedge E_j^m = 1 | E^{e_1}, \dots, E^{e_\zeta}] \\ &\quad + \sum_{j: |j-i| > c_3} Pr[E_i^m = 1 \wedge E_j^m = 1 | E^{e_1}, \dots, E^{e_\zeta}]) - \mu_m^2 \\ &\approx \sum_{i=1 \dots l} (\sum_{j: |j-i| \leq c_3} Pr[E_i^m = 1 \wedge E_j^m = 1 | E^{e_1}, \dots, E^{e_\zeta}] \\ &\quad + \sum_{j: |j-i| > c_3} p_i p_j) - \mu_m^2 \end{aligned} \quad (6)$$

The approximation leading to Equation 6 is based on the idea that the occurrences of a motif at two sufficiently distant positions in the sequence are almost independent. This approximation reduces the number of times Equation 2 needs to be computed from quadratic to linear in l .

The above calculations assumed a single input sequence. The generalization to multiple sequences is simple: since the counts can be assumed independent across sequences, we can compute the total expectation and variance by summing over all sequences.

Objective measure of explanation

We can now compute a statistic $Z(m|e_1, \dots, e_k) = \frac{N_m - \mu_m}{\sigma_m}$ that we call the *conditional z-score* of m , given e_1, \dots, e_k . This statistic is our measure of motif explanation. A high value of this statistic implies that the motif m occurs more often than is expected in random sequences, even when we condition on the known occurrences of motifs e_1, \dots, e_k . A low value, on the other hand, suggests that m is *not* over-represented under the above condition, which in turn means that motifs e_1, \dots, e_k together *explain* m in the sense described at the beginning of this section. Notice that there is no simple relationship between $Z(m|e_1, \dots, e_k)$ and $Z(m|e_i)$. Often, $Z(m|e_1, \dots, e_k) \approx Z(m|e_i)$ for some i , if e_i is very similar to m . However, it is possible that several motifs are needed to explain m well. In the rest of this paper, we shall write $Z(m|E)$ to denote the conditional z-score, where $E = \{e_1, \dots, e_k\}$. For $E = \phi$, we call $Z(m|E) = Z(m)$ the *unconditional z-score* of m .

The analytical form of the distribution of the statistic $Z(m|e_1, \dots, e_k)$ is unknown. The unconditional z-score, $Z(m)$, has been shown to be normally distributed, for large sequences (Waterman (1995), Nicodème et al.

(1999)). This statistic was used for motif finding by van Helden et al. (2000), Tompa (1999), and Sinha and Tompa (2000), among others. Using simulated data and a Chi-square test, we verified that, for $k > 0$, a similar normality assumption can be made for large sequences. For input sequences of smaller size (e.g. less than 4000 bp), some motifs fail the normality test. However, as the validation section shows, this turns out not to be a major problem in practice.

ALGORITHM

The problem we defined in the introduction can now be restated as follows:

BEST EXPLANATORS PROBLEM

Given: A set of sequences S , a set X of motifs in S and a real number τ .

Find: The smallest subset E of X such that for all $x \in X$, $Z(x|E) < \tau$.

We propose a greedy algorithm to solve this problem. The algorithm begins by choosing the motif with the highest (unconditional) z-score as the first element of E . Then it adds motifs to E iteratively. In each iteration, the motif that is *least* explained by the current E is chosen. The process stops when each motif in X is sufficiently well explained. Clearly, the set E output is a feasible solution. However, it may not be the smallest such set. In the next section, we test the algorithm on simulated data, and the results suggest that it performs remarkably well.

To reduce the running time, we use a preprocessing step that reduces the size of X by removing elements that are almost surely not going to be in the optimal set E . We do so by removing from X any motif m for which there exists a motif $e \in X$ such that $Z(e) > Z(m)$ and $Z(m|e) < \tau'$. (τ' is a threshold that, in practice, can safely be set to 4.) The preprocessing step dramatically reduces the running time of the greedy algorithm by typically filtering out about 80% of the motifs in X , while removing real motifs extremely rarely.

Our implementation uses the motif-finding tool of Sinha and Tompa (2000), called YMF, to generate the initial set X of all motifs with unconditional z-score at least 5. X is used as input to our program. YMF is a tool that uses an enumerative algorithm to find all motifs with high unconditional z-scores. It assumes a restricted alphabet $\Sigma = \{A, C, G, T, R, Y, W, S\}^\dagger$ and allows motifs of the form $\Sigma^a N^b \Sigma^c$, with $|a - c| \leq 1$ and $0 \leq b \leq 11$. We configured the program to allow at most 2 characters from the set $\{R, Y, W, S\}$ in a motif, and set $a = c = 3$. YMF models random sequences using a 3rd order Markov chain and our algorithm uses the same Markov chain. The

[†]an empirical study in Sinha and Tompa (2000) shows that other IUPAC symbols rarely occur in yeast binding site consensi

constant c_1 in Equation 4 is set to 23, so that any two motif occurrences (of length at most 17) with a gap of less than 6 bases between them are handled accurately. The constants c_2 in Equation 4 and c_3 in Equation 6 are set to the values 1 and 6 respectively.

The algorithm was implemented in C++ and the code is available from the authors upon request. Using the above-mentioned constants and the preprocessing step, it generally runs in less than 10 minutes on a PentiumIII machine, on inputs like those used in the next two sections (typically, 50 sequences of length 800bp each, $|X| = 1000$, $|E| = 5$).

VALIDATION EXPERIMENTS

In this section, we evaluate the ability of our algorithm to recover multiple significant and distinct motifs from a set of sequences. We first measure the accuracy of our algorithm by running it on random sequences in which a known set of 5 motifs was planted, and seeing if it recovers them as the top five explainers. The planted motifs are chosen from a set P which contains 18 motifs corresponding to known binding sites from SCPD (Zhu and Zhang (1999)) and fitting the YMF model. Each simulated data set is characterized by two parameters: n , the number of sequences, and $\{z_1, z_2, \dots, z_5\}$, the z-scores for each of the five planted motifs. Each data set is generated as follows: (i) Create n random sequences of length 800, using the third order Markov model for yeast. (ii) Choose five motifs m_1, m_2, \dots, m_5 at random from P and plant motif m_i k_i times (where k_i is chosen so that the unconditional z-score of m_i will be as close as possible to z_i), at random positions in the sequences, making sure not to overwrite previously planted motifs. If m_i contains IUPAC ambiguity codes, they are instantiated following the Markov model. Experiments are parameterized by z-scores rather than by the numbers of times motifs were planted, because two motifs can have very different significance even though they occur the same number of times in the input sequences. We then run the YMF motif-finder on the generated sequences, process its output using our algorithm and compare the five best explainers obtained to the planted motifs. Table 1 reports on the accuracy of the results for different numbers of sequences (5, 10 and 50), and z-scores of the different planted motifs. The algorithm is found to be very accurate: for example, for 50 sequences and motifs planted with z-scores varying between 20 and 12 (first row of the table), 46 of the 50 experiments yielded the *exact* set of five planted motifs. In the four other experiments, one of the five explainers reported was erroneous, but only because one of its characters was a generalization or specialization of the planted motif. Notice that even though the weakest planted motif is often ranked worse than 500th by YMF, our

| Nb. seq. | Z-scores planted ^(a) | Nb. errors ^(b) | Err. magnitude ^(c) | Nb. failures ^(d) | 5th motif rank ^(e) |
|----------|---------------------------------|---------------------------|-------------------------------|-----------------------------|-------------------------------|
| 50 | 20,18,16,14,12 | 0.08 | 0.72 | 0 | 57 |
| | 16,14,12,10,8 | 0.38 | 0.56 | 0 | 483 |
| | 14,12,10,8,6 | 0.4 | 0.85 | 0 | 686 |
| 10 | 20,18,16,14,12 | 0.30 | 0.70 | 0 | 93 |
| | 16,14,12,10,8 | 0.85 | 0.59 | 0.3 | 262 |
| | 14,12,10,8,6 | 1.31 | 1.17 | 0.56 | 604 |
| 5 | 20,18,16,14,12 | 0.5 | 0.55 | 0.1 | 62 |
| | 16,14,12,10,8 | 1.45 | 0.78 | 0.55 | 174 |
| | 14,12,10,8,6 | 1.94 | 1.50 | 0.75 | 544 |

Table 1. Results obtained on synthetic data with 5 planted motifs m_1, \dots, m_5 , averaged over 50 experiments. (a) Unconditional z-scores of the five planted motifs. (b) Number of *errors* among the 5 explanators e_1, \dots, e_5 : for each planted motif m_i , count one error if $m_i \neq e_j \forall j$. (c) Error magnitude, defined as $\min_{j=1 \dots 5} GSD(m_i, e_j)$, where $GSD(m_i, e_j)$ is the minimum number of generalizations or specializations needed to transform m_i into e_j . Count one generalization for $\{A, C, G, T\} \rightarrow N$, and 0.5 generalization for all other legal generalizations. Specializations are counted similarly. m_i and e_j might need to be padded with N's at their extremities. (d) Number of *failures*: for each motif m_i , if $\min_{j=1 \dots 5} GSD(m_i, e_j) \geq 3$, m_i has no clear homolog, so we count this as a *failure* instead of an error, and we disregard the magnitude. (e) Number of motifs output by YMF with unconditional z-scores higher than that of m_5 .

| Motif | Z-score | Comment | Rank in original list |
|------------------|---------|------------------------------------|-----------------------|
| CGGNNNNNNNNNNCCG | 13.0603 | Exact GAL consensus | 1 |
| CGYGYG | 7.33919 | Similar to PHO consensus | 36 |
| CTYATC | 6.99068 | Exact NIT consensus | 65 |
| CCGNNGGA | 6.82371 | Exact PDR consensus | 52 |
| CCGNNNNNNNNNNYGG | 6.1258 | Part of PHO4 binding site in PHO84 | 51 |
| AAGNNNNNNNNNNRWA | 5.97038 | Part of PHO4 binding site in PHO84 | 279 |
| CGTNNNNNNNYGA | 5.83989 | Exact ABF1 consensus | 195 |

Table 2. The seven best explanators found in a gene set formed by merging the GAL, PHO, NIT, PDR and ABF1 clusters.

method almost always extracts it correctly as a real motif.

The numbers in Table 1 behave as expected: the accuracy decreases when fewer sequences are available, because the assumption about the normality of the conditional z-scores begins to fail. Nonetheless, even for only five input sequences, the set of five explanators we report contains, on average, only about 0.5 motif that is clearly incorrect. The accuracy also decreases when the z-score of the motifs planted goes down. In fact, for five sequences containing no planted motif at all, the best motif is expected to have a z-score of about 6.5 (data not shown), so we can not expect to recover planted motifs with z-scores near this threshold without introducing false positives. Indeed, a vast majority of the failures observed for experiments with 5 and 10 sequences correspond to motifs planted with such low z-scores.

The second type of validation experiments used real biological data in an artificial scenario. We considered 5 arbitrary gene families for which binding sites are known (from van Helden et al. (1998), NIT (7 genes), PHO (5 genes), PDR (7 genes), GAL (6 genes) and from Zhu

and Zhang (1999), ABF1 (19 genes)). We merged the 5 families into one large group of 44 genes, and ran our program. The explanators reported are presented in Table 2. Among the 7 explanators the algorithm produced, 4 were exactly the known consensus of one of the gene families and the three others were parts of the longer PHO4 binding sites. Notice that the ABF1 consensus was ranked 195th in the list produced by YMF, yet it was correctly reported as one of the seven real motifs. These results are very encouraging because the data set represents a typical application scenario, where several unknown regulons are grouped into a loose cluster of genes, given as input. In such cases, our program should be able to give an accurate description of several binding sites.

RESULTS ON GENE CLUSTERS

We now report on the results obtained on sets of *Saccharomyces cerevisiae* genes that either have similar expression patterns or are functionally related, or both. Table 3 lists the sets of genes on which the algorithm was run, and

| FAMILY | GENES | ORIGIN |
|--------|--|---|
| MET | met1, met14, met19, met2, met25, met3, met30, met6, mup3, sam1, sam2 | S(MET) |
| GAL | gal1, gal2, gal7, gal80, gey1 | S(GAL) |
| CLN2 | 27 genes | S(CLN2) |
| MCM | 38 genes | S(MCM) |
| NSU | 37 genes | M(Nitrogen/sulphur utilization) |
| DEO | ttr1, sml1, rnr1, rnr2, rnr4, rnr3, ybr014c, pac1, ydl010w, trr2, trr1, cdc21 | M(deoxyribonucleotide metabolism) |
| NUT | pet9, aac3, aac1, fcy2, fcy21, odc2, ypr011c, ygr096w, yhr002w, ygl186c, yor071c, fcy22, yor192c | M(nucleotide transport) |
| PHO | dic1, pho84, pho86, pho88, pho87, pho89, mir1, ynr013c, yer053c, yjl198w | M(phosphate transport) |
| LIT | pxa2, acb1, bio5, yat1, git1, pxa1, faa2, faa4, snq2, itr1, itr2, pdr5, pdi1, yer024w, ykl1174c, ybt1 | M(lipid and fatty-acid transport) |
| PPP | gnd1, rki1, rpe1, zwf1, gnd2, ygr043c, tal1, tk11, tk12 | M(pentose-phosphate pathway) |
| TRI | kgd1, kgd2, aco1, cit1, cit3, lpd1, fum1, idh1, idh2, idp1, idp2, mdh1, osm1, yel047c, yjl200c, ylr164w, ymr118c, yjl045w, sdh1, sdh2, sdh4, lsc1, lsc2 | M(tricarboxylic-acid pathway) |
| GLY | aco1, cit2, icl1, mdh2, mls1, icl2 | M(glyoxylate cycle) |
| ION | cch1, aqy2, tok1, aqy1, yll053c, por2 | M(ion channels) |
| ATE | tat1, can1, agp3, agp2, agp1, bio5, hnm1, dip5, uga4, gap1, mup1, sam3, mmp1, tat2, gnp1, alp1, hip1, bap2, lyp1, ort1, put4, ykl174c, yor071c, bap3, mup3 | M(amino-acid transporters) |
| DEA | sgs1, ras2, ras1, lag2, nca3, uth1, lag1, lac1, sir4, bck2 | M(cell death) |
| HOP | pho84, pho86, pho87, pho89, mir1, ynr013c, yjl198w | M(homeostasis of phosphate) |
| CEN | 31 genes | M(organization of centrosome) |
| GLU | fba1, tdh2, pgk1, gpm1, tpi1 | M(glycolysis and gluconeogenesis) $\cap T(C_1)$ |
| ENE | ath1, ts11, sip2, hsp82, pgm2, pig1 | M(metabolism of energy reserves) $\cap T(C_5)$ |
| GRO | plc1, yck2, nhp6a, ymr263w, tpm1 | M(cell growth) $\cap T(C_6)$ |
| RIB | 63 genes | M(ribosomal proteins) $\cap T(C_1)$ |
| TRA | tef2, etf2, tif3, cdc33, hyp2 | M(translation) $\cap T(C_1)$ |
| REP | ogg1, rad5, rad27, cdc9, rad51, hys2, cdc2, pol32, rdh54, rad53 | M(DNA repair) $\cap T(C_2)$ |
| DET | enb1, adh5, pad1, pdr5, sit1, yil121w, yor273c, cad1 | M(detoxification) $\cap T(C_4)$ |

Table 3. Sets of yeast genes on which our program was run. The 800bp region upstream of the transcription start site of each gene was used. Origin of gene sets - S(): set of coexpressed genes reported by Spellman et al. (1998). M() : Genes from the given MIPS functional categories (Mewes et al. (1999)). $T(C_i)$: i^{th} gene cluster reported by Tavazoie et al. (1999)

Table 4 contains all motifs reported by our program (with $\tau=7$) for each set. The first four sets (MET, GAL, CLN2 and MCM) each contain coexpressed genes for which the binding sites are well studied (Zhu and Zhang (1999), Spellman et al. (1998)). For three of these four sets, each of the two explanators reported corresponds to a known binding sites. For MET, one of the two was a known motif. Notice that for CLN2, we identify the binding sites of both MCB and SCB as being distinct motifs, despite the fact that their sequences have a five nucleotide overlap. We consider these results as strong validation of our algorithm.

The next groups of sequences come from the functional classification of yeast genes in the MIPS database (Mewes et al. (1999)). We selected 28 functional classes containing between 5 and 40 genes. We ran our algorithm on all 28 sets. The 14 sets for which significant motifs were found are listed. Finally, in an attempt to benefit from both the functional classification and the expression array data, we considered the following genes sets: Tavazoie et al. (1999) report 30 gene clusters C_1, \dots, C_{30} obtained from the expression data reported in (Cho et al. (1998)) across two cell cycles. We considered all 86 MIPS classes,

M_1, \dots, M_{86} , having between 20 and 200 genes. For each such class M_i , we find the cluster C_j such that $\alpha = |M_i \cap C_j|$ is maximized. If $\alpha \geq 5$, we ran the algorithm on $M_i \cap C_j$.

Several gene sets appear to contain more than one significant motif. Many of these correspond to known binding sites in some of the genes of the set. However, several motifs that have very high z-scores are not listed in TRANSFAC (Wingender et al. (1996)) or SCPD (Zhu and Zhang (1999)). These motifs are good candidates for being novel binding sites.

DISCUSSION

An important assumption in our approach is the comparability of conditional z-scores of different motifs. We have noted previously that for sufficiently large sequence lengths, conditional z-scores are observed to follow the standard normal distribution, which means that, in random sequences, the distribution of $Z(m|E)$ is the same regardless of what m is. For relatively small sequence lengths (or, alternately, fewer sequences of same length), say 4000 bases, this assumption is not valid. On

| FAMILY | MOTIF | Z-SCORE | REFERENCES |
|--------|-------------------|---------|--|
| MET | CACGTG | 13.571 | Cbf1p-Met4p-Met28p complex binding site (van Helden et al. (1998)) |
| | CACNNNNNNNNNNNCAC | 7.63831 | |
| GAL | CGGNNNNNNNNNNNCCG | 27.8816 | GAL4 binding site (Zhu and Zhang (1999)) Part of most GAL4 binding sites (Wingender et al. (1996)) |
| | CGSNNNNNCTS | 7.30795 | |
| CLN2 | ACGCGW | 27.6889 | Binds MCB (Zhu and Zhang (1999)) Binds SCB (Zhu and Zhang (1999)) |
| | CRCGAAA | 7.99284 | |
| MCM | CCYNNNNNGGA | 9.35708 | Part of Mcm1p binding site (Spellman et al. (1998)) Part of Mcm1p binding site (Spellman et al. (1998)) |
| | AAANRGG | 7.91772 | |
| NSU | CTTATC | 9.68912 | Bound by GATA, DAL80 in DAL3 (Wingender et al. (1996)) |
| DEO | ACGCGT | 11.3229 | Binds MCB in CDC21 (Zhu and Zhang (1999)) Binds MCB in CDC21 (Zhu and Zhang (1999)) |
| | CCRNGGC | 8.30409 | |
| NUT | CGYNNNCRC | 8.42115 | |
| PHO | CGGNNNNGSS | 9.2287 | Binds PHO4 (Zhu and Zhang (1999)) Binds PDR1;PDR3 in SNQ2, PDR5 (Zhu and Zhang (1999)) |
| | SCACGTGS | 8.52335 | |
| LIT | CCGNSGR | 9.84957 | |
| PPP | GGGNNNNNGGR | 7.72371 | |
| TRI | CCGANNNNCGS | 10.332 | |
| | CGCNNNNNGCG | 7.07833 | |
| GLY | CCGNNNNNSSG | 11.6583 | |
| | CTWNNNNNGC | 7.18257 | |
| | CGGNNNNNNNSCG | 7.01786 | |
| ION | GCGNCSY | 7.42118 | |
| ATE | CGSNNNNNCSG | 10.6879 | Binds UAS GABA in UGA4 (Zhu and Zhang (1999)) |
| | RCGGCR | 8.71892 | |
| DEA | CTAGRC | 8.47908 | |
| | GSCNNNNCCS | 7.64476 | |
| | AGCNNNNNNASS | 7.24539 | |
| HOP | ACGTGS | 9.51188 | Binds PHO4 in PHO84 (Zhu and Zhang (1999)) |
| | TCGNNNNNNNSCR | 8.28188 | |
| | CACNNNNNNNNNNNGAC | 7.56527 | |
| CEN | AACNNNNACA | 7.778 | |
| GLU | CWCACA | 8.39103 | |
| ENE | CSSNNNNNNNNNCCC | 14.477 | |
| | CCSSNNCCCC | 10.3551 | |
| GRO | CGANNNNNNNNNCGA | 8.53649 | |
| | ACANNNNNNNNNWCA | 7.11369 | |
| RIB | GTANGGR | 12.4595 | Part of Rap1 binding site (Zhu and Zhang (1999)) Part of Rap1 binding site (Zhu and Zhang (1999)) Part of Rap1 binding site (Zhu and Zhang (1999)) |
| | AYCNNNACA | 9.6257 | |
| | ATGNNYGG | 8.0773 | |
| TRA | CCNNNCCS | 10.4008 | Part of Rap1 binding site in TEF2 (Zhu and Zhang (1999)) |
| REP | ACGCGT | 11.9047 | Binds MCB in CDC9 (Zhu and Zhang (1999)) |
| DET | CGGNGTC | 8.31184 | |

Table 4. Best explanators found in each set of genes from Table 3, using $\tau = 7$. The z-scores reported are the unconditional ones. References are given for each motif that corresponds to a known binding site in at least one of the genes of the set.

the other hand, our tests (see the section on validation experiments) suggest that the algorithm performs very well even on such scale, hinting that the effect of non-normality is too weak to cause the algorithm to fail. For shorter sequences, one possibility worth investigating is to approximate conditional word counts using binomial or Poisson distributions.

Often, enumerative motif-finders, such as those described in van Helden et al. (1998), Tompa (1999), and Sinha and Tompa (2000), are only able to deal with relatively small motif lengths. However, the actual binding

site m may be of longer length. In such cases, it is likely that two or more shorter *fragments* m_1, m_2, \dots, m_k of the longer motif are over-represented. Notice that m should explain well the occurrences of m_1, m_2, \dots, m_k . This provides us with an elegant way to decide when two overlapping motifs can be assembled into a longer one.

The primary reason to use the greedy algorithm to solve the BEST EXPLANATORS problem is efficiency. We also implemented and tested a Gibbs sampling strategy to solve this problem. The accuracy of the solutions obtained was comparable to that of our greedy approach, but its running

time was substantially worse.

The technique of motif explanation can be seen as augmenting the background model with a set of known motifs. This is of particular interest when the Markov model of the background does not capture some known ubiquitous patterns, like poly-ATs or repeats.

CONCLUSION AND FUTURE WORK

The typical output of many motif-finding techniques is a long list of motifs containing some real motifs (those that are the most likely to correspond to actual binding sites) but also a very large number of their artifacts (random variations of the real motifs). This paper presents a systematic method to separate the real motifs from their artifacts, thus producing a short list of very high quality motifs. Using synthetic data sets, we show that the output of our program is very accurate. When run on biological sequences, the program (in conjunction with YMF) identifies several known binding sites, as well as a number of significant motifs that are not documented.

Calculating conditional z-scores is fairly computationally intensive. We make some approximations to ease this computation but more or better approximations could significantly speed up these calculations.

In this paper, we assumed that motifs were modeled by consensus strings. The problem we addressed also exists when motifs are represented by weight matrices. A possible approach in this case is the following: Given weight matrices w_1, w_2, \dots, w_k of k explanators, and the background distribution w_0 , we may model background sequences using a Hidden Markov Model based on w_0, w_1, \dots, w_k , similar to that in meta-MEME (Grundy et al. (1997)). We may then compute the log-likelihood ratio of a new motif using this null model.

ACKNOWLEDGMENTS

We are extremely grateful to our advisor Martin Tompa for his insightful guidance throughout this project. We are also thankful to Emily Rocke and Jeremy Buhler for useful comments and discussions. We also thank the paper reviewers for their suggestions.

This material is based upon work supported in part by an NSERC fellowship, by the National Science Foundation and DARPA under grant DBI-9601046 and by the National Science Foundation under grant DBI-9974498.

REFERENCES

Bailey, T. L. and C. Elkan (1995, October). Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Machine Learning* 21(1-2), 51–80.

Cho, R., M. Campbell, E. Winzler, L. Steinmetz, A. Conway, L. Wodicka, T. Wolfsberg, A. Gabrielian, D. Landsman, D. Lockhart, and R. Davis (1998, July). A genome-wide transcriptional analysis of the mitotic cell cycle. *Molecular Cell* 2(1), 65–73.

Grundy, W. N., T. L. Bailey, C. P. Elkan, and M. E. Baker (1997). Meta-meme: Motif-based hidden markov models of protein families. *Computer Applications in the Biosciences* 13(4), 397–406.

Lawrence, C. E., S. F. Altschul, M. S. Boguski, J. S. Liu, A. F. Neuwald, and J. C. Wootton (1993, 8 October). Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. *Science* 262, 208–214.

Mewes, H., K. Heumann, A. Kaps, K. Mayer, F. Pfeiffer, S. Stocker, and D. Frishman (1999). MIPS: a database for protein sequences and complete genomes. *Nucleic Acids Research* 27, 44–48.

Nicodème, P., B. Salvy, and P. Flajolet (1999, January). Motif statistics. Technical Report RR-3606, INRIA Rocquencourt.

Rocke, E. and M. Tompa (1998, March). An algorithm for finding novel gapped motifs in DNA sequences. In *RECOMB98: Proceedings of the Second Annual International Conference on Computational Molecular Biology*, New York, NY, pp. 228–233.

Sinha, S. and M. Tompa (2000, August). A statistical method for finding transcription factor binding sites. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*. AAAI Press.

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

Tavazoie, S., J. D. Hughes, M. J. Campbell, R. J. Cho, and G. M. Church (1999, July). Systematic determination of genetic network architecture. *Nature Genetics* 22, 281–285.

Tompa, M. (1999, August). An exact method for finding short motifs in sequences, with application to the ribosome binding site problem. In *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology*, Heidelberg, Germany, pp. 262–271. AAAI Press.

van Helden, J., B. André, and J. Collado-Vides (1998, Sept. 4). Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. *Journal of Molecular Biology* 281(5), 827–842.

van Helden, J., M. del Olmo, and J. E. Pérez-Ortín (2000). Statistical analysis of yeast genomic downstream sequences reveals putative polyadenylation signals. *Nucleic Acids Research* 28(4).

Vilo, J., A. Brazma, I. Jonassen, A. Robinson, and E. Ukkonen (2000, August). Mining for putative regulatory elements in the yeast genome using gene expression data. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, San Diego, CA, pp. 384–394. AAAI Press.

Waterman, M. S. (1995). *Introduction to Computational Biology*. Chapman & Hall.

Wingender, E., P. Dietze, H. Karas, and R. Knüppel (1996). TRANSFAC: a database on transcription factors and their DNA binding sites. *Nucleic Acids Research* 24(1), 238–241. <http://transfac.gbf-braunschweig.de/TRANSFAC/>.

Zhu, J. and M. Q. Zhang (1999, July/August). SCPD: a promoter database of the yeast *Saccharomyces cerevisiae*. *Bioinformatics* 15(7/8), 563–577. <http://cgsigma.cshl.org/jian/>.