*5730–5739* *Nucleic Acids Research, 2006, Vol. 34, No. 20* *Published online 13 October 2006*
*doi:10.1093/nar/gkl585*

# A graph-based motif detection algorithm models complex nucleotide dependencies in transcription factor binding sites

**Brian T. Naughton*, Eugene Fratkin[1,]*, Serafim Batzoglou[1] and Douglas L. Brutlag**

Department of Biochemistry and [1]Department of Computer Science, Stanford University, CA 94305, USA

Received June 8, 2006; Revised July 25, 2006; Accepted July 27, 2006

## ABSTRACT

**Given a set of known binding sites for a specific transcription factor, it is possible to build a model of the transcription factor binding site, usually called a motif model, and use this model to search for other sites that bind the same transcription factor. Typically, this search is performed using a position-specific scoring matrix (PSSM), also known as a position weight matrix. In this paper we analyze a set of eukaryotic transcription factor binding sites and show that there is extensive clustering of similar *k*-mers in eukaryotic motifs, owing to both functional and evolutionary constraints. The apparent limitations of probabilistic models in representing complex nucleotide dependencies lead us to a graph-based representation of motifs. When deciding whether a candidate *k*-mer is part of a motif or not, we base our decision not on how well the *k*-mer conforms to a model of the motif as a whole, but how similar it is to specific, known *k*-mers in the motif. We elucidate the reasons why we expect graph-based methods to perform well on motif data. Our MotifScan algorithm shows greatly improved performance over the prevalent PSSM-based method for the detection of eukaryotic motifs.**

## INTRODUCTION

A transcription factor is a protein that binds to a transcription factor binding site, and, in doing so, regulates the expression of a nearby gene. Transcription factor binding sites (TFBS) are short DNA sequences (usually between 5 and 20 bp) that generally conform to a consensus but individually may exhibit considerable variability. These short sequences are also called DNA motifs. In this paper, we will refer to the models representing a set of transcription factor binding sites as motifs, and the individual *k*-mers that comprise them as motif occurrences. Finding these binding sites is an important biological problem stymied by the lack of practical experimental methods to assay transcription factor binding under a wide variety of conditions (1). The development of more accurate computational methods for detecting these binding sites would lead to deeper insights into complex biological processes such as development, differentiation, and oncogenesis.

To date, most computational biologists working in the area of DNA motifs have focused on the problem of *de novo* motif detection. This problem can be summarized as follows: given a set of DNA sequences that are thought to utilize a common binding site, identify these sites by searching for a set of similar *k*-mers that are statistically over-represented in these sequences. The canonical example of this problem is searching for transcription factor binding sites in the promoters of co-expressed genes. This problem is often referred to as the motif-finding problem. This problem has been addressed dozens of times, and has led to the development of numerous algorithms, including MEME (2) AlignAce (3), Weeder (4) and BioProspector (5). We have developed a *de novo* motif-finding algorithm based on similar principles to this manuscript (6).

An equally important, but relatively unappreciated problem, is the detection of new occurrences of known motifs. This problem is crucial to many biologists. Whenever a novel gene or transcript is detected, one of the first questions that will be asked is whether there are known binding sites in its promoter. Thanks to the recent development of tiling arrays (7) and other high-throughput technologies, we are on the verge of detecting huge numbers of new transcripts in all eukaryotic model organisms, yet sensitive methods for detecting the binding sites that control the expression of these transcripts are lacking.

Typically, methods tackling this problem have focused on weight matrix-based methods, or more complex probabilistic models that also model nucleotide dependencies. In this paper, we analyze the distribution of *k*-mers in a large number of eukaryotic motifs, and find complex dependencies that are not easily modeled probabilistically. We show that the inherent complexities in *k*-mer distribution are due to both

*To whom correspondence should be addressed. Tel: 650 723 5976; Fax: 650 723 6783; Email: briannau@stanford.edu
*Correspondence may also be addressed to Eugene Fratkin. Email: fratkin@cs.stanford.edu

functional and evolutionary effects, and leverage this analysis to develop a novel, graph-based algorithm for the detection of transcription factor binding sites.

### The position-specific scoring matrix

By far the most common representation of DNA motifs is the position-specific scoring matrix (PSSM), also known as a position weight matrix (PWM). Stormo describes the PSSM in detail (8). In a PSSM, the motif is of fixed size. Each position in themotif has four associated probabilities: the probability of an A, G, C and T at that position. The positions are assumed to be independent, so a PSSM defines a product multinomial model: one can calculate a score for a *k*-mer given a motif model by simply multiplying the probabilities of each nucleotide in the *k*-mer at each position (or, more typically, summing the logs of the probabilities). The most popular motif scanning method, TESS (9), uses a PSSM representation of motifs.

### Modeling nucleotide dependencies in motifs

Though the PSSM assumes the independence of positions in the motif, it has been noted several times that some motifs exhibit significant dependencies between positions (10–12). There are two principal ways that nucleotide dependencies could emerge. First, a change in the sequence of a binding site could require a compensatory change elsewhere in the site, modifying its protein binding characteristics. Alternatively, if the occurrences of the motif are related to one another through evolution, and not created anew each time, then we expect to see a phylogeny of *k*-mers within the motif. These intramotif relationships could result in significant dependencies, even in the absence of selection pressure. There have been several attempts to use richer representations of the motif to model nucleotide dependencies between positions, some of which we describe below.

Barash *et al*. (13) developed a number of Bayesian network representations of varying complexity for modeling motifs, with the intent of relaxing the independence assumption of the PSSM, and described methods for learning these models. They developed an Expectation Maximization method that could learn the structure of the model while performing *de novo* motif finding. The authors showed that their expressive motif models improve upon the sensitivity and specificity of PSSM-based motif detection for simulated data and yeast regulatory sequences. As is typical with Bayesian networks, there is a crucial issue of how complex the network should be. If there are too many parameters, then the model requires a lot of data to train; too few, and the model falls short of its potential expressive power.

Zhou and Liu (12) described another, simpler extension to the PSSM model. In their formulation, they incorporate only pairs of non-overlapping correlated positions. They then learn the model using a Markov chain Monte Carlo method. By their estimation, 25% of experimentally verified motifs in the TRANSFAC database show statistically significant correlations between positions. They incorporated this method into a Gibbs sampling algorithm and showed improved performance in *de novo* motif finding.

Other analyses, such as that of Xing and Karp (14), have incorporated prior experimental data on the characteristics of certain transcription factor binding domain interactions into a Bayesian framework. Their MotifPrototyper models incorporate generic structural signatures from classes of transcription factor binding sites, which they can generalize to novel motifs belonging to the same family. This approach shows great promise, since most transcription factors use one of a small number of types of DNA binding domains.

Relying on complex, heavily parameterized models can be problematic, especially if their underlying assumptions do not hold. King and Roth (15) developed a simple non-parametric method for motif detection, based on the PSSM. Their method interpolates between a standard PSSM, based on all of the occurrences of the motif, and a mixture of PSSMs, where each occurrence of the motif is modeled by its own PSSM. Their model can identify members of the motif that would be outliers using a PSSM model. The authors note that this non-parametric framework can model arbitrary dependencies, albeit without describing what those dependencies are.

### Motivation for a graph-based method of motif detection

As motivation for the development of this algorithm, we offer an example of how a PSSM model can act contrary to our expectations, even for a very simple motif. This example applies analogously to more complex probabilistic models. We imagine a motif model comprised of three *k*-mers: AAA, AAA and AGG. Using a PSSM, a candidate *k*-mer, AGG, gets a score of $1.0 \times 0.33 \times 0.33$. A candidate AAG *k*-mer scores $1.0 \times 0.67 \times 0.33$. Therefore AAG scores higher than AGG, despite the fact that we have evidence that the protein binds AGG, but no such evidence exists for AAG. Using a probabilistic model, we cannot ensure that an exact match always scores higher than an inexact match. This is a desirable property if we believe the known *k*-mers from the motif are an unbiased sample of the true motif.

Many biologists would be surprised to learn that an exact match to a known transcription factor binding site may be missed when they use a PSSM to search a DNA sequence. We also note that the addition of new AGG *k*-mers to the model significantly reduces the score that a candidate AAA will achieve. Therefore, with a probabilistic model, these additional AGG *k*-mers act not only as evidence for AGG, but also as evidence against AAA.

We can think of the PSSM approach as measuring the distance of a candidate *k*-mer from the 'centroid' of the motif model. In the case of a mixture of PSSMs, it would be a weighted average distance between the candidate *k*-mer and a number of PSSM 'centroids'. However, a candidate *k*-mer that is close in sequence to a subset of *k*-mers in the motif, but not necessarily its 'centroid' is still likely to be a true member of the motif. For this reason, it makes more sense to evaluate a candidate *k*-mer based on its nearest neighbors, rather than a model of the entire motif.

## MATERIALS AND METHODS

### Calculating pairwise nucleotide dependencies in motifs

We gauged the degree of correlation between two columns in a motif alignment using chi-square values, and used a Monte Carlo simulation to estimate *P*-values.

The formula for the chi-square value is given below.

$$\chi^2 = (O-E)^2/E. \quad \mathbf{1}$$

For a pair of positions in a motif, the observed frequency of a pair of nucleotides is simply the number of times that pair occurs together in the *k*-mers of the motif. The expected frequency of a pair of nucleotides is the product of the frequency of each nucleotide in its respective column. For each motif in each dataset we calculated a chi-square value for every pair of positions, and noted the maximum chi-square value for that motif. We also generated a distribution of maximum chi-square values for this motif. We randomly permuted the columns of the motif alignment at each position, removing all positional dependencies, and calculated the maximum chi-square value for all pairs of positions. After one million iterations, we obtain a null distribution of maximum chi-square values, with which we can assign a *P*-value to our original motif. There are two reasons we did not directly use the chi-square *P*-value. First, there are a number of hypothesis tests being performed in each motif, one for each pair of positions, which must be corrected for. Furthermore, the accuracy of this *P*-value is in doubt if the number of motif members is small. By randomly generating a distribution of these minimum *P*-values for each motif, we ameliorate these problems.

Since we tested a number of motifs for significant dependencies, we also required a multiple hypothesis test correction. Here we used the R package q-value (16) and its default parameters to turn our *P*-values into q-values, and so control the false discovery rate (FDR).

### Nucleotide substitution rates

Given a *k*-mer from a motif, we calculated the prior probability that another randomly chosen *k*-mer from the motif will have a substitute nucleotide at each position. As an example, we consider a motif where the first column in the alignment has nine As and two Gs. For a *k*-mer in the motif with an A at this position, the probability that another *k*-mer in the motif chosen at random will have a G as a substitute is 20%. Here, we use the notation $P(G \mid A) = 0.2$; similarly, $P(A \mid A) = 0.8$, $P(G \mid G) = 0.1$ and $P(A \mid G) = 0.9$. These probabilities apply to one position in one motif, but we would like to know the expected probabilities over all positions in all motifs. For each position in each motif, we calculated the probability that nucleotide β was a substitute for nucleotide α. We weighted this probability by the probability that this nucleotide came from this position, given α. For example, if there is an A at this position, then it is more likely to have come from an A-rich position. The formula, along with the necessary normalization, is given in Equation 2 below.

For α, β, γ ∈{*A, C, G, T*}, and given *N* columns in total, then

$$E[P(\beta \mid \alpha)] = \frac{\sum_{i=0}^{N} P_i(\beta \mid \alpha) P_i(\alpha)}{\sum_{i=0}^{N} \sum_{\gamma} P_i(\gamma \mid \alpha) P_i(\alpha)} \quad \mathbf{2}$$

**Table 1.** Nucleotide substitution rates in motifs

Yeast motifs

| 1a | A | C | G | T |
|---|---|---|---|---|
| A | 0.785 | 0.058 | 0.083 | 0.085 |
| C | 0.054 | 0.793 | 0.066 | 0.077 |
| G | 0.077 | 0.066 | 0.793 | 0.054 |
| T | 0.085 | 0.083 | 0.058 | 0.785 |

| 1b | A | C | G | T |
|---|---|---|---|---|
| A | | 0.280 | 0.400 | 0.392 |
| C | 0.250 | | 0.320 | 0.357 |
| G | 0.360 | 0.320 | | 0.250 |
| T | 0.392 | 0.400 | 0.280 | |

JASPAR motifs

| 1c | A | C | G | T |
|---|---|---|---|---|
| A | 0.684 | 0.107 | 0.134 | 0.120 |
| C | 0.087 | 0.670 | 0.089 | 0.109 |
| G | 0.109 | 0.089 | 0.670 | 0.087 |
| T | 0.120 | 0.134 | 0.107 | 0.684 |

| 1d | A | C | G | T |
|---|---|---|---|---|
| A | | 0.324 | 0.405 | 0.379 |
| C | 0.276 | | 0.270 | 0.345 |
| G | 0.345 | 0.270 | | 0.276 |
| T | 0.379 | 0.405 | 0.324 | |

1a and 1c show substitution probabilities for yeast and JASPAR motifs. Each column in the matrix represents the probability that the nucleotide at the head of the column will have each of the other nucleotides as a substitute. 1b and 1d show the same probabilities, but conditioned on the fact that a substitution has occurred. This highlights the differences in substitution probabilities.

By averaging the values for all of the columns in all of the motif alignments, we obtain a 4 × 4 matrix, where each column in the matrix contains the probability that given a specific nucleotide in a motif, we will see each of the 4 nt at the same position in another *k*-mer in the motif. The matrices for yeast and JASPAR are given in Table 1.

*Training the algorithm on simulated motifs.* We trained the two parameters of our algorithm, $\Theta_{SS}$ and $\Theta_{IK}$, as follows. We randomly generated a set of 1000 motifs, where the information content and length of the motif is sampled from the averaged distributions from yeast and JASPAR-motifs. These simulated motifs had no specific dependencies introduced. We used a simple hill-climbing algorithm to train, and the same ROC curve as described in the Testing the Algorithm on Real Data as a measure of performance. Finally, we trained the same parameters on the yeast and JASPAR motifs individually. Gratifyingly, the parameter values were the same for all three experiments. For the three experiments, $\Theta_{SS}$ and $\Theta_{IK}$ came out to be 0.7 and 0.3, respectively. These values are robust to small changes in value. For fairness, the optimal number of pseudocounts used by the PSSM (the sole adjustable parameter of the PSSM) was also trained based on the same set of motifs and the same conditions.

## RESULTS

### The motif data set

The complete dataset we use consists of a set of yeast motifs, and a set of motifs from multicellular eukaryotes. The yeast data comes from a genome-wide study by Harbison *et al.* (17) We retrieved occurrences from 102 motifs identified in *Saccharomyces cerevisiae* as having a *P*-value of

<0.001 and as being conserved in at least one other yeast genome (http://jura.wi.mit.edu/fraenkel/download/release_v24/GFF). The authors discovered these motifs by analysis of ChIp–chip data with a number of different motif-finding algorithms, a literature review and comparative genomics. We also retrieved occurrences from 95 of 106 motifs from the JASPAR database (18) for which *k*-mers as well as weight matrices are available. The JASPAR database is a curated set of motifs from multicellular eukaryotes, 49 of which are human. For all of the following experiments, we use 63 yeast motifs and 90 JASPAR motifs that contain 10 or more *k*-mers and >1 unique *k*-mer.

*Complex nucleotide dependencies in motifs*. In this section we analyze the distribution of *k*-mers in eukaryotic motifs, and show that there are extensive nucleotide dependencies, which we identify as clusters of *k*-mers in a graphical representation of the motif. This clustering makes motifs inherently difficult to model probabilistically. Specifically, we demonstrate how the PSSM falls short in modeling these motifs.

*Pairwise dependencies*: First, like Zhou and Liu (12), we calculated how many motifs have at least one pair of positions with a statistically significant nucleotide dependency (for more details see Materials and Methods). At a FDR of 5%, we find 18 out of 63 yeast motifs tested, or ∼29%, show significant pairwise dependencies. In the JASPAR database, we find 24 of 90 motifs, or ∼27%, with significant dependencies. These values are in agreement with Zhou and Liu's findings for TRANSFAC motifs (12,19).

Pairwise dependencies like these can be modeled fairly easily in a weight matrix model, by simply including the conditional probabilities at each correlated position. However, pairwise dependencies are not the only possible type of dependency. In order to analyze more complex dependencies, we use a graph-based representation of motifs. Here, *k*-mers are represented as nodes in the graph, and edges connect nodes if the Hamming distance (number of mutations) between the two *k*-mers is below a certain threshold. A pairwise dependency will make a cluster in the graph for each correlated nucleotide pair; other types of dependencies will result in different clustering patterns. For instance, a set of exact matches in the motif may not produce a significant pairwise dependency, but will appear as a clique in the graph. A combination of dependencies that individually may not be statistically significant may still alter the graph structure in identifiable ways. We call these dependencies due to *k*-mer clustering complex dependencies to differentiate them from pairwise dependencies.

*Complex dependencies*: The PSSM representation of the motif maintains the correct nucleotide distribution at each position in the motif but eliminates any other structural information. Therefore comparing real motif graphs with graphs generated using PSSMs serves two purposes: it gives us a baseline against which we can compare the structure of real motif graphs, and highlights areas where the PSSM fails to correctly model the motifs. For each motif in the dataset, we constructed a PSSM. We then randomly generated 10 000 motifs using this PSSM as a basis, each with the same number of *k*-mers as the original motif, and constructed a graph for each. This population of graphs is the benchmark against which we compare the true motif graphs.

To measure the overall amount of clustering in each graph we use the clustering coefficient (20). If we define the neighborhood of a vertex as the vertices directly connected to it, then the clustering coefficient of this vertex is simply the number of edges in its neighborhood, divided by the total number of possible edges in the neighborhood. The clustering coefficient for the graph is the average value for all vertices. The higher the clustering coefficient, the more clustered the graph is, and generally, the more dependencies there are in the motif. Duplicate *k*-mers are a common source of clustering in the graph, and will turn out to be useful in our experiments, therefore as another measure of clustering we count the number of *k*-mers in each motif that are duplicates of any other *k*-mer in the same motif.

We find that in yeast, the clustering coefficient of real motifs is greater than the average clustering coefficient in the PSSM-generated motifs in 32 cases, and fewer in 9 cases. The average clustering coefficient is ∼1.25 times greater in real motifs than in the PSSM-generated motifs. For the JASPAR motifs, the clustering coefficient is greater in real motifs in 33 cases, and fewer in 4 cases, and the average clustering coefficient is ∼1.5 times greater in real motifs than in the PSSM-generated motifs. Therefore there is significantly more clustering in real motifs than in the PSSM-generated motifs, and the difference is much more significant than a measure of pairwise dependencies alone would indicate.

The number of duplicate *k*-mers is greater in real yeast motifs than the average PSSM-generated motif in 40 cases, and fewer in only 6 cases. In JASPAR, the number of duplicate *k*-mers is greater in real motifs in 42 cases, and fewer in only 8 cases. The high number of duplicate *k*-mers will not come as a surprise to many biologists. However, when attempting to describe a *k*-mer distribution probabilistically, a group of duplicate *k*-mers will act like a spike of probability density for this particular sequence. Large deviations in probability density like this are inherently difficult to parameterize. It follows that if there are more duplicate *k*-mers in real motifs, then the number of unique *k*-mers is necessarily fewer in these motifs, so that the sequence of these motifs is more constrained than the nucleotide distribution alone indicates.

In sum, these experiments demonstrate that the distribution of *k*-mers in motifs deviates from the distribution implied by the nucleotide distribution in highly significant, and sometimes surprising ways. These complex dependencies manifest as clusters in the graph. For a given motif, it is difficult to predict how the distribution of *k*-mers will behave, and because of this, parameterizing these distributions is difficult. The wide variety of possible motif structures is evident in Figure 1.

## Motif clustering and gene expression

The non-random distribution of *k*-mers in motifs we observed, and especially the overabundance of duplicate *k*-mers, is evidence that there may be selection pressure acting to maintain particular motif sequences. If a motif sequence is specifically selected for, then it is possible that this sequence affects the expression of its associated genes

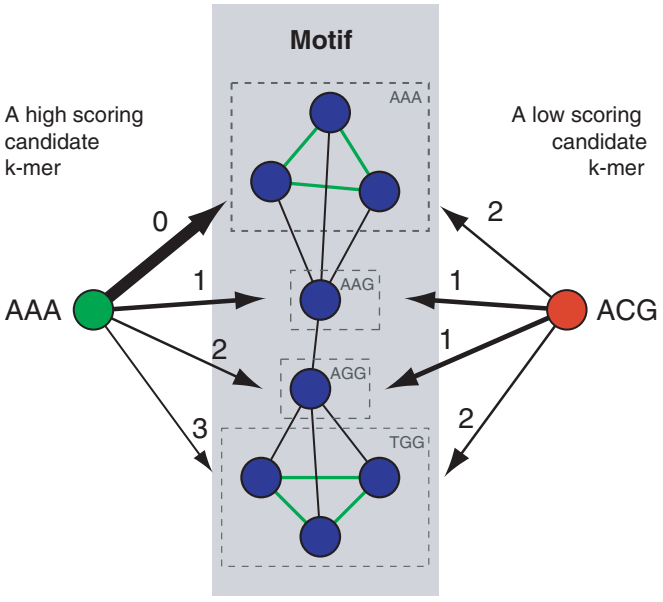


**Figure 1.** Graphs of four yeast motifs (**a**–**d**) and four JASPAR motifs (**e**–**h**) where MotifScan performed better than a PSSM. Each node represents a *k*-mer in the motif; gray edges represent a Hamming distance of one, green edges represent a Hamming distance of zero. The color of the nodes represents the ratio of the expected number of this *k*-mer in a motif randomly generated from a PSSM to the number in the real motif. A red node indicates that the PSSM generates this *k*-mer too frequently, a blue node indicates that it generates it too infrequently. If the PSSM were a perfect model of the motif, all nodes would be magenta (i.e. 50% red, 50% blue).

(21). If this is the case, then clusters in the motif graph correspond to clusters of gene expression.

To investigate this effect in more depth, we used three yeast microarray experiments downloaded from the Stanford Microarray Database (http://smd.stanford.edu), from the papers Segal *et al*. (22), Brauer *et al*. (23) and Wang *et al*. (24). Brauer *et al*. (23) measured gene expression in response to glucose limitation, Segal *et al*. (22) measured gene expression under a broad set of conditions, and Wang *et al*. (24) measured the rate of mRNA decay. We limited our investigation to motif occurrences that were present alone in the promoters of genes. Including promoters that contain several different motifs would make it difficult to infer expression changes due to the sequence of the individual motif. For each pair of *k*-mers in each motif that fulfill this criterion, we calculated the Hamming distance, and measured the co-expression of their respective genes using a Pearson correlation. Then, for each motif, we have a list of Hamming distances and the associated gene expression correlation.

As one might expect, the data are noisy, so to improve the specificity of these results we combined data from all three microarray experiments. We tested to see if there was a statistically significant correlation ($P < 0.01$) between Hamming distance and co-expression using Kendall's tau, a rank-based measure of correlation. We find that out of a total of 33 motifs for which there is sufficient data available, 7 have statistically significant negative correlations in all three microarray experiments. In other words, for these motifs, similar *k*-mers have similar expression patterns and clusters of *k*-mers in the motif graph correlate with clusters of gene expression. The seven motifs are YDR026c, DAL82, BAS1, SPT23, HAP4, ADR1 and MBP1. No motifs were foundto have statistically significant positive correlations in all three experiments. The most obvious example of clustering affecting gene expression is ADR1 (Figure 1b), which has two unconnected clusters.

The eventual goal of the analysis of transcription factor binding sites is to understand the expression of the genes that are under their control. Most current analyses of gene expression that utilize motifs note the presence or absence of motifs in the promoter of a gene, but do not pay heed to the actual sequence of the motifs. Here we have shown that the *k*-mer clusters we see in the motifs graphs are functionally relevant in some cases, perhaps gaining the cell finer control over gene expression.

## Motif clustering and DNA duplication

The clustering of *k*-mers with similar gene expression patterns explains a fraction of the clustering we see in yeast motifs. Here we show how evolution may also play a significant role. Clusters of *k*-mers could occur due to selection pressure and convergent evolution, as suggested by the gene expression example above; these clusters could also be due to *k*-mers being related through relatively recent duplication events.

To test whether *k*-mers in yeast motifs could be related through recent DNA duplications, we analyzed the conservation of nearby motif occurrences and the DNA flanking these occurrences. For each motif, we took every pair of *k*-mers located fewer than 100 nt apart (but not overlapping), and calculated the probability of these *k*-mers being identical, compared with *k*-mers >100 nt apart. We find that motif occurrences 100 bases or fewer apart are 25% more likely to be identical than those >100 bases apart, indicating that these cases might be due to tandem DNA duplications. However, this similarity may also be explained by variations in GC content.

We calculated the amount of conservation in the 10 nt flanking each side of the motif occurrences. We find that for a pair of identical *k*-mers fewer than 100 nt apart, the DNA flanking these *k*-mers is significantly more conserved than for a pair of non-identical *k*-mers from the same motif. The results of this experiment are shown in Figure 2. This data again suggest that some nearby, identical motif occurrences are related through duplication events. The alternative, that the flanking DNA has functional significance, requires that the conserved flanking positions be dependent on other positions in the motif. This is certainly possible, particularly for the nearest flanking positions (i.e. the motif is 1 nt longer than reported, and dependent on another nucleotide within the motif), but is more unlikely for positions further away.

Therefore we believe that clusters in the motif graphs emerge partly due to evolutionary effects, where some binding sites are the product of DNA duplication events that generate a phylogeny of *k*-mers in the graph. These relationships naturally leads to more clustering, and more exact matches in the graph. There is some evidence that short DNA duplications are common in eukaryotic genomes (25). In yeast, tandem duplications followed by chromosomal rearrangements have been suggested as a common evolutionary

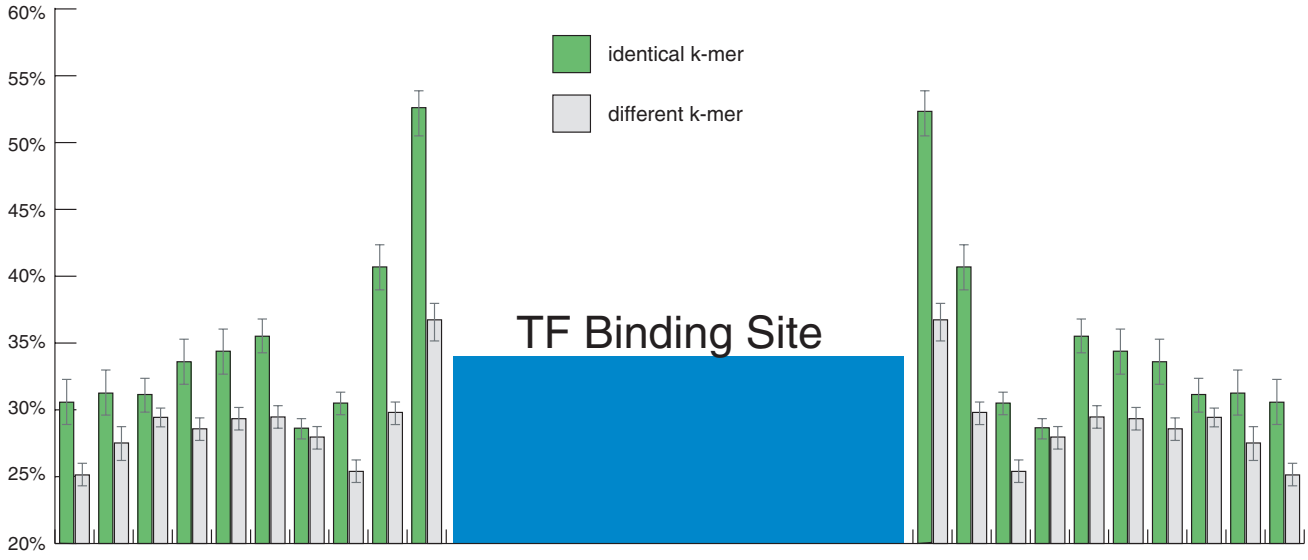


**Figure 2.** Conservation of flanking DNA for identical and different *k*-mers in yeast motifs. This graph shows the average conservation of flanking nucleotides for pairs of *k*-mers located within 100 bases of each other from all yeast motifs in *S.cerevisiae*. The length of *k*-mers varies from 5 to 19 nt. Nucleotides surrounding groups of identical *k*-mers are more likely to be identical than nucleotides surrounding groups of different *k*-mers from the same motif. The two sides of the binding site are symmetrical since transcription factor binding sites do not have 5′–3′ directionality. Note that given the background nucleotide distribution, we expect a value of ∼26% identity to indicate no conservation. Error bars represent 1 SD and were estimated with the bootstrap.

mechanism (26,27), and gene duplications are also common (28). However, unlike the evolution of genes, motif evolution need not rely on DNA duplications (29).

### Nucleotide substitution rates in motifs

Before we describe the algorithm itself, there is one final analysis that will prove important. One major advantage of a graph-based method over a probabilistic method for the motif detection problem is that because we retain all known motif occurrences in the model, we can identify the individual substitutions that would be required for the candidate *k*-mer to become a known motif *k*-mer. The rate of nucleotide substitution in motifs is significantly non-uniform, so this information becomes extremely useful for motif detection. Accordingly, we calculated empirical nucleotide substitution frequencies in eukaryotic motifs for use in our algorithm. Specifically, we calculated the likelihood of each nucleotide substituting for every other in each motif in the dataset. These frequencies are shown in Table 1. For details of how the table was generated see Materials and Methods.

Under neutral selection pressure, transitions are more common than transversions (30). We also see this trend in nucleotide substitution rates in motifs. This is not surprising since purines are biochemically similar to purines and pyrimidines to pyrimidines, which we expect to make a difference for protein binding. This effect is significant in yeast and JASPAR motifs, with some substitutions being up to 50% more likely than others. One might also expect that a nucleotide's complement would be a likely substitute since the base pair at that position would remain the same; however, unlike the transition–transversion effect, this is not always the case. We can leverage the data in these matrices in our algorithm. For instance, for a JASPAR motif comprised of CAA *k*-mers, we know that the prior probability that TAA is also a *k*-mer in the motif is 50% higher than GAA. In a weight matrix, this information is absent.

### The motif detection algorithm

Our method differs from those discussed in the introduction in a two key ways, which are justified by our analysis of transcription factor binding sites. First, our scheme takes a nearest-neighbor approach to motif detection. Similar to the *k*-nearest neighbors classification method, a candidate *k*-mer's score is based on a comparison with the most similar members of the motif, not on a model based on the motif as a whole. Therefore we do not abstract the *k*-mers of the motif to another representation, but retain all known motif members in our model. Second, we incorporate prior probabilities of nucleotide substitutions within motifs into our score. Our experiments have demonstrated significant differences in these probabilities.

*MotifScan's scoring system.* To score a candidate *k*-mer with our model, we first define what it means to be a good candidate *k*-mer for a motif, and outline a heuristic based on that. We describe three desiderata here:

(i) *Sequence Similarity (SS)*: The candidate should be close in sequence to at least one member of the motif. The fewer mutations it is from this matching *k*-mer, the higher the score should be.
(ii) *Identical k-mers (IK)*: The more copies of this matching *k*-mer there are in the motif, the higher the score should be. The more times a *k*-mer appears, the more likely it is that a closely related *k*-mer is also part of the motif. We refer to a number of duplicate copies of a *k*-mer in the motif as an 'identity group'.
(iii) *Identity group similarity (IGS)*: The greater the number of identity groups of *k*-mers in the motif similar to the

candidate, the higher the score should be. Each new identity group is evidence in favor of this candidate.

In total, there are two parameters to train, corresponding to the first two desiderata. The third desideratum is omitted, since the optimal value was found to be very close to unity. Each parameter is raised to an exponent, which depends on properties of the motif (see Equation 3 below). The first parameter, $\Theta_{SS}$, determines the relative score of zero mutations versus one, two or more mutations. We score successive numbers of mutations as $\Theta_{SS}^{d}$, where $d$ is the number of mutations. For example, if we choose $\Theta_{SS}$ to be 0.5, then a motif $k$-mer two mutations from the candidate $k$-mer contributes 0.25 ($0.5^2$) as much to the final score as an exact match. The second parameter, $\Theta_{IK}$, determines how much we value additional $k$-mers in an identity group. Here the exponent is the number of $k$-mers in the identity group. If we choose $\Theta_{IK}$ to be 0.5, then an identity group of two members contributes 1.5 times (0.5 + 0.25) as much to our score as an identity group of one member (0.5), and an identity group of three members contributes 1.75 times as much (0.5 + 0.25 + 0.125). For details on how these parameters were trained see Materials and Methods.

he $k$-mer with more likely substitutions. Then the score for the candidate is calculated according to Equation 3. A simple example in Figure 3 illustrates the process. The score for a candidate $k$-mer is calculated according to Equation 3. A simple example in Figure 3 illustrates the process.

$$\text{Score} = \sum_{i=1}^{N} \Theta_{\text{SS}}^{d} \Theta_{\text{NS}(b1,\,b2)} \sum_{j=1}^{n_i} \Theta_{\text{IK}}^{j}. \qquad \mathbf{3}$$

Here $N$ is the total number of unique $k$-mers, $d$ is the Hamming distance and $n_i$ is the number of $k$-mers in identity group $i$. $\Theta_{\text{NS}(b1,b2)}$ (the nucleotide substitution parameter) is the appropriate value from our nucleotide substitution matrix, where $b1$ (the nucleotide in the candidate $k$-mer) is substituting for $b2$ (the nucleotide in the motif $k$-mer). If $d$ is greater than 1, then $\Theta_{\text{NS}(b1,b2)}$ is the average of all of the substitution probabilities (given in Table 1, a and c).

*Pseudo-k-mers*. One advantage that probabilistic methods have over enumerative is the ability to generalize based on small sample sizes, owing to the inherently probabilistic nature of the model, and the addition of pseudocounts, which act as a regularizer (31). In a similar vein, we attempted to improve our algorithm by generating a small number of 'pseudo-$k$-mers' based on the distribution of nucleotides in our motif. The distribution of nucleotides itself uses a small pseudocount (0.1). These 'pseudo-$k$-mers' act just like pseudocounts in a PSSM, i.e. as regularizers to make small datasets 'smoother'. After cross-validation on simulated motifs, we found that the addition of pseudo-$k$-mers did not improve performance.

*Significance tests: P-values and the false discovery rate*. To gauge whether a particular score is significant, we need to transform the heuristic score into a $P$-value. Here we describe several ways of doing this. For each of the yeast motifs, we use the probes from Harbison *et al.*'s (17) ChIp–chip experiments in which the motif did not appear to generate a null distribution of scores, which we can use to assign a $P$-value to any given score. These $P$-values will be conservative since our assumption is that the sequences used to generate the null distribution do not contain any real motif

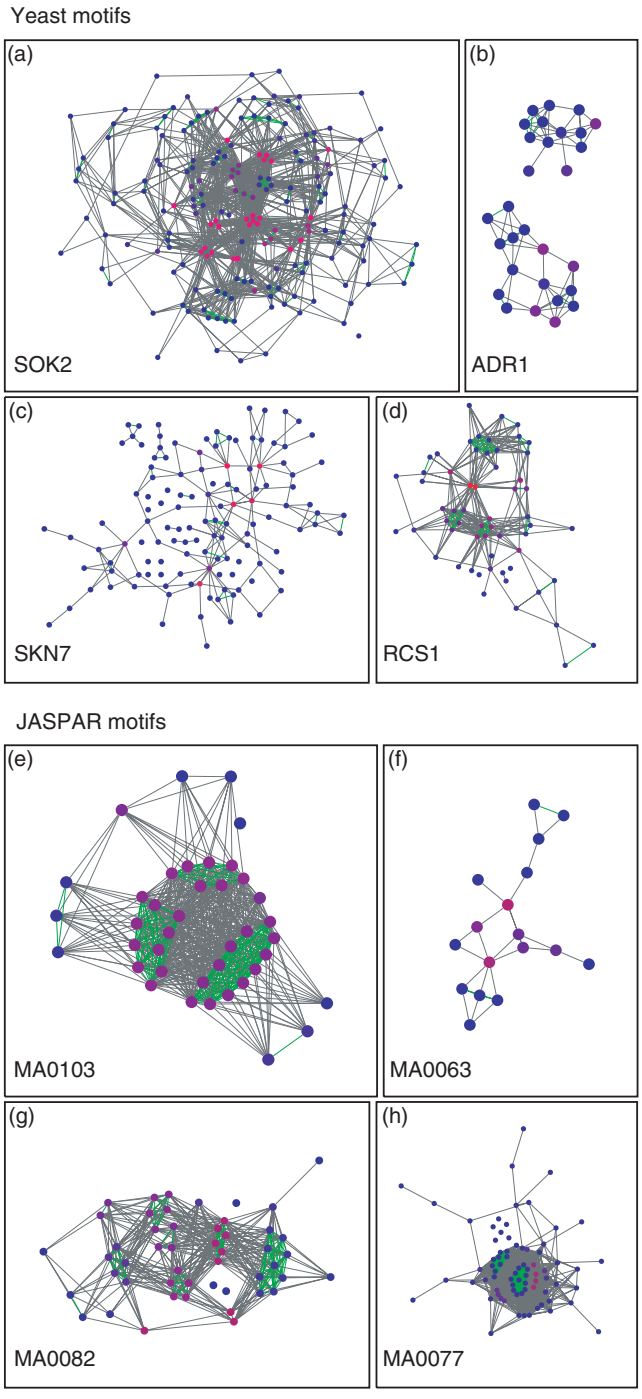


**Figure 3.** An example of the MotifScan algorithm assessing two candidate $k$-mers. Gray edges represent a Hamming distance of one, green edges represent exact matches. The thickness of the arrows represents the amount of weight the algorithm places on each match. The Hamming distance of each candidate-motif match is noted above the arrows. Even though for both candidates, the average Hamming distance to the motif is the same, the high scoring candidate is an exact match to a cluster of three $k$-mers, which contributes the majority of its score, whereas the low scoring candidate is two mutations away from each of the clusters of three $k$-mers.

occurrences. If they do, then the resulting *P*-values will be less significant.

For the JASPAR motifs, we use scores based on a null distribution from a uniformly random sequence of DNA. The resulting *P*-values are then corrected for based on the calculated GC content in the input DNA sequences, and their associated Markov dependencies. The length of the Markov chain depends on the length of the input sequences, with 20 data points required per parameter. This method is not nearly as reliable as using real data, therefore for more accurate results using these motifs, we advise using a precomputed table of Markov dependencies based on DNA similar to the input sequences (e.g. a collection of promoter DNA from the appropriate species). An even better option is to use real data to directly generate the *P*-value tables, as we did for yeast motifs. The appropriate background DNA to use is context-specific, depending on where in the genome one is searching.

We set an upper limit on the FDR of our results using the sequential method of Benjamini and Hochberg (32). Although the positive FDR (pFDR) method of Storey (33) is more powerful, Storey states that as $\pi_0$, the proportion of null hypotheses that are true, tends to 1, this method and Benjamini's methods become identical. In our case, we expect $\pi_0$ (the fraction of *k*-mers not in the motif) to be close to 1.

*Testing the algorithm on real data*. We evaluated the performance of our motif detection using leave-one-out cross-validation: (i) For every *k*-mer in every motif in the dataset, we generated a random DNA sequence 5000 bases long, and seeded this sequence with the *k*-mer. (ii) We built a PSSM model and a MotifScan model using all of the *k*-mers in the motif except the seeded *k*-mer and assigned scores to all of the positions in the sequence. Note that identical *k*-mers to that seeded may still exist in the motif. (iii) We sorted all of the resulting scores from all of the sequences from best to worst. As we traversed this list, we flagged each result as a true positive or a false positive.

To gauge performance, we used a modified version of a receiver operating characteristic curve (34) (ROC curve). The ROC curve is a plot of the true positive fraction (sensitivity) versus the true negative fraction (1-specificity). The area under this curve is a measure of performance over a range of sensitivities. A modification of the ROC curve, known as the $ROC^{50}$, is often used for database searching. This curve plots true positives against false positives, but only up to the first 50 false positives. The reason for doing this is that most of the ROC curve is uninformative, as there can be orders of magnitude more true negatives than true positives. We use an $ROC^N$ curve, which is very similar to an $ROC^{50}$, with the sole change that we use *N* as the number of allowed false positives instead of 50, where *N* is the total number of *k*-mers in the motif. This way the number of false positives allowed scales linearly with the number of true positives in the dataset.

We compared the performance of MotifScan and the PSSM by identifying motifs where one of the two methods significantly outperformed the other, and ignoring those motifs where the results were comparable. If the area under the ROC curve for a certain motif was 5% more, or greater, for one method than the other, this was counted as a win for that algorithm.

With the yeast dataset, we find that 25 of 63 motifs have an area under the ROC curve at least 5% greater with our algorithm than with a PSSM. Only 1 of 63 motifs has an area at least 5% greater with a PSSM than with our algorithm. With the JASPAR dataset, the figures are 46 of 90 and 8 of 90, respectively. A list of the most significantly different motifs in each database is given in Table 2. The greatest improvements in $ROC^N$ area for MotifScan over the PSSM were 69 and 74% for the yeast and JASPAR motifs, respectively. The greatest improvements in $ROC^N$ area for the PSSM over MotifScan were only 6 and 13%, respectively. An Excel file containing the complete list of ROC areas is available at http://motifscan.stanford.edu/roc_results.xls.

Figure 1 includes graphs of four yeast motifs and four JASPAR motifs where MotifScan performs well relative to the PSSM; specifically, for each of these graphs, the area under the ROC curve is at least 5% greater using MotifScan than using a PSSM. The graphs were chosen to be a representative subset of the complete set. There are a number of notable features in these graphs. In general, we can see that motifs have diverse structures, that clustering is prevalent, and that exact matches are common. In ADR1, we see two unconnected clusters, owing to a significant pairwise dependency. Accordingly, few of these *k*-mers are generated by the PSSM. MA0063, MA0082 and MA0103 have central regions where the mass of the PSSM is concentrated, and clusters of nodes at the peripheries that the PSSM rarely generates. MA0077, RCS1 and SOK2 all show significant numbers of subclusters and especially groups of exact matches. Some of these subclusters are generated by the PSSM, but many neighboring clusters are not. The yeast motif that MotifScan performs best on relative to the PSSM, SKN7, has a diffuse structure, with no obvious center. Almost none of the *k*-mers in this graph are generated sufficiently often by the PSSM.

*Using MotifScan*. The MotifScan algorithm can be accessed through a web-application, at http://motifscan.stanford.edu. The user may scan a DNA sequence for any or all of the yeast motifs or JASPAR motifs described above. The algorithm is also available for download at the same location. It is coded in Python (http://www.python.org), and requires Python version 2.3 or newer. MotifScan will use psyco (http://psyco.sourceforge.net) for a performance gain, if it is installed.

## DISCUSSION

We have developed a novel, graph-based supervised motif detection algorithm, which addresses apparent limitations in current probabilistic models. We compared the performance of our method to that of the PSSM, the *de facto* standard for motif detection, for the detection of both yeast and multicellular eukaryote motifs, and showed greatly improved performance. Our experiments indicate that the PSSM, and probabilistic methods in general, are a poor fit to many eukaryotic motifs, and that enumerative methods such as ours have many advantages. We find that there is extensive clustering of *k*-mers in eukaryotic motifs, which is

**Table 2.** A table of results comparing the performance of MotifScan and a PSSM for the detection of yeast and multicellular eukaryotic motifs

## Yeast motifs

MotifScan > PSSM+ 5%

| Motif Name | # k-mers | Area under ROC curve: MotifScan | PSSM |
|---|---|---|---|
| ADR1 | 29 | 100% | 49% |
| CAD1 | 22 | 78% | 67% |
| CIN5 | 135 | 100% | 84% |
| FKH1 | 154 | 100% | 90% |
| GCN4 | 177 | 100% | 82% |
| GLN3 | 79 | 98% | 79% |
| HAP4 | 37 | 90% | 82% |
| MSN2 | 32 | 72% | 60% |
| MSN4 | 37 | 100% | 73% |
| PHD1 | 116 | 100% | 62% |
| RAP1 | 112 | 89% | 76% |
| RCS1 | 59 | 90% | 75% |
| RDS1 | 10 | 100% | 88% |
| RFX1 | 11 | 93% | 84% |
| ROX1 | 28 | 99% | 82% |
| SKN7 | 125 | 91% | 22% |
| SOK2 | 184 | 100% | 65% |
| SPT2 | 13 | 65% | 56% |
| SPT23 | 53 | 100% | 83% |
| SUT1 | 42 | 31% | 22% |
| SWI4 | 128 | 100% | 90% |
| SWI6 | 179 | 100% | 81% |
| TEC1 | 96 | 100% | 92% |
| UME6 | 88 | 95% | 88% |
| YAP7 | 91 | 90% | 62% |

90-100%
80-90%
60-80%
40-60%
20-40%
0-20%

PSSM > MotifScan + 5%

| Motif Name | # k-mers | Area under ROC curve: MotifScan | PSSM |
|---|---|---|---|
| ABF1 | 29 | 79% | 85% |

## JASPAR motifs

MotifScan > PSSM+ 10%

| Motif Name | # k-mers | Area under ROC curve: MotifScan | PSSM |
|---|---|---|---|
| MA0001 | 97 | 86% | 58% |
| MA0002 | 29 | 88% | 67% |
| MA0005 | 90 | 54% | 33% |
| MA0006 | 24 | 100% | 80% |
| MA0008 | 25 | 100% | 78% |
| MA0011 | 12 | 100% | 26% |
| MA0014 | 12 | 22% | 7% |
| MA0015 | 80 | 82% | 66% |
| MA0020 | 21 | 100% | 86% |
| MA0031 | 20 | 100% | 65% |
| MA0034 | 25 | 29% | 12% |
| MA0037 | 63 | 100% | 65% |
| MA0038 | 53 | 66% | 37% |
| MA0040 | 18 | 72% | 60% |
| MA0041 | 47 | 77% | 61% |
| MA0044 | 13 | 50% | 5% |
| MA0054 | 70 | 100% | 55% |
| MA0056 | 20 | 100% | 79% |
| MA0057 | 16 | 36% | 22% |
| MA0063 | 17 | 100% | 48% |
| MA0067 | 31 | 95% | 25% |
| MA0070 | 18 | 67% | 53% |
| MA0077 | 76 | 93% | 63% |
| MA0080 | 57 | 100% | 65% |
| MA0081 | 49 | 100% | 70% |
| MA0084 | 28 | 73% | 39% |
| MA0086 | 40 | 100% | 83% |
| MA0087 | 23 | 100% | 78% |
| MA0089 | 34 | 100% | 87% |
| MA0092 | 29 | 67% | 35% |
| MA0095 | 17 | 100% | 86% |
| MA0098 | 40 | 100% | 82% |
| MA0103 | 41 | 100% | 69% |
| MA0105 | 18 | 74% | 60% |

PSSM > MotifScan + 10%

| Motif Name | # k-mers | Area under ROC curve: MotifScan | PSSM |
|---|---|---|---|
| MA0007 | 24 | 48% | 59% |
| MA0018 | 16 | 18% | 31% |
| MA0024 | 10 | 79% | 92% |
| MA0045 | 14 | 4% | 15% |

We show only those motifs where the performance of the two algorithms is significantly different. The number of motifs where MotifScan outperforms the PSSM is significantly greater in both yeast and JASPAR motifs.

substantially different from the prevailing 'diffused consensus' (or consensuses) model. There are two major reasons why this is the case.

We find that the *k*-mer clusters have functional significance. For a significant fraction of yeast motifs, the sequence of the motif, and not simply its presence or absence, affects the expression levels of the associated genes. Therefore, assigning a binding site to the appropriate motif subcluster may be an important step in understanding its gene expression. This observation is useful for future, integrative analyses of transcription factor binding sites and gene expression, and affirms the utility of a graph-based model in other aspects of transcription factor binding site analysis.

We showed that there is an evolutionary cause for *k*-mer clusters in motifs. Identical *k*-mers are more likely to be located in the same intergenic regions than non-identical *k*-mers, and clusters of identical *k*-mers within a motif are more likely to share conservation in flanking nucleotides. This makes it likely that DNA duplications are producing some of these motif occurrences.

In general, probabilistic methods suffer in motif detection because they assume that the *k*-mers in the motif are random

samples from a smooth distribution, with consensus sequences as the modes of the distribution. In fact, as is evident in the graphs in Figure 1, the distribution may have many 'spikes' of probability density due to small subclusters, particularly clusters of duplicated *k*-mers. Conversely, there are also regions of negligible probability density, i.e. *k*-mers that we would expect to appear in the motif given the nucleotide composition, that never do. Irregularly shaped distributions like these are naturally difficult to model parametrically. Since virtually all *de novo* motif-finding algorithms are based around the PSSM, or a diffused consensus model, it is not unreasonable to suggest that currently known motifs could be missing *k*-mers that become statistically significant if analyzed with the model we propose.

## ACKNOWLEDGEMENTS

We thank Serge Saxonov, Zachary Pincus, Jeffrey Chang and Devin Scannell for useful discussions. B. N. is supported by the Stanford Biochemistry Department. Work in the Batzoglou laboratory is supported in part by NSF grant EF-0312459, NIH grant U01-HG003162, the NSF CAREER Award and the Alfred P. Sloan Fellowship. Funding to pay the Open Access publication charges for this article has been waivered by Oxford University Press.

*Conflict of interest statement*. None declared.

## REFERENCES

1. Garten,Y., Kaplan,S. and Pilpel,Y. (2005) Extraction of transcription regulatory signals from genome-wide DNA–protein interaction data. *Nucleic Acids Res*., **33**, 605–615.
2. Bailey,T.L. and Elkan,C. (1994) Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proc. Int. Conf. Intell. Syst. Mol Biol*., **2**, 28–36.
3. Hughes,J.D., Estep,P.W., Tavazoie,S. and Church,G.M. (2000) Computational identification of cis-regulatory elements associated with groups of functionally related genes in *Saccharomyces cerevisiae*. *J. Mol. Biol*., **296**, 1205–1214.
4. Pavesi,G., Mereghetti,P., Mauri,G. and Pesole,G. (2004) Weeder Web: discovery of transcription factor binding sites in a set of sequences from co-regulated genes. *Nucleic Acids Res*., **32**, W199–W203.
5. Liu,X., Brutlag,D.L. and Liu,J.S. (2001) BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. *Pac. Symp. Biocomput*., 127–138.
6. Fratkin,E., Naughton,B.T., Brutlag,D.L. and Batzoglou,S. (2006) MotifCut: regulatory motif finding with maximum density subgraphs bioinformatics. In *Proceedings of International Conference on Intelligent Systems and Molecular Biology*. Fortaleza, Brazil.
7. Bertone,P., Stolc,V., Royce,T.E., Rozowsky,J.S., Urban,A.E., Zhu,X., Rinn,J.L., Tongprasit,W., Samanta,M., Weissman,S. *et al*. (2004) Global identification of human transcribed sequences with genome tiling arrays. *Science*, **306**, 2242–2246.
8. Stormo,G.D. (2000) DNA binding sites: representation and discovery. *Bioinformatics*, **16**, 16–23.
9. Schug,J. and Overton,G.C. (1997) TESS: Transcription Element Search Software on the WWW Technical Report CBIL-TR-1997–1001-v0.0. Computational Biology and Informatics Laboratory, School of Medicine, University of Pennsylvania, PA.
10. Benos,P.V., Lapedes,A.S. and Stormo,G.D. (2002) Probabilistic code for DNA recognition by proteins of the EGR family. *J. Mol. Biol*., **323**, 701–727.
11. Bulyk,M.L., Johnson,P.L.F. and Church,G.M. (2002) Nucleotides of transcription factor binding sites exert interdependent effects on the binding affinities of transcription factors. *Nucleic Acids Res*., **30**, 1255–1261.
12. Zhou,Q. and Liu,J.S. (2004) Modeling within-motif dependence for transcription factor binding site predictions. *Bioinformatics*, **20**, 909–916.
13. Barash,Y., Elidan,G., Friedman,N. and Kaplan,T. (2003) Modeling dependencies in protein–DNA binding sites. In *Proceedings of the Seventh International Conference on Research in Computational Molecular Biology*. Berlin, Germany.
14. Xing,E.P. and Karp,R.M. (2004) MotifPrototyper: a Bayesian profile model for motif families. *Proc. Natl Acad. Sci. USA*, **101**, 10523–10528.
15. King,O.D. and Roth,F.P. (2003) A non-parametric model for transcription factor binding sites. *Nucleic Acids Res*., **31**, e116.
16. Storey,J.D. and Tibshirani,R. (2003) Statistical significance for genomewide studies. *Proc. Natl Acad. Sci. USA*, **100**, 9440–9445.
17. Harbison,C.T., Gordon,D.B., Lee,T.I., Rinaldi,N.J., Macisaac,K.D., Danford,T.W., Hannett,N.M., Tagne,J.B., Reynolds,D.B., Yoo,J. *et al*. (2004) Transcriptional regulatory code of a eukaryotic genome. *Nature*, **431**, 99–104.
18. Sandelin,A., Alkema,W., Engstršm,P., Wasserman,W. and Lenhard,B. (2004) JASPAR: an open access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Res*., **32**, D91–D94.
19. Matys,V., Fricke,E., Geffers,R., Gossling,E., Haubrock,M., Hehl,R., Hornischer,K., Karas,D., Kel,A.E., Kel-Margoulis,O.V. *et al*. (2003) TRANSFAC: transcriptional regulation, from patterns to profiles. *Nucleic Acids Res*., **31**, 374–378.
20. Watts,D.J. and Strogatz,S.H. (1998) Collective dynamics of 'small-world' networks. *Nature*, **393**, 440–442.
21. Lapidot,M. and Pilpel,Y. (2003) Comprehensive quantitative analyses of the effects of promoter sequence elements on mRNA transcription. *Nucleic Acids Res*., **31**, 3824–3828.
22. Segal,E., Shapira,M., Regev,A., Pe'er,D., Botstein,D., Koller,D. and Friedman,N. (2003) Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature Genet*., **34**, 166–176.
23. Brauer,M.J., Saldanha,A.J., Dolinski,K. and Botstein,D. (2005) Homeostatic adjustment and metabolic remodeling in glucose-limited yeast cultures. *Mol. Biol. Cell*., **16**, 2503–2517.
24. Wang,Y., Liu,C.L., Storey,J.D., Tibshirani,R.J., Herschlag,D. and Brown,P.O. (2002) Precision and functional specificity in mRNA decay. *Proc. Natl Acad. Sci. USA*, **99**, 5860–5865.
25. Thomas,E.E. (2005) Short, local duplications in eukaryotic genomes. *Curr. Opin. Genet. Dev*., **15**, 640–644.
26. Achaz,G., Coissac,E., Viari,A. and Netter,P. (2000) Analysis of intrachromosomal duplications in yeast *Saccharomyces cerevisiae*: a possible model for their origin. *Mol. Biol. Evol*., **17**, 1268–1275.
27. Stolovicki,E., Dror,T., Brenner,N. and Braun,E. (2006) Synthetic gene recruitment reveals adaptive reprogramming of gene regulation in yeast. *Genetics*, **173**, 75–85.
28. Gu,Z., Cavalcanti,A., Chen,F.C., Bouman,P. and Li,W.H. (2002) Extent of gene duplication in the genomes of *Drosophila*, nematode, and yeast. *Mol. Biol. Evol*., **19**, 256–262.
29. Stone,J.R. and Wray,G.A. (2001) Rapid evolution of *cis*-regulatory sequences via local point mutations. *Mol. Biol. Evol*., **18**, 1764–1770.
30. Gojobori,T., Li,W.H. and Graur,D. (1982) Patterns of nucleotide substitution in pseudogenes and functional genes. *J. Mol. Evol*., **18**, 360–369.
31. Durbin,R., Eddy,S.R., Krogh,A. and Mitchison,G.J. (1998) *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge, UK.
32. Benjamini,Y. and Hochberg,Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R Stat. Soc. Ser B*, **57**, 289–300.
33. Story,J.D. (2003) The positive false discovery rate: a Bayesian interpretation and the q-value. *Ann. Stat*., **31**, 2013–2035.
34. Gribskov,M. and Robinson,N.L. (1996) The use of receiver operating characteristic (ROC) analysis to evaluate sequence matching. *Comput. Chem*., **20**, 25–34.