Integrating Genomic Conservation Data with Motif Discovery

by

Timothy W. Danford

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

March 2004
[June 2004]

Author .....
Department of Electrical Engineering and Computer Science
March 15, 2004

Certified by .....
David K. Gifford
Professor
Thesis Supervisor

Accepted by .....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Integrating Genomic Conservation Data with Motif Discovery

by

Timothy W. Danford

Submitted to the Department of Electrical Engineering and Computer Science
on March 15, 2004, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

Abstract

We formalize a probabilistic model of inter-species sequence conservation for motif discovery, and demonstrate that adding large-scale genomic conservation data to an existing motif discovery procedure improves the quality of that procedure's results. Existing motif discovery algorithms reveal binding motifs that are statistically over-represented in small sets of promoter regions. To the extent that binding motifs form a reliable part of a cell's regulatory apparatus, and that apparatus is preserved across closely related species, these binding motifs should also be conserved in the corresponding genomes. Previous studies have tried to assess levels of conservation in genomic fragments of several yeast species. Our approach computes the conditional probability of inter-species sequences, and uses this probability measure to maximize the likelihood of the data from different species with a motif model.

Thesis Supervisor: David K. Gifford
Title: Professor

Contents

Chapter 1

Introduction

We present a method for integrating global genome alignments into an existing motif discovery procedure that uses the Expectation Maximization algorithm. We begin with a brief survey of existing motif discovery algorithms, to place this procedure in its larger context. We continue by showing results to support our contention that the individual base alignments in regions of globally-aligned genomes are more commonly conserved within known motifs than in the genome as a whole. We then give a description of the data, and show how to modify the EM algorithm to incorporate it. Finally, we present results showing that adding a global alignment to the data improves the quality of discovered motifs, and the frequency with which known motifs are re-discovered, relative to the unimproved algorithm.

1.1 Previous Motif Discovery Work

A "motif" might be intuitively defined as a short sequence of letters that repeats throughout a larger set of words. Historically, biologists first searched for motifs in the primary sequences of closely related proteins; this was referred to as the "local alignment" problem, to distinguish it from procedures seeking global letter-for-letter matches between two sequences. Global alignment algorithms seek to maximize a total score on the alignment of two sequences, but it can be difficult to generalize the algorithms to more than two sequences at a time and assessing the reliability of local

matches can be problematic [1].

Local alignment algorithms take a different approach: ignoring global scores, they seek to find the substrings in the given sequences with high local correspondence. Many of these algorithms were originally used to search for corresponding regions of homologous proteins, in order to infer functional or structural characteristics [2]. Later work applied the same algorithms to DNA sequences; these tools became more useful as entire genomes were sequenced and made publicly available.

Starting with bacterial genomes, and later continuing with the genomes of yeast and other more complex organisms, these tools were enlisted and refined to systematically uncover the coherent fragments of DNA that give clues to the structure of the surrounding regulatory apparatus. Often the local alignments were organized to reveal the portions of DNA that would bind to transcription factors; these binding sites were summarized as instances of a "binding motif," and the process itself became known as "motif discovery."

Three choices are implicit in any motif discovery algorithm: how locally aligned sites are summarized into "motifs," how the algorithm searches the space of possible alignments (or how it scores intermediate local alignments), and how reliably biological conclusions can be drawn from the results, given the choice of algorithm and the input data.

Current motif discovery algorithms often make the first choice in one of two ways. Many algorithms choose to model the locally aligned sites as instances of an explicit probabilistic model. Two frequently used algorithms, MEME and AlignACE, use a product multinomial motif model [3, 4, 5]. This model contains a set of independent multinomial distributions (over separate nucleotides or, in the case of motif discovery in proteins, amino acids) modeling the contents of each position in the motif separately. Other algorithms have used this model as well [6, 5, 7, 8]. Although the independent column model is simplistic, it has been shown to accurately capture enough content in known binding motifs to be useful [9, 10]. Some algorithms have attempted to relax the independence assumption between distributions in adjacent positions [11].

Other algorithms have rejected the probabilistic modeling of motifs in favor of exact or discrete models. This approach usually models a "motif" as either a set of substrings or a substring along with an acceptable number of "mismatches." Sites are instances of the motif if they either match a member of the set of substrings [7, 12, 13, 14] or fall within a certain Hamming distance (the allowed number of mismatches) from the given substring [15, 16, 17].

The choice of motif model affects the way the algorithm optimizes that model over a given input data set, and different models also have different inherent advantages and drawbacks. Probabilistic models often lead to more tractable search algorithms, and describe position-specific ambiguities in a compact and easily-understood format. The notion of the information content of a probabilistic model is attractive, as well as the possibility of identifying sites that "almost" match the motif. Discrete models, on the other hand, easily model site-independent ambiguity and lead (in certain restricted circumstances) to exact or exhaustive search algorithms. Discrete models avoid the question (unavoidable in probabilistic models) of identifying sites that match the model; counting these sites is often essential to measures of "overrepresentation" or "discrimination" essential to many algorithms.

Once the motif model is given, a choice of the search or optimization algorithm for that model must be made. With the probabilistic models, two popular choices are the Expectation Maximization (EM) algorithm [18, 4, 6, 7] and the Gibbs Sampling algorithm [19, 5, 8, 20]. Fundamentally, these two optimization procedures are quite similar. Associated with any motif model and set of sequences (the input data) is a likelihood for each possible motif site within the input sequences. Gibbs sampling randomly chooses subsets of these sites with probability proportional to their likelihood and updates its motif model from those sites; this amounts to a random walk through this set of potentially bound sites until a motif emerges. EM moves through the same probabilistic landscape with its likelihood (Expectation) and update (Maximization) steps, but in a deterministic "hill-climbing" fashion. What Gibbs sampling achieves through random walking, EM achieves through random restarts to this local optimization step (since EM is performing the equivalent of gradient descent for local

optimization, adequately searching the search space for a global optimum requires "random restarts" of the algorithm at different points in the space to avoid re-finding local optima).

Different independence assumptions in the EM and Gibbs sampling algorithms produce corresponding constraints on the results of those algorithms. One common assumption is the "single-sequence" assumption; this assumes that, at most, one motif can occur per sequence. Several forms of EM commonly make this assumption. These variants of EM for motif discovery are termed OOPS and ZOOPS, which stand for "one occurrence per sequence" and "zero-or-one occurrenc per sequence," respectively.

Gibbs sampling commonly makes a different assumption, that individual bits of the alignment vector are independent of each other (constrained by, at worst, overlapping motif effects). No requirements are enforced on the sum of the bits of an alignment vector E_i. When this assumption is made in EM, it is commonly termed TCM, or "two-component model" EM.

The algorithms built around a discrete motif model show a wider range of search techniques. Some algorithms perform an exhaustive search [21, 7]. Others use data structures (such as suffix trees) to discover "structured" discrete motif models [17, 15]. Some algorithms use a randomized search (such as random hashing) to look for exact models that can't be discovered in an exhaustive search [16], or convert the input data into a graph or other data structure, and apply greedy solutions to the intractable problems that arise [14].

The third choice to be made, once a motif model and an algorithmic approach for optimizing it are chosen, is one of interpretation: how do we choose the input data (and incidental parameters) for our procedure? Under what conditions may we draw biological conclusions from what are often statistical results? The primary choice to be made for input data is, "Which biological sequences do we choose as input to our local alignment algorithm?" In the context of searching genomic promoter regions for binding sites (or some other functional piece of sequence), this is a question about the division, classification, and selection of genomic sequence. In searching for a binding motif, previous work has often assumed that co-regulated genes, those genes regu-

lated by a common transcription factor, should share a similar simple binding motif in their promoter regions. This lead to early approaches combining motif discovery with expression clustering experiments; genes whose expression profiles clustered tightly under different conditions were assumed to have a common regulator, and their promoter regions were used as input to motif discovery tools [22, 23].

Once the motif discovery algorithm has suggested one or several putative binding motifs, it is necessary to ask what biological conclusions can be drawn. Assuming that the sequences given as input accurately reflect the phenomenon under investigation, and that the relevant assumptions about binding mechanism (that a transcription factor has a consistent binding motif between genes and that the genes or regions in question are bound by a fixed set of factors) are accurate, it will generally follow that discovered motifs are related to the mechanism in question.

The relevancy of discovered motifs is related to their correct generalization of the input region set. A discovered motif is relevant in so far as it acts as an accurate classifier: if the presence or absence of a motif correctly predicts whether a region (or the corresponding gene) is part of the phenomenon under investigation, then the motif is a good candidate for a role in the causal or mechanistic description of that phenomenon.

This leads to characterizations of "enrichment" or "overrepresentation" in the world of motif discovery. Algorithms that use discrete motif models often contain steps to count instances of that motif in both the input and "background" set of regions and calculate overrepresentations scores. The algorithm of Buhler and Tompa uses a simple argument about expected counts to find enriched buckets in their randomized hashing scheme [16]. Other algorithms, notably that of Friedman and that of MDScan, use a hyper-geometric distribution to model expected motifs counts [7, 24]. Some algorithms eschew a background model in favor of a background region set; reported motifs are those that discriminates well between foreground and background region sets [13]. Explicit use of a "negative" sequence set should lower the number of false motifs reported, a common problem among many motif discovery algorithms [25].

Algorithms with probabilistic motif models often use a statistical notion of the "background" region set (that is, the sequence set from which the input set is drawn) in the optimization of the model. Many algorithms estimate parameters of an n-th order Markov model to use in conjunction with the motif model [4, 20, 26]. Some algorithms assumed the use of a 0^{th} order model; others have investigated the advantages of a higher order model [27, 28]. Ultimately, the likelihood of a site in these algorithms depends not just on the current notion of the target motif, but also on how well the background model explains either the site itself, or the surrounding sequence.

Finally, many probabilistic algorithms use a calculation of overrepresentation either as a pre-processing step (culling the seeds to EM, for instance) or a post-processing filter. Conversely, some discrete-model algorithms will use a probabilistic tool as a means of refining discrete result [7, 24]. Ultimately, many tools mix and match several of these options in complementary and overlapping ways, often tailored to the specific problem at hand.

In this work, we will try to make as few irrevocable choices as possible in our selection of a motif discovery algorithm. Instead we wish to modify an existing motif discovery algorithm to incorporate the use of genomic conservation data. This is followed with a comparison of the modified algorithm with the results of the original method. We will assume the use of an algorithm with a probabilistic motif model and we use calculation of motif overrepresentation as a post-processing step; in essence, we are modifying a vanilla EM or Gibbs motif discovery tool to use the conservation data.

1.2 Previous Genome-wide Conservation Work

Existing work on the conservation and evolution of genomic or proteomic sequences consists of variations on one basic theme: simultaneous explanation of multiple sequences within one coherent, parametrized probabilistic model. Different methods make different choices as to what kinds of models are admissible, which (if any) parameters are made explicit, what sorts of sequence regions are modeled, and to what

end this modeling is performed [29, 30, 31, 32, 33, 34, 35, 36].

The most basic kind of conservation model assembles a parametrized tree structure (a "phylogenetic tree") to accurately explain long sequences. Usually such a sequence is the primary sequence of a protein or a gene, and the tree is used to aid multiple alignment algorithms. The model of evolution in this case has two parts, the topology of the tree itself, and the notion of branch length or evolutionary "time." While these may have a tenuous relationship with the sequences' real evolutionary histories, together they provide a framework for modeling different rates of parsimony between sequence regions. Some models will account for insertion and deletion along with substitution.

This conceptual framework is simple. The sequence regions are presumed to be noisy instantiantions of an "ancestor" sequence. Inferring the structure and parameters of a single tree suggests a way in which the sequences may have evolved, or at least provides a quantitative comparison of global similarity.

Sometimes the "distance" parameter is explicit, as in a maximum likelihood approach to phylogeny. Other times the parameter is implicit as in the parsimony approach to tree-building.

One variation relaxes two of these assumptions. First, multiple models of conservation and evolution are admitted (we simultaneously consider several differently structured and parametrized trees). Second, the assumption that different sites within the same sequence region are related in the same way to other sequences is relaxed. The multiple tree models are used to explain the differential conservation of different sites in the same sequence region.

This relaxation leads, in turn, to a new use for conservation modeling. Conservation models are inferred as before, only at finer sequence detail. But if the scale of the sequence sites that are explained with different models is small enough that the same (or "similar") sites may occur multiple times in the same sequence regions, we can now ask an important question: "Which abstract sequence strings are conserved more frequently than we would expect if conservation was random?"

This question suggests a semantic generalization for the parameters of the inferred

phylogenetic trees. Previously, the conservation tree models were parametrized with some "distance"; assuming sequences are evolving at constant (or proportional) rates, this distance can be read as an "evolutionary time."

When we model different sites in the same sequence with different models simultaneously, this definition needs to change. Presumably, most sites have been in a single sequence for a similar length of (real) time. A significant difference in evolutionary distance implies that different sites are changing at different rates.

Much work has gone into identifying functional sites through such differential evolution. Those sites that share both sequence similarity and consistently high conservation parameters ("close" evolutionary distances) are more likely to be functional, assuming that functional sites evolve more slowly than the surrounding sequence under looser evolutionary stabilization. The parameter of the evolutionary model has undergone a semantic loosening, from "time" to a sort of "importance."

Several other interesting questions can be asked in this regime of conservation modeling. First, if the sequences we are modeling come from different species, we may ask "which species will give assessments of differential conservation that best pick out the functional sites?" Some previous papers have attempted to quantify the answer to this question, at least in the case of closely-related yeast species [37].

An additional variation is available here in the choice of a tree parameter: should it be real-valued or binary? We will designate these choices as the "distance" and "functional" form of the conservation parameter, although the division is not so sharp. Choosing a real-valued tree parameter implies a maximum-likelihood approach to tree-building, and the final parameter often looks like an evolutionary distance. The binary parameter instead assumes the sequences (and their sites) are noisy instantiations of sites which are (in reality) either completely the same or completely different. By assuming that sequence sites are modeled by the binary choice of "motif" or "background" conservation distributions in this work, we will be implicitly using the latter approach.

But one major assumption still remains: the similarity of different sites is conditioned on the parameters of the model, and no systematic restriction is made on the

types of models available to represent the same *sequence* at different *sites*. If the goal is not to build a realistic phylogenetic tree but to discover functional sequence sites, then this is a reasonable assumption to make.

One recent paper, Kellis et. al., systematically makes this assumption and uses it to search for sites considered to be functional [38]. The sites of a putative motif are *a priori* classified according to a binary category (for instance, genic vs. intergenic sites). An implicit alignment is assumed, and a single conservation model is inferred for each kind of site. A putative motif is said to be "functional," and included in the output, if the parameters of each model are sufficiently different. This process is repeated for several different forms of conservation model.

Our approach in this work is most similar to this last method. Known functional sites are used to infer a "functional" conservation model, and the corresponding "background" model. Rather than using these models exclusively to rank putative motifs, however, we use this score of differential conservation to bias a traditional method of motif discovery toward such regions of differential conservation. In this way, our method provides an algorithm for simultaneously fusing sequence similarity between sites and between species into a single model for motif discovery.

1.3 Outline of Work

The first part includes the Data section, the Models section, and the Methods section. These three sections parallel each other. The Data section gives a qualitative description of the data we will use in this paper, along with some preliminary statistics and an outline of data's practical problems. The Models section presents a formal mathematical description of the data, as well as the models whose parameters our algorithm learns from the data. The Method section explains how these models are evaluated; it also lays out our solutions to the practical data challenges explained in the Data section.

The second part consists of the Results section, and the Discussion and Conclusions sections. Here we summarize the results of running our modified algorithm, and

compare them to the results of the corresponding unmodified algorithm. Finally, we discuss the choices we made, possible avenues for future work, and argue that adding conservation data improves our results. We provide summary statistics in the Results section, and a full listing of all results and corresponding statistics in the Appendix.

Chapter 2

Data

We have four data sources available for input to both versions of the motif discovery algorithm:

- *Factor Set*: A set of transcription factors in *Saccharomyces cerevisiae*, for which binding motifs are known with high confidence.
- *Genome*: A reference sequence for each intergenic region of DNA in *S. cerevisiae*.
- *Binding Data*: For each transcription factor, a list of genes whose promoter is bound by that factor.
- *Conservation*: A global-alignment for each region of the reference genome to the genomes of K other species.

2.1 Factor Set

The first data set we will use is a list of yeast transcription factors. In particular, to test the results of our augmented algorithm (and compare against the results of the original algorithm), we choose yeast transcription factors whose binding sites are well characterized. From a list of factors with well-characterized TRANSFAC motifs [39], we have chosen ten factors whose motifs are well-represented in their bound probes. This factor set is summarized in Table 2.1.

In this table, `Bound Regions` is the number of contiguous intergenic regions we will use as input to the EM algorithm when trying to (re)discover the motif for this factor. Each of these regions contains a probe in the binding data with a p-value below a specific cutoff (0.001 in this case). `Bound Motif Sequences` is the number of sequences in the bound set that contain *at least* one instance of the consensus motif. "Total Motif Sequences" is the number of sequences in the total sequence set (4984 separate regions were considered as potential promoter-containing sequence regions) that contain at least one instance of the consensus motif.

Factor	*Consensus Motif*	*Bound Regions*	*Bound Motif Sequences*	*Total Motif Sequences*
ABF1	TC$^{A}_{G}$nnnnnnACG	168	160	1227
CBF1	$^{A}_{G}$TCAC$^{A}_{G}$TG	26	23	231
GCN4	TGACTCA	50	34	170
GCR1	CTTCC	13	9	2441
HAP3	CCAATnA	20	12	975
HAP4	CCAATnA	39	25	975
MCM1	CCnnn$^{A\,A\,A}_{T\,T\,G}$GG	57	40	783
REB1	CGGGT$^{A\,A}_{G\,G}$	90	73	660
STE12	TGAAACA	43	24	540
YAP1	TTACTAA	35	20	595

Table 2.1: The base set of factors and corresponding known motifs.

2.2 Reference Genome

Our second data source is the reference sequence for yeast, the public *Saccharomyces cerevisiae* genome. We discuss the procedure for choosing which portions of the genome to search for motifs in Section 2.5. It is important to note, however, that we view both the binding and conservation data as annotations to this reference genome.

2.3 Binding Data

Once we choose a transcription factor to investigate, we select the genes that we will consider to be co-regulated according to the results of a genome-wide binding location assay [40, 41]. This data is given in a matrix: each experiment (combining both a factor and a condition) contains 5419 values, the p-value for each probe in the data set. Since our interest in discovering motifs is centered on the sequences between ORFs (as opposed to the sequences of the bound probes from the binding data), we consider a region to be bound when it contains one or more bound probes. Furthermore, in this work we only focus on the bound probe sets of factors in the YPD condition.

2.4 Conservation Data

Finally, the conservation data accompanies the reference sequence in the form of a global alignment to each intergenic region. In particular, for each spatially adjacent pair of ORFs in the genome, we are given the sequence covering the region between these ORFs, and a global alignment of that region with the corresponding regions of the other species [38].

2.5 Combining the Data

These data are consistent with each other and must be combined. The combination of sequence, binding, and conservation data presents several hurdles; failure to consistently clear these hurdles can introduce noise to the input data and harm later results.

The first hurdle is selecting, for a given factor, the bound probes. For our purposes, this is equivalent to the problem of choosing a p-value threshold for the binding data. Some work has focused on sharpening the threshold of an arbitrary p-value through combination with other data (with expression data for instance, see [42]). Here, we sidestep the issue by first selecting the binding threshold to be sufficiently tight to

avoid significant false positives (we hope) in the bound probe set: this is 0.001. We then choose to focus only on those factors whose *a priori* motifs are well represented within their bound regions. Since the ultimate question is not *whether* we re-discover the motif, but whether one technique discovers it when the other does not, this should not bias our conclusions. Instead, it should limit those cases where *both* techniques fail to rediscover the known motif, thus allowing us to identify regions of improvement more accurately.

The second hurdle is converting the selected probes into into a set of regions from the genome. The sequences we wish to mine for motifs are the regions between two consecutive coding sequences, what we call "intergenic" regions. Probes are smaller sequences that sometimes fall within these larger regions, and probes are the units to which the binding data assigns "bound" or "unbound" classification. As mentioned earlier, we give the same classification to our intergenic regions by calling such a region "bound" when it contains one or more bound probes.

Finally, we need to ensure that the sequences of *Saccharomyces cerevisiae* given in the conservation data's global alignments match the sequences we have calculated for our bound intergenic regions. Once we have performed this final check, we can use the appropriate "bound" regions' global alignments as input to our motif discovery algorithms.

Chapter 3

Models

3.1 Sequence Data

We use $\mathcal{S}$ to denote the set of all possible intergenic sequences.

$$\mathcal{S} = \{S_1, \ldots, S_m\} \tag{3.1}$$

Each sequence S_i is a finite string of values from the set Σ, with length L_i. S_i can be viewed as a function from the set of integers $[1 \ldots L_i]$ to Σ.

$$\begin{aligned} S_i &= S_i(1) \ldots S_i(L_i) \\ S_i(j) &\in \Sigma \\ \Sigma &= \{\mathrm{A}, \mathrm{T}, \mathrm{G}, \mathrm{C}, \mathrm{gap}\} \end{aligned} \tag{3.2}$$

We will consider motif models with a fixed width W. To shorten our notation, we use S_{ij} to denote the W-width word beginning at position j in S_i. Conversely, $\tilde{S}_{ij}$ will denote the string of bases in S_i with S_{ij} removed.

$$\begin{aligned} S_{ij} &= S_i(j), \ldots, S_i(j+W-1) \\ \tilde{S}_{ij} &= S_i(1), \ldots, S_i(j-1), S_i(j+W), \ldots, S_i(L_i) \end{aligned} \tag{3.3}$$

3.2 Conservation Data

This notation for a set of sequences can describe the input to most motif discovery tools. The addition of conservation alignments, however, adds a new dimension to our data. We now have up to K sequences from other species associated with each sequence S_i. These other sequences are given in the form of global alignments; that is, they are matched base-for-base with the possibility of gaps. This explains the addition of the "gap" symbol to the alphabet Σ in (eq. 3.3) above.

For each $k \in [1 \ldots K]$, we associate an "alignment function" for the k^{th} species, A_k. For each base $S_i(j)$ in the reference sequence, $A_k(S_i(j))$ is the corresponding base in aligned genome k; this aligned symbol is usually a nucleotide $\{A, T, G, C\}$, but may also be a gap symbol.

In a slight abuse of notation, we will allow the use of such an alignment function A_k on strings to denote application on each element of the string: $A_k(S_{ij}) = A_k(S_i(j)) \ldots A_k(S_i(j+W-1))$, and similarly with $A(\tilde{S}_{ij})$.

3.3 Motif Models

We indicate the parameters of the motif and background models with $\Theta = (\Theta_M, \Theta_B)$. Typically, Θ_B is the parameter set of a Markov sequence model. The technical details of these background models are well-established [28, 27, 26, 4], and are orthogonal to our choice of conservation model.

We use Θ_M to denote the parameters of a product multinomial motif model, as have almost all other probabilistic motif discovery tools. Our notation will be straightforward; if our motifs are of length W, then:

$$\Theta_M = [\theta_0 \ldots \theta_{W-1}] \tag{3.4}$$

Each θ_i is a multinomial distribution over the elements of the alphabet Σ. We use the notation $\theta_i[S_j(k)]$ to name the likelihood of base $S_j(k)$ under this model. We will abuse notation (as above) to calculate the likelihood of an entire site simultaneously.

$$\Theta_M[S_{ij}] = \prod_{k=0}^{W-1} \theta_k[S_i(j+k)] \tag{3.5}$$

The likelihood of a base under the background model will be notated in a similar way. If $\Theta_B[S_i(j)]$ is the likelihood of the base at position j of sequence S_i (a calculation which, under a Markov background model of order n implicitly takes into account $S_i(j-n)\ldots S_i(j-1)$), then:

$$\Theta_B[\tilde{S}_{ij}] = \prod_{k=0}^{j-1} \Theta_B[S_i(k)] \times \prod_{k=j+W}^{L_i-1} \Theta_B[S_i(k)] \tag{3.6}$$

Therefore, the simultaneous likelihood of a single motif site S_{ij} and its surrounding background sequence under the parameters of a model Θ is:

$$\Pr\{S_i|\Theta\} = \Theta_M[S_{ij}]\Theta_B[\tilde{S}_{ij}] \tag{3.7}$$

3.4 Site Models

Dual to the motif model Θ is, for a given input dataset $\mathcal{S}$, the alignment vector. In Expectation Maximization this is the hidden data; in Gibbs sampling, it is the vector in whose space the sampling randomly walks. The alignment (as distinguished from the "alignment functions" which encapsulate the conservation data) is the set of motif *sites* that are generalized by the motif *model.*

To indicate an alignment, we define a vector $\mathbf{E}$ of equal size to the input set.

$$\mathbf{E} = E_1, \ldots, E_M \tag{3.8}$$

The contents of this vector vary depending on the form of model evaluation we use. If we are using a one-per-strand search algorithm (such as the OOPS or ZOOPS variants of EM; c.f Section 1.1), each E_i is an integer in $[1 \ldots L_i]$, indicating the starting position of the motif site in S_i.

$$E_i \in [1 \ldots L_i] \tag{3.9}$$

If we use a per-site search algorithm (such as the TCM variant of EM, or a Gibbs sampling algorithm) to evaluate our motif model, then each E_i is actually a bit vector of length $L_i - W$.

$$E_i = e_{i,0} \ldots e_{i,L_i-W} \tag{3.10}$$

Each bit indicates whether the corresponding word is a site for the motif model. We use $\mathcal{M}$ to name the set of sites for the current model Θ. Therefore, if $e_{ij} = 1$ for a particular pair (i, j), this implies that $S_{ij} \in \mathcal{M}$.

The alignment vector $\mathbf{E}$ (equivalently, the set $\mathcal{M}$) is a sufficient statistic for the motif model Θ_M. This is based on the assumption that we only ever calculate the parameters of Θ from the frequencies of bases in fixed positions of the sites in $\mathcal{M}$ (and possibly a set of unchanging psuedocounts). Given a threshold likelihood and a fixed background model, the reverse calculation can be made: the motif model can be used to check every potential site and find the positive elements of $\mathbf{E}$ (the sites in $\mathcal{M}$). These operations for converting Θ into $\mathbf{E}$, and vice versa, form the basis of most iterative probabilistic motif discovery methods (including both Expectation Maximization and Gibbs sampling).

3.5 Conservation Models

As we will see in Section 3.7, our model of conservation will be estimated as if each aligned position in the sequence set $\mathcal{S}$ was an independent sample from one of two distributions. The likelihood of $A_k(S_i(j))$ depends only on the identity of the reference base $S_i(j)$ and whether the position is considered to be part of a motif site or not (the latter information indicates whether the required probability is drawn from a "foreground" or "background" distribution). We name this pair of distributions $\Xi = (\Xi_M, \Xi_B)$, and we estimate the two models Ξ_M and Ξ_B separately for positions that

are a priori known to be in a site or in the background.

We assume, as described in Section 3.2, that there are K sequences aligned with each region of the reference genome. At each position (i, j) the K-tuple $(A_1(S_i(j)), \ldots, A_K(S_i(j)))$ is an element of $\Sigma_1 \times \Sigma_2 \times \ldots \Sigma_K$. The calculation (from observed frequencies) of

$$\Pr\{A_1(S_i(j)) \ldots A_K(S_i(j)) | S_i(j)\}$$

for both foreground and background positions, is a key preliminary step in our method. Therefore, using the shorthand $\mathcal{A}(S_i(j)) = (A_1(S_i(j)) \ldots A_K(S_i(j)))$, we define the shorthand symbols Ξ:

$$\Xi_M\{\mathcal{A}(S_i(j))\} = \Pr\{\mathcal{A}(S_i(j)) | S_i(j), S_{ij} \in M\} \quad (3.11)$$

$$\Xi_B\{\mathcal{A}(S_i(j))\} = \Pr\{\mathcal{A}(S_i(j)) | S_i(j), S_{ij} \notin M\} \quad (3.12)$$

As before with Θ, we abuse notation slightly and let Ξ apply to S_{ij} and $\tilde{S}_{ij}$ as well:

$$\Xi_M(\mathcal{A}(S_{ij})) = \prod_{j'=0}^{W-1} \Xi_M(\mathcal{A}(S_i(j + j'))) \quad (3.13)$$

$$\Xi_B(\mathcal{A}(\tilde{S}_{ij})) = \prod_{j'=0}^{j-1} \Xi_B(\mathcal{A}(S_i(j')) \times \prod_{j'=j+W}^{L_i} \Xi_B(\mathcal{A}(S_i(j'))) \quad (3.14)$$

The justification for this expansion will be given in the following section.

3.6 Independence Assumptions

The central equations of Expectation Maximization and Gibbs sampling already encompass the sequence and motif models. As a final step, we will show how to rewrite these equations to use the conservation model Ξ as well. For this step, we need to make two assumptions about the conservation model:

- *Independence:* We assume that the conservation of a sequence region of identical functional class is equivalent to the joint conservation of each included position in the region, and that this joint probability factors into the probability of conservation at each position. In other words, that the probabilities of conservation at different positions are independent. In the equations which follow, we use the term **pos** as the name of a (hidden) variable whose value indicates where in a given site a certain base is positioned.

$$\Xi\{\mathcal{A}(S_i)\} = \Xi_M\{\mathcal{A}(S_{ij})\} \times \Xi_B\{\mathcal{A}(\tilde{S}_{ij})\} \tag{3.15}$$

$$\Xi_M\{\mathcal{A}(S_{ij})\} = \prod_{k=0}^{W-1} \Xi_M\{\mathcal{A}(S_i(j+k)), \mathbf{pos} = k\} \tag{3.16}$$

- *Isotropism:* We will assume that sequence conservation is determined up to its functional class (functional or non-functional). This assumption requires that sequence conservation behave identically across all known motifs.

  We start by expressing the assumption that conservation "looks" the same within a single site. If the variable **pos** denotes the relative position of a base $S_i(j)$ within a motif site, we assume that conservation at a base is independent of that position. If $j^* = j + k$, then:

$$\Xi_M\{\mathcal{A}(S_{ij}), \Theta\} = \Xi_M\{\mathcal{A}(S_{ij})\} \tag{3.17}$$

$$\Xi_M\{\mathcal{A}(S_i(j^*)), \mathbf{pos} = k\} = \Xi_M\{\mathcal{A}(S_i(j+k))\} \tag{3.18}$$

  Second, we wish to express the assumption that conservation looks the same across different bases in the same site, across different sites: $\forall (i,j), (i',j') \in M$ and $\forall k \in [1 \ldots W]$,

$$\Xi_M\{\mathcal{A}(S_i(j+k))\} = \Xi_M\{\mathcal{A}(S_{i'}(j'+k))\} \tag{3.19}$$

3.7 EM Update Equations

Existing EM Equations

At its core, the Expectation Maximization algorithm is a method for finding the parameter vector Θ which maximizes the posterior probability of an observed data set D [43]. The algorithm is an iterative one, and depends on the presence of unobserved data U.

$$\texttt{EM}(D) = \arg\max_{\Theta'} P(D|\Theta') \tag{3.20}$$

When using EM to perform Motif Discovery, the data D is the observed set of sequences $\mathcal{S}$, and equation (3.20) has a similar form [4]. The unobserved data is the "alignment vector" $\mathbf{E}$.

$$\texttt{MEME}(\mathcal{S}) = \arg\max_{\Theta'} P(\mathcal{S}|\Theta') \tag{3.21}$$

EM's iterative process has two steps, an "Expectation" step (or E-step) and a "Maximization" step (or M-step). The E-step calculates a posterior probability over the unobserved data using the current parameter vector Θ; the M-step maximizes the expectation (calculated with respect to this probability distribution over the hidden data) of the data's joint log likelihood. The next value of the Θ parameter is chosen to maximize this last value.

In terms of standard motif discovery EM, the E-step equation is written:

$$\pi_i(j) = P(E_i = j|\mathcal{S}, \Theta^{(n)}) \tag{3.22}$$

Equation 3.22 is the equation for a probability distribution over the integers $[0, \ldots, L_i - 1]$; in other words, we will be explaining and deriving the equations for the single-motif-per-sequence variants (OOPS and ZOOPS) of EM. Using the standard assumptions of model-independence typical to an algorithm such as MEME, equation (3.22) takes the form:

$$\pi_i^{\text{old}}(j) = P(E_i = j|\mathcal{S}, \Theta^{(n)}) \tag{3.23}$$

$$= \frac{\lambda_j \Theta_M[S_{ij}]\Theta_B[\tilde{S}_{ij}]}{\sum_{j'} \lambda_{j'} \Theta_M[S_{ij'}]\Theta_B[\tilde{S}_{ij'}]} \tag{3.24}$$

In these equations, the variable λ_j denotes the per-site prior probability of alignment, and is usually uniform (it will also encapsulate the per-sequence parameter γ for EM variants that allow zero alignments to a sequence). The values of $\pi_i(j)$ for all possible j are the explicit parameters necessary for calculating an expected log likelihood function of the joint hidden and observed data.

$$g(\Theta) = E_{\pi(\Theta^{(n)})}\{\log P(\mathcal{S}, \mathbf{E}|\Theta)\} \tag{3.25}$$

The corresponding M-step chooses a new value for the parameter vector that maximizes this expected log likelihood:

$$\Theta^{(n+1)} = \arg\max_{\Theta'} E_{\pi(\Theta^{(n)})}\{g(\Theta')\} \tag{3.26}$$

The creators of the MEME algorithm showed how, as a result of the independence assumptions implicit in their motif model, this equation is maximized by a matrix of per-position base frequencies (with pseudo-counts), averaged over the posterior probability of $\mathbf{E}$.

New EM Equations

The first change we make to the equations of EM is that we assume our observed data now consists of the aligned sequences $\mathcal{A}(S)$ along with any sequence S.

$$\texttt{MEME+}(\mathcal{S}, \mathcal{A}(\mathcal{S})) = \arg\max_{\Theta'} P(\mathcal{S}, \mathcal{A}(\mathcal{S})|\Theta', \Xi) \tag{3.27}$$

Furthermore, the corresponding expectations over the hidden data $\mathbf{E}$ now involve an additional fixed parameter Ξ, the estimated foreground and background conserva-

tion probabilities.

$$\pi_i(j) = P(E_i = j|\mathcal{S}, \mathcal{A}(\mathcal{S}), \Theta^{(n)}, \Xi) \tag{3.28}$$

The M-step equation has a similar updated form; furthermore, the independence assumptions of the conservation model mirror those of the motif model to such an extent that the maximizing form of Θ stays the same.

$$\Theta^{(n+1)} = \arg\max_{\Theta'} E_{\pi(\Theta^{(n)})}\{\log P(\mathcal{S}, \mathcal{A}(\mathcal{S}), \mathbf{E}|\Theta', \Xi)\} \tag{3.29}$$

Therefore, we need only show how the updated E-step equation (3.28) is expanded in terms of both the parameters Θ and Ξ.

$$\begin{aligned}
\pi_i^{\text{new}}(j) &= P(E_i = j|\mathcal{S}, \mathcal{A}(\mathcal{S}), \Theta^{(n)}, \Xi) && (3.30)\\
&= \frac{P(\mathcal{A}(\mathcal{S})|E_i = j, \mathcal{S}, \Theta^{(n)}, \Xi)P(E_i = j|\mathcal{S}, \Theta^{(n)}, \Xi)}{P(\mathcal{A}(\mathcal{S})|\mathcal{S}, \Theta^{(n)}, \Xi)} && (3.31)\\
&= \frac{P(\mathcal{A}(\mathcal{S})|E_i = j, \mathcal{S}, \Theta^{(n)}, \Xi)P(E_i = j|\mathcal{S}, \Theta^{(n)}, \Xi)}{\sum_{j'} P(\mathcal{A}(\mathcal{S})|E_i = j', \mathcal{S}, \Theta^{(n)}, \Xi)P(E_i = j'|\mathcal{S}, \Theta^{(n)}, \Xi)} && (3.32)
\end{aligned}$$

This final form for $\pi_i^{\text{new}}(j)$ involves both the old (i.e., without conservation) equation for the expectation over j, and a new factor (in both numerator and denominator) which explains the aligned sequences given the base sequence. These factors are the new components of the EM equations which bias the search for conserved sequence motifs. We start by re-writing this last equation in a way that illuminates the old expectation equation's role.

$$\pi_i^{\text{new}}(j) = \frac{\Xi_M[S_{ij}]\Xi_B[\tilde{S}_{ij}]\pi_i^{\text{old}}(j)}{\sum_{j'} \Xi_M[S_{ij'}]\Xi_B[\tilde{S}_{ij'}]\pi_i^{\text{old}}(j')} \tag{3.33}$$

The sum in equation (3.22)'s denominator factors out, and we can rewrite this as:

$$\pi_i^{\text{new}}(j) = \frac{f_{ij}\lambda_j\Theta_M[S_{ij}]\Theta_B[\tilde{S}_{ij}]}{\sum_{j'} f_{ij'}\lambda_{j'}\Theta_M[S_{ij'}]\Theta_B[\tilde{S}_{ij'}]} \tag{3.34}$$

Here, the term $f_{ij} = \Theta_M[S_{ij}]\Theta_B[\tilde{S}_{ij}]$ indicates how a per-site factor for the conservation model is multiplied into the existing EM equations.

We will refer to EM (MEME and its variants) or Gibbs algorithms using the original equations, (eq. 3.20), as "Plain" MEME or Gibbs. If the algorithm uses our updated formula, (eq. 3.27), for its likelihood equation we will call it MEME+, or MEME with conservation.

Chapter 4

Method

We will detail the novel elements of our method in Sections 4.2 and 4.3. We have already explained (in Section 3.7) the way in which the updated EM algorithm handles our new data. We explain how we systematically learn the conservation parameters for the updated EM, how we attempt to rediscover the known consensus motifs for our factor set, and how we calculate the final post-processing statistics to assess our success.

4.1 Conservation Assessment

We have used the symbols $\Xi_M(\mathcal{A}(S_{ij}))$ and $\Xi_B(\mathcal{A}(S_{ij}))$ to indicate the probability of seeing certain aligned configurations of bases (and gaps) in either the "functional" (motif) or "background" regimes. Although we have indicated how these probabilities are incorporated into the existing discovery algorithms, we need to explain how their parameters are explicitly calculated. Since Ξ_M and Ξ_B differ only in which data they summarize, we will refer generically to Ξ in what follows to show how they are both learned.

Since we assume a finite alphabet (Σ) and a finite number (K) of aligned sequences, Ξ takes the form of an K-dimensional matrix with $|\Sigma|^K$ entries. Each element of the matrix will contain a raw count; we will denote these counts with the notation Ξ^*. Therefore, $\Xi^*(A_1(S_{ij}) = b_1, \ldots, A_K(S_{ij}) = b_K)$ is the number of times

we saw the bases $(b_1, \ldots, b_K)$ aligned together in the dataset.

Therefore, the probability $\Xi(\mathcal{A}(S_{ij}))$ of our model can be expressed:

$$\Xi(b_1, \ldots, b_K) = \frac{\Xi^*(b_1, \ldots, b_K)}{\sum_{(b'_1, \ldots, b'_K) \in \Sigma^K} \Xi^*(b_1, b'_1, \ldots, b'_K)} \tag{4.1}$$

All that remains is to indicate which aligned positions in the genome are used as the data for learning the Ξ_M and Ξ_B and use in the rediscovery of a certain factor's motif . We show how this is done in the following section, using the notation `CONS_MOTIF` and `CONS_BG` to indicate the matrices Ξ^*_M and Ξ^*_B respectively.

4.2 Algorithms

We describe here the algorithms to learn the parameters of each conservation model, and the subsequent use of those models in EM. The term `FACTOR_SET` is the set of all known factors (Table 2.1). We use the notation `CONS += r(i)` to indicate that the model `CONS` is updated to incorporate the data `r(i)`. The `em()` function denotes the ZOOPS-variant of the EM algorithm; when given the additional parameters `CONS_MOTIF` and `CONS_BG`, the learned conservation models, it incorporates them into its likelihood equation as detailed in Section 3. The `stats()` function is the method that calculates the statistics for each discovered motif. We state the formulae for these statistics in Section 4.3.

In both both figures we define the set `B` to be the "probes bound by [a factor] `F`." This is the direct interpretation of the binding location data; given a factor (and a threshold p-value, which is always 0.001), the data indicates a set of probes that the experiment determines are "bound." The set `I(B)` represents the translation of that data into the conservation data set. For each probe, one or more intergenic regions of sequence (that is, contiguous blocks of sequence between consecutive open reading frames) are indicated. These regions are the input sequence set `S` to the EM algorithm.

The first figure specifies how the conservation matrices (Ξ_M and Ξ_B) are calcu-

lated, prior to any motif discovery process.

```
cons(FACTORS) =
 1.  For each factor F in FACTORS
 2.        Let M = motif of F
 3.        Let B = probes bound by F
 4.        Let I(B) = intergenic regions corresponding to B
 5.        For each R in I(B)
 6.              Let R = r(1)...r(L)
 7.              Let S = {i : r(i) is in a site of M}
 8.              For each i in S
 9.                    CONS_MOTIF += r(i)
 10.             For each i' in [1...L] - S
 11.                   CONS_BG += r(i')
 12.  return CONS_MOTIF, CONS_BG
```

The second half of our method attempts to re-discover the known motifs for our training factor set, both with and without the conservation matrices. Motif discovery is run twice for each factor, using all the same conditions and differing in only whether conservation probability is incorporated. Finally, the statistics are calculated for both sets of results.

```
main(FACTOR_SET) =
 1.  For each factor F in FACTOR_SET
 2.        Let B = probes bound by F
 3.        Let I(B) = intergenic regions corresponding to B
 4.        Let CONS_MOTIF = cons(FACTOR_SET - {F})
 5.        Let CONS_BG = cons(FACTOR_SET - {F})
 6.        Let M_PLAIN = em(I(B))
 7.        Let M_CONS = em(I(B), CONS_MOTIF, CONS_BG)
 8.        Let STATS_PLAIN = stats(M_PLAIN)
 9.        Let STATS_CONS = stats(M_CONS)
```

```
10. return STATS_PLAIN, STATS_CONS
```

We show the entire method in a graphical form in Figure 4-1.

4.3 Statistics for Discovered Motifs

In this section we describe the statistics calculated for each motif discovered by the EM algorithm, with and without conservation. These statistics are meant to measure, in relatively independent ways, how close each motif is to either capturing the notion of the "bound set" from the binding data, and how close it is to capturing the identity of the known motif. The first such type of statistic would be calculated in a realistic motif-discovery situation where the target motif is unknown; the second is specific to this work, and gives an unbiased way to compare the results of each method.

Some of these statistics (the hypergeometric and binomial statistics are on a per-sequence basis; that is, a motif is used to classify a sequence in a binary manner. A disadvantage to this approach is that these statistics ignore information about multiple motif instances. However, our motif discovery method (Zero-or-One-Occurrence-Per-Sequence EM) doesn't model this explicitly during discovery (although it can be re-discovered after the fact). A counter-balancing advantage to these statistics, therefore, is that they are equally applicable to results from different forms of motif discovery with different assumptions about motif exclusivity on sequences.

To determine whether a motif classifies a sequence into the "bound" category, we convert the motif's frequency matrix into a PSSM, using a zero-order projection of our background model. We then use a simple thresholding scheme to indicate motif occurrence: 70% of the maximum possible PSSM score, given the (reduced) 0^{th} order Markov background model parameters. One occurrence of a motif classifies a sequence as "bound."

4.3.1 Result Probability and Significance

Next we calculate the number of motif occurrences M in the total intergenic region set of size N. If our bound set has size B, the number of observed sites m is a random variable with a probability given by either the binomial distribution:

$$p(m|B,M,N) = \binom{N}{B} R^m (1-R)^{B-n} \tag{4.2}$$

or the hypergeometric distribution:

$$p(m|B,M,N) = \frac{\binom{M}{m}\binom{N-M}{B-m}}{\binom{N}{B}} \tag{4.3}$$

where $R = \frac{M}{N}$, which is the observed rate of motif-bearing sequences across all intergenic regions.

In either situation, we alsc calculate a p-value by summing the probability density function over the "tail" of more significant results:

$$\text{p-value}(m|B,M,N) = \sum_{m'=m}^{B} p(m'|B,M,N) \tag{4.4}$$

Therefore, given a fixed sized bound set, we calculate two similar significance scores for any motif count result.

4.3.2 Motif Entropy

We also calculate the entropy of each discovered motif as a measure of its specificity.

$$E(\theta) = -\frac{1}{W} \sum_{i=0}^{W-1} \sum_{j=0}^{|\Sigma|} \theta_{ij} \log \theta_{ij} \tag{4.5}$$

4.3.3 Site-Based Error Rates

Finally, we calculate the probability that the discovered motif is the "correct" motif. This is done through the proxy of the "correct motif's" sites in the bound set.

The probability of being the "correct" motif is calculated in two parts: the ratio

of false positive and false negative rates for the known motif sites. Given a motif for a transcription factor, and the bound set of that transcription factor, we calculate the set of known sites M. For any motif θ that should explain that transcription factor's binding, we determine a similar bound set M_θ.

The set M_θ can be determined in one of two ways. The first way takes the top positions (by alignment score) from the final run of the EM algorithm and consider those points to be the "best" instances of the motif. The second approach converts the frequency matrix θ into a PSSM (a log-ratio matrix incorporating some low order perspective of the background model), and scores a match relative to an *a priori* score threshold. We take the second approach in this work.

Given a set M and a discovered site set M_θ, we then calculate the false positive and false negative rates:

$$f_p(\theta) = \frac{|M_\theta - M|}{|M_\theta|} \tag{4.6}$$

$$f_n(\theta) = \frac{|M - M_\theta|}{|M|} \tag{4.7}$$

4.3.4 Mismatches with "Correct" Motif

False positive and negative rates show how well the set of discovered motif sites matches the "known sites" of a reference motif. This is a model-agnostic way of approaching the issue, and can be useful in situations (for instance) where the sites are determined without use of a model, or when the reference motif model is not available.

If both discovered and reference motifs have available models, however, we can take the approach of directly comparison without the indirection of sequence sites. For instance, if both discovered and reference motif models are in the same probabilistic form we might compare them by calculating a Euclidean distance, or some form of relative entropy.

In this work, the reference motifs come in the form of regular expressions. Even

if we convert them to an equivalent probabilistic form, however, we also run into the problems of registration and orientation; there is no guarantee that the motifs we discover will be the same size and shape as the reference motif. We may also encounter problems if the discovered motif partially overlaps the reference motif, or is that motif's reverse complement.

Intuitively, both regular expressions and probabilistic motif models choose some subset of possible sites (either by "accepting" the relevant site's string, or by assigning it a non-zero probability). We have assumed, in this work, that we discover motifs of the right "size" (that is, length) for comparison to the relevant reference motif. Therefore, given the sequences of two sites, we may ask the question "How many mismatches do these sites have?"

Expanding this to a regime with regular expressions and probabilistic motif models, we ask the larger question, "What is the expected number of mismatches between a site drawn from the probabilistic model and a site accepted by the regular expression?" We assume that the regular expression chooses any accepted site string with equal probability. We solve the registration and orientation problems by minimizing over all possible overlapping shifts, and over both orientations.

If R is a regular expression, W the number of words accepted by R, Θ the probabilistic motif model, and Σ^A the totality of possible words accepted by either R or Θ, then we begin by calculating a score for the "shifted" match of R to Θ, given the shift of R (relative to Θ) by i letters:

$$S(\Theta, R, i) = \frac{1}{W} \sum_{w \in R} \sum_{w' \in \Sigma^A} \Theta[w'] \mathrm{Mis}(w, \sigma_i(w')) \tag{4.8}$$

Here σ_i is a shift operator on words, and $\mathrm{Mis}(w, w')$ counts the number of mismatches between w and w'. This mismatch count should be intuitive, although we note that when w' and w do not overlap exactly and one (or both) words contain unaligned letters, those unaligned letters are counted as mismatches. This fact means that mismatch scores are not comparable between motifs of different sizes.

The number S is then minimized over all possible shifts to calculate our score M:

$$M(\Theta, R) = \min_{i \in [1-len(R),\ldots,len(\Theta)-1]} S(R, \Theta, i) \tag{4.9}$$

We report the value of M (minimized over both strand orientations as well) as the expected number of mismatches. One technical detail of this calculation involves the wrinkle that the regular expressions we use to express known motifs are not "masked." This means that the mismatch scores of differently shaped motifs cannot be compared; this is not a problem however, since we only need to compare mismatch scores between motifs of the same shape discovered by different techniques.

4.4 ROC Curves

Hypergeometric and binomial probabilities and significance scores give some indication of how each motif is over-represented in the set of bound regions. However, this measure is also dependent on the (arbitrary) binding threshold used to gather these regions from the binding data. One interesting question would be to calculate these statistic for a different threshold: do our results improve when the threshold is loosened, or tightened?

An equivalent way of thinking about this problem was alluded to in the Introduction. The binding data, and any given motif, provide two different binary classifications on the set of possible regions. If we regard the binding data as roughly "ground truth," one indication of a motif's quality is how well its classification overlaps with the binding classification (or vice-versa). Given two such classifications, we could calculate the false positive (accuracy) and false negative (specificity) rates.

A Receiver Operating Characteristic (ROC) curve is a plot of these values for a parametrized classifier (against some fixed classifier). In this situation, the binding data is our parametrized classifier, and we wish to plot how well it matches the classification given by a discovered motif. We can vary the p-value threshold of binding continuously between 0.0 and 1.0, plotting `accuracy` and `1.0 - specificity` at each point. This is the ROC curve for a motif.

We will calculate these curves for all the known motifs, as well as for every dis-

covered motif (given in the Results section). This will not play a major role in our results and analysis, but the curves can give some indication of how well our results (or the known motifs themselves) generalize across different binding thresholds. The full tables of curves will be reported in the Appendix.

4.5 EM Algorithm Parameters

The EM algorithm for motif discovery requires the use of a number of parameters, such as the choice of starting points, the length of the iteration process, the method of sequential motif erasure, etc. In all procedural details, we try to follow the ZOOPS EM specification from Bailey et. al. as closely as possible [4]. However, we attempt to avoid model selection issues (each seed will take the pre-determined shape of the known motif for which we're searching), and we do include a higher-order background model.

We will also make the following choices for parameters to the EM algorithm:

- The motif mask for each seed is identical to the mask of non-wildcard (N) positions in the corresponding known motif.
- Each motif search is initiated with five seeds: one completely uninformative, and four additional seeds. Each of the four additional seeds is composed of identical base distributions θ_i, skewed (using pseudocounts) towards one of the four possible bases. (i.e., for the re-discovery of GCR1's motif, we use the seeds NNNNN, AAAAA, TTTTT, GGGGG, CCCCC).
- Starting at each seed, 15 sequential motif are discovered. Sequential erasure is carried out between successive runs for a single seed; the erasure arrays are maintained between different seeds.
- For all runs of the EM algorithm, we used a pre-calculated 2^{nd}-order Markov model for the background.

- For each run, the EM algorithm was iterated until pairwise Euclidean distance between motif models fell below a threshold of 1% the distance between the starting seed and result of the first iteration, or until 150 iterations had passed.

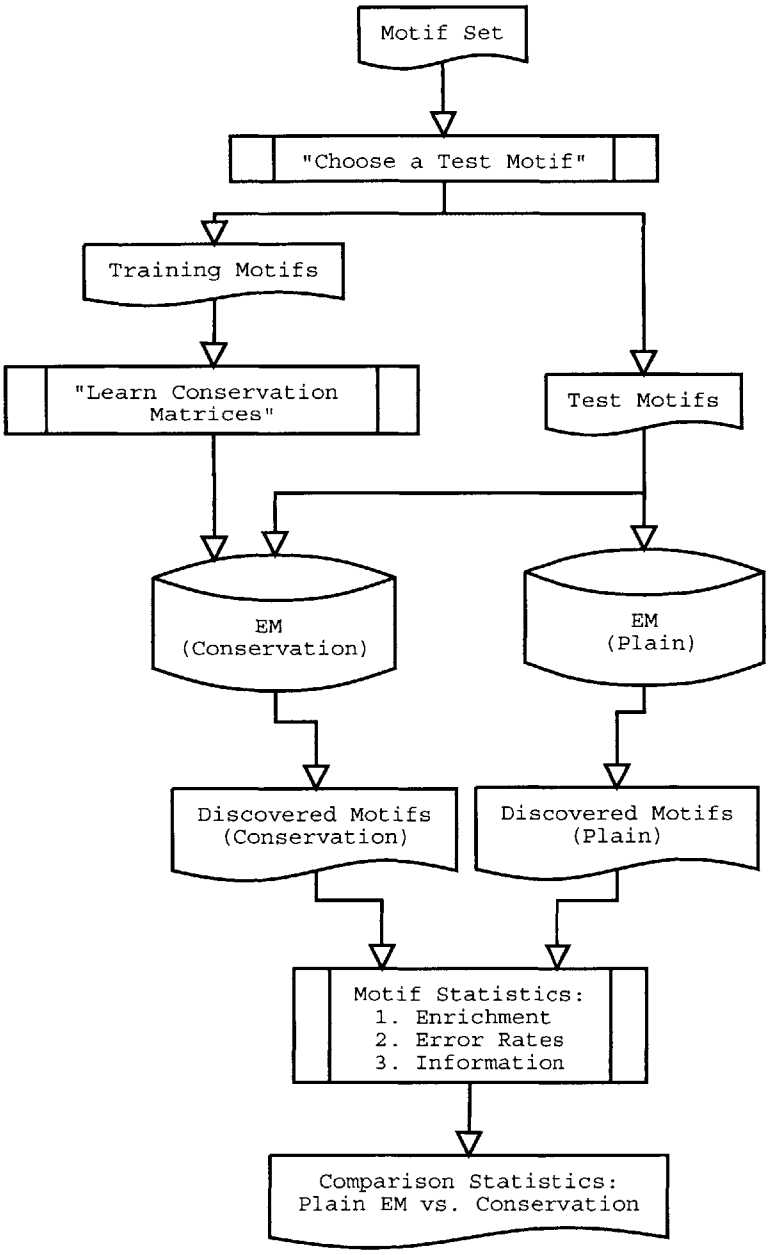

Figure 4-1: A diagram of the algorithm flow.

Chapter 5

Results

Our results fall into two conceptual categories: the results of calculating prior parameters before motif discovery, and the results of motif discovery using those parameters. The first category consists of the empirical parameters of our background and conservation models; the second is the list of discovered motifs for each factor, and the statistics associated with those motifs. We present only examples or summaries of results in each section; full tables of results are given in the appendix, Section A.

5.1 Background Model

We give the estimated parameters of the higher-order (2^{nd}) Markov model we use to model the background in Table 5.1. This shows the probability of seeing a given base (whose identity is specified by the column labels) in the a consecutive position after all possible combinations of two preceding bases (given in the two sets of row labels). For example, in this table, a `G` is seen 17.772% of the time, if the two preceding bases were `AT`.

5.2 Conservation Matrices

Our method depends on a representation of per-site conservation in "bound" positions. We calculate a joint distribution of bases (and gaps) across all K aligned

		A	T	G	C
A	A	0.42649	0.25508	0.17772	0.1407
	T	0.35432	0.32278	0.17316	0.14974
	G	0.34511	0.27682	0.18792	0.19015
	C	0.35722	0.30074	0.169	0.17304
		A	T	G	C
T	A	0.30443	0.36941	0.1507	0.17546
	T	0.22736	0.42623	0.16125	0.18516
	G	0.30168	0.31977	0.17239	0.20616
	C	0.31414	0.35122	0.14922	0.18542
		A	T	G	C
G	A	0.41512	0.26395	0.18311	0.13782
	T	0.3163	0.34069	0.196	0.147
	G	0.32055	0.27852	0.18917	0.21177
	C	0.33505	0.29836	0.17815	0.18844
		A	T	G	C
C	A	0.34854	0.2985	0.1766	0.17635
	T	0.24855	0.39206	0.17899	0.1804
	G	0.28437	0.30568	0.18855	0.2214
	C	0.3055	0.33015	0.16716	0.1972

Table 5.1: Markov Background Parameters

species at each position, for a matrix of frequencies with 5^K entries. These matrices are only relevant in the presence of an equivalent description of "background," or non-functional, site conservation. Examples of two such matrices are given in the following table, for the factor ABF1.

For display purposes, we have reduced the full matrices into three submatrices each; a submatrix represents the comparison of the base sequence (*Saccharomyces cerevisiae*) with one of the three other aligned species. The row labels are the observed bases in *cerevisiae*, and the columns indicate the possible values in the aligned sequence. The entry in row A and column T of such a matrix is the probability of seeing a T in the corresponding aligned sequence, conditioned on the probability of an A in the base sequence, marginalized over the values of all the other sequences, and normalized across possible values for the given aligned sequence.

ABF1 Motif Conservation

	-	A	T	G	C
A	0.05252	0.87735	0.02629	0.03212	0.01172
T	0.07136	0.02145	0.85431	0.01289	0.03999
G	0.03684	0.05189	0.01344	0.8894	0.00843
C	0.04125	0.01432	0.04759	0.02541	0.87142
	-	A	T	G	C
A	0.18659	0.74765	0.019	0.03066	0.01609
T	0.2097	0.03143	0.70742	0.02145	0.03001
G	0.16055	0.05858	0.00843	0.74229	0.03016
C	0.1474	0.03175	0.05393	0.01115	0.75577
	-	A	T	G	C
A	0.15016	0.71997	0.03941	0.06855	0.02192
T	0.15836	0.0514	0.67747	0.03143	0.08135
G	0.12711	0.06861	0.04186	0.73226	0.03016
C	0.15532	0.03967	0.06502	0.02858	0.71141

ABF1 Background Conservation

	-	A	T	G	C
A	0.09314	0.74758	0.04915	0.07907	0.03106
T	0.09135	0.04977	0.74434	0.03163	0.08291
G	0.09507	0.13873	0.04942	0.67805	0.03873
C	0.0906	0.05451	0.13906	0.04103	0.67479
	-	A	T	G	C
A	0.25279	0.55532	0.0659	0.08886	0.03712
T	0.25058	0.06887	0.55231	0.03977	0.08846
G	0.24988	0.16746	0.07027	0.46537	0.04702
C	0.25394	0.07378	0.16491	0.04812	0.45926
	-	A	T	G	C
A	0.22899	0.51293	0.08777	0.10895	0.06135
T	0.22493	0.09092	0.5106	0.06323	0.11031
G	0.23333	0.16787	0.08841	0.4389	0.07149
C	0.22497	0.0904	0.1696	0.07248	0.44254

5.3 Motif Discovery Results and Statistics

We summarize the results of our attempt to rediscover the ten known motifs in the Table 5.2. We give, for each factor and technique, the order in which the "best" motif was discovered (here, "best" is defined as the minimum mismatch score relative to the factor's known motif). In some situations, the best motif still has a high mismatch score and doesn't appear to be the correct motif; we list these cases using parentheses around the order value.

In one case (YAP1) both methods fail to rediscover the known motif. In two more cases, GCR1 and HAP3, the standard EM technique misses the known motif which the MEME+ technique manages to adequately recover. The full set of mismatch values, ordered by discovery time, from which this table is derived are found in Section A.4 in the Appendix.

Factor	*MEME+*	*MEME (Plain)*
ABF1	1	3
CBF1	1	1
GCN4	2	3
GCR1	5	(12)
HAP3	3	(12)
HAP4	3	46
MCM1	14	22
REB1	1	1
STE12	1	48
YAP1	(9)	(24)

Table 5.2: Motif Rediscovery Order Summary

If we know which discovered motif was the "best," we can then ask the question, "Where would we have ranked this motif without any prior knowledge?" As outlined in the Introduction, many motif discovery algorithms include a post-processing step to rank the algorithms results. One common ranking heuristic is "enrichment" in the input regions (relative to the background set), and a common measure of enrichment is the hypergeometric significance score (or p-value). For each of the "best" motifs indexed in the table above, we ask what the motif's rank is when the total motif

set is ordered by ascending hypergeometric score. The results are shown in table 5.3. We have also calculated the joint probability of the *Saccharomyces cerevisiae* sequence (using both the background model, and the "best" motif aligned at the final alignment vector output by the EM algorithm); the log-ratio of these probabilities (for the "best" motifs of both techniques, for each factor) is calculated, and shown in the table as well.

Factor	*MEME+*	*MEME (Plain)*	$\log(\frac{p_c}{p_{nc}})$
ABF1	1	1	-401.348
CBF1	1	1	-18.227
GCN4	1	1	-30.591
GCR1	17	49	33.450
HAP3	8	6	-24.643
HAP4	1	1	-17.837
MCM1	1	1	21.965
REB1	1	1	39.042
STE12	6	23	15.945
YAP1	71	26	-34.746

Table 5.3: Motif Scoring Order Summary

There are a variety of other ways to describe and compare the results of our two motif discovery techniques. The sheer number (in this case, 1500) of discovered motifs precludes listing them in a straightforward list. Furthermore, we have calculated over a dozen different statistics for each motif. We can also calculate metrics (such as the discovery time of the minimum mismatch score, from table 5.2) to describe the differences in *how* each technique found its results. Instead of exhaustively quoting these statistics, we will settle for giving the average of several common statistics, calculated for the "top" motifs and separated by factor.

Tables 5.4 and 5.5 show the average of seven key statistics over only the best 20 motifs, ranked by hypergeometric p-value, for MEME+ and Plain MEME respectively. The `Entropy`, `FalsePos`, and `Mismatch` averages depend only on the discovered motif; however, the `HG PValue`, `Bin PValue`, `Bound` count, and `Total` count scores depend on a binding threshold. For these latter scores, the standard threshold of 0.001 is used.

Factor	Bound	Total	HG PValue	Bin PValue	Entropy	FalsePos	Mismatches
ABF1	43.85	781.7	0.03315	0.00000	0.9234	0.9003	0.00095
CBF1	17.8	1,690.55	0.0368	0.03693	0.66149	0.866	4.90288
GCN4	41.3	3,092.35	0.04313	0.04103	0.62837	0.921	4.464
GCR1	11.95	4,720.5	0.73805	0.7382	0.2591	0.99583	3.10514
HAP3	16.45	3,355.5	0.23474	0.23496	0.4213	0.9475	0.94597
HAP4	28.6	2,663.9	0.01983	0.01499	0.52604	0.95676	0.94901
MCM1	26.8	1,263.45	0.02955	0.00000	0.39144	0.96696	0.07501
REB1	74.15	3,315.3	0.02081	0.0055	0.43709	0.91461	4.75924
STE12	37.2	3,425.95	0.0123	0.01236	0.55459	0.94878	4.03396
YAP1	30.85	3,669.55	0.11221	0.11276	0.44661	0.99286	4.7149

Table 5.4: MEME+: Average Statistics, Top 20 Motifs

Factor	Bound	Total	HG PValue	Bin PValue	Entropy	FalsePos	Mismatches
ABF1	46.95	718.85	0.02466	0.00000	0.8249	0.91657	0.00094
CBF1	20.65	1,800.4	0.0229	0.02296	0.70268	0.868	4.90461
GCN4	41.8	3,139.6	0.02655	0.02669	0.68032	0.962	4.72736
GCR1	11.5	4,635.8	0.75762	0.75778	0.23263	0.95833	3.0693
HAP3	19.3	4,146.4	0.27816	0.27865	0.35633	0.9675	1.01288
HAP4	32	3,314.05	0.08135	0.08171	0.48588	0.94595	0.94448
MCM1	24	928.15	0.03319	0	0.39927	0.96875	0.07808
REB1	70.65	2,987.35	0.01897	0.00578	0.46419	0.9	4.91699
STE12	38.35	3,415.85	0.0072	0.0073	0.57572	0.93902	3.99954
YAP1	33.25	3,655.35	0.07261	0.07297	0.5689	0.97857	4.42765

Table 5.5: MEME (Plain): Average Statistics, Top 20 Motifs

In the appendix, we give lists the top discovered motif lists for each factor (Section A.2), the average statistic tables for all motifs (Tables A.1 and A.2 in Section A.1), plots of entropies for all discovered motifs (Section A.5), and ROC curves for all known and discovered motifs (Figure A-21 and Section A.7, respectively)

Chapter 6

Discussion

We discuss the assumptions in our conservation model, the choice of statistics in analyzing the results of motif discovery both with and without conservation, and avenues for future work.

6.1 Model Assumptions

In deriving our new equation for sequence likelihood (3.27) that takes conservation into account, we made use of several assumptions. Some of these, such as the assumption that the probabilities of motif and background sequence are independent or the assumption of column independence in the product multinomial motif model, are shared with other motif discovery algorithms. They may not be valid in all situations, but our algorithm (founded on those assumptions) will remain valid in exactly the situations which are valid for other algorithms.

The *conservation isotropism* and *conservation independence* assumptions (eqs. 3.17, 3.18, 3.19, 3.16, 3.15) are specific to our technique and to the conservation data, and therefore warrant justification. The first part of the *isotropism* assumption, expressed in the equation $P(\mathcal{A}(S_i)|S_i, e, \Theta) = P(\mathcal{A}(S_i)|S_i, e)$, is the conditional independence of an aligned sequence from the parameters of the motif model, given the reference sequence and the knowledge that the site in question is drawn from the "functional" distribution.

In other words, when it comes to conservation, there are no "special" motifs. If a particular region is a motif site, it will be conserved across species with the same probability as all other sites for the same motif, as well as sites for other motifs. This is, in practical terms, one of the most important new assumptions we make: conservation can be estimated (in both putatively functional and non-functional regions of DNA) prior to any motif discovery. The base-by-base conservation of the reference sequence with the aligned species' sequences provides a static re-weighting of the data to sharpen our motif discovery.

Relaxing this assumption would imply the assumption of a functional relationship between the motif model parameters and the conservation probability. This approach would assume the use of a more complex model, a problem which would be exacerbated by the relative scarcity of known motifs to learn the larger number of parameters.

The *independence* assumption is expressed in equations 3.15 and 3.16. We choose to make this assumption for two reasons. The first might be termed "aesthetic": the independence of conservation across different positions of the same site mirrors the position independence assumption implicit in the product multinomial motif model.

Of course, we could keep the form of the motif model's likelihood (maintaining the latter position independence assumption) while modeling the conservation of a K-mer as a whole and without further independence assumptions. This approach would, of course, run into the same problem mentioned in our discussion of isotropism: the scarcity of known motif sites is still a problem.

The second part of the *isotropism* assumption is expressed in the equation $P(A(S_i)|S_i, e, \texttt{pos} = j) = P(A(S_i)|S_i, e)$. Having factored the likelihood of conservation in a potential site into the product of likelihoods at each position in the site, we assert that relative position within a site doesn't affect conservation.

One can imagine a situation where this is not a valid assumption: perhaps motifs are uniformly better conserved in their centers than at their edges, or perhaps their conservation depends on their orientation relevant to the downstream gene (for instance, the downstream edge of a motif might be consistenly better conserved than

the upstream edge).

Simple tests of both these questions were performed: in general, known motifs were too short to contain significant changes in conservation between the edges and center, and no significant bias in direction of conservation was observed. In avoiding unsupported hypotheses about conservation, this second *isotropism* assumption allows us to learn a simpler conservation model with more data.

6.2 Data Analysis

One objection to our model of conservation is that it makes too many independence assumptions. Specifically, we have built in the assumption of independence between conservation probabilities at different positions from the very start. This assumption is implicit in using "alignment functions" A_k to indicate base-to-base matchings between species, and in the inclusion of the "gap" symbol in the alphabet Σ.

These assumptions certainly ease many of the technical difficulties associated with the data. If we take each aligned position as a separate (independent) data point when learning the conservation models, we avoid some problems associated with having between 10 and 20 known motifs to work with.

We have also side-stepped any consideration of the global alignment technique used to generate the conservation data. In general, our approach is orthogonal to the underlying alignment algorithm: it will work no matter what the alignment algorithm was, although its success depends on the accuracy of that alignment. An assumption of independence between positions glosses over local irregularities in the alignment. We presume only that the preponderance of positions from the reference sequence are correctly aligned, and the conservation model will be accuarately estimated.

We can easily spot a problem with this approach. It is possible that a site of a known motif in the reference sequence might be gapped, or aligned with a gapped region. In this case the corresponding site in the aligned sequence will be too short or too long, and it's not clear what that should mean for finding evidence of "conservation" for the reference site.

A second objection to our interpretation of results could be with the use of a "mismatch" score. Clearly, in comparing the ability of two methods for re-discovering the same motifs, we must establish some metric for accuracy. Our "expected number of mismatches" score seems to reasonably capture an intuitive idea for the number of site-strings simultaneously matched by a scoring matrix and a regular expression. In the absence of such explicit models, however, it would be necessary to use a site-based error model (for instance, the site false-positive and false-negative rates outlined in Section 4.3.3). Furthermore, any method which relies on averaging across samples from a probabilistic model will be sensitive to relative differences in the information introduced in that model by the methods being compared.

A third objection with this method might arise with the small number of factors tested for re-discovery. Our confidence in the success of adding conservation will grow as the number of "difficult" motifs which are re-discovered with the new data grows. However our method relies on both a relatively small set of known motifs from TRANSFAC, and on the ability of the location data to indicate a set of "motif-bearing" intergenic regions: we did *not* rely on the presence of a motif, or any other data set, to choose the regions for input to the motif discovery algorithms. We were therefore restricted to choosing, as test cases, those factors whose known motifs and bound regions intersected. As the quality of the total known motif information in *Saccharomyces cerevisiae* grows, this restriction can be relaxed. In the meantime, we feel that the noticeable improvement of MEME when conservation is added gives us hope for more comprehensive tests in the future.

6.3 Future Work

Some directions for future work are based on the choices we outlined in Section 1.1. For instance, we have used a simple form of EM: the ZOOPS, or "zero or one occurrence per sequence," model. One generalization would be to use a more complicated (and noise-resistant) method of evaluating the motif model. We could also substitute the randomized Gibbs sampling method for the deterministic EM

algorithm; our modifications to the likelihood equations in Section 3.7 should be agnostic to the choice of evaluation algorithm.

The second direction for improving the results of our method would be to improve the way we choose the EM starting points. We've attempted to ignore these issues by choosing several equally spaced (and equally uninformative) starting seeds. We have also side-stepped any model selection issues, explicitly choosing to learn motifs with the correct width. We do this because issues of model selection have been dealt with elsewhere, and are separate from the improvement brought by conservation.

A third avenue for future work would be to increase the complexity of either the motif or background models. For the background, we could try using even higher-order Markov models. For the motifs, we might not make the assumption of complete independence between separate positions. This would work well hand-in-hand with a similar relaxation of the corresponding assumption for conservation. We imagine a method that simultaneously learns a motif model with correlations between sequential positions, and a conservation model that pays attention to the total conservation score of a W-word, as well as the possibility of gaps in the aligned species.

Another way of expanding the motif model is to introduce parameters describing spatial arrangements and relative orientations. Current motif discovery tools are often generalized to discover multiple motifs. These generalizations rarely make any assumptions beyond the mutual exclusion of sites in the same sequences (an assumption we have made here). However, the discovery of motifs through analysis of spaced "dyads" has been explored [44]. Other tehniques used deterministic algorithms for finding variably-spaced motifs [17]. These are steps toward discovering motifs with considerations of mutual spacing. Motif signals which would otherwise be too weak to be discovered on the basis of sequence alone may become significant through their non-random arrangement with respect to other motifs. Similarly, we can imagine motif models that account for non-random orientation or spacing with respect to the regulated gene.

Another direction for attacking the "multiple motif" question in the context of conservation might be to discover simple (single) motifs using conservation, and then

continue by assembling more complex motifs out of these conserved building blocks.

Other directions for future work lie in relaxing the assumptions of Section 3.7. One major assumption implicit in our model was that of independence between the probability of conservation and the (current) motif model (eq. 3.17). Relaxing this assumption is a straightforward way to extend this work. Known motifs could be used to assess a functional form for $P(\Theta|\mathcal{A}(S_i), S_i)$, or a functional form could be assumed and known motifs used to learn the parameters of that function. The clear generalization of considering each set of aligned bases from a site to be a product of independent samples from the corresponding motif position is an example of such a "functional form."

Extending the work in this way also depends on the semantics behind the probabilistic motif model learned with the EM or Gibbs procedure. One logical way of looking at the product multinomial model is that it expresses an ambiguity on the part of the transcription factor itself: a model might assign equal probability to two different sites simply because the factor binds those two sites with nearly equal frequency. In this situation, our assumption of independence seems much too strong. We would expect that knowledge of the motif and reference base should predict the corresponding position in other species more surely than only knowing the reference base. (In other words, what's being preserved is the status of "this site is bound" across species, and what matters is whether our model's ambiguity means that this "true" site status is ambiguous, or only our knowledge of the site.)

On the other hand, the uncertainty in the motif model could reflect *our* uncertainty in what sequences are bound by the motif's factor. In this case, there might be relatively few "true" bound words which are only noisily described by the discovered sites. With these model semantics, assuming conditional independence of conservation and motif models is more reasonable.

Another direction for relaxing conservation might be to relax the base-to-base alignment assumption in a method that looks for coherent motifs from other species in corresponding (or closely corresponding) positions.

Yet another direction would be to introduce a more elaborate model of evolution.

Right now, with our multiple alignments, we've assumed a simple "one root, multiple leaves" tree form of evolution. However, some methods of alignment construct a tree simultaneously [45]; these differently-shaped phylogenies might affect our model of conservation, that is, the functional form of Ξ.

One final avenue for future work is in the direction of relaxing the strict "base-to-base" alignment assumptions we have made here. This method would assume that region alignments are given, not as mappings of positions in the reference genome sequence to bases in aligned genomes, but as indications that corresponding regions from two genomes should contain similar motif structure. Such a "region-to-region" alignment, followed by the discovery of a motif (or motif constellation) within one region, would bias the search towards a similar conclusion in aligned regions. In other words, we would maintain a correspondence between regions while eliminating the $\mathcal{A}$ base-to-base alignment information.

This generalization is *not* possible in the setting of single-motif OOPS EM setting. However, the direction of modification to Gibbs sampling and more complex forms of EM (ZOOPS and TCM) is immediately obvious: if the loose alignment indicates regions i and i' correspond, then knowing S_i contains a motif should influence the prior per-site (or, in ZOOPS, per-sequence) probability of discovering a motif.

Chapter 7

Conclusions

It would be difficult to conclude that adding the conservation data *dramatically* changes negative results into positive: this is, in part, due to the fact that the training examples are fairly easy. In six out of ten examples, the plain MEME algorithm operating on the 0.001 bound regions and without help from conservation will manage to find the known motif within 10 or 15 motif-iterations (starting with the uninformative prior motif).

However, we do conclude that adding even such a simple model of conservation to motif discovery is useful. We make this claim in three parts. First, adding conservation data does not *harm* the results: this is a necessary and non-trivial conclusion. Adding an additional bias does not, in the test cases we have seen, drive the results of the algorithm away from the correct answer. Furthermore, the motif discovery time statistics bear out this conclusion. With the possible exception of YAP1, adding conservation does not *delay* the discovery of the correct motif; when the plain MEME algorithm finds the correct motif immediately, so does the MEME+ algorithm.

Our second conclusion is that adding the conservation data improves the rediscovery of known motifs. In the the case of GCR1 and HAP3, MEME+ rediscovers the known motif where standard MEME fails. In the cases of HAP4, MCM1, and STE12, the enhanced MEME+ algorithm substantially improves the order in which the correct motif is rediscovered. Only YAP1's motif presents a serious problem for both approaches.

These conclusions depend on the calculation of the mismatch score (this score correlates well with low false positive/false negative site counts, as we would expect). One way in which this score could be deceiving is if the MEME+ technique returns "weaker" motifs that matched a larger set of strings, and therefore were more likely to have lower mismatch scores. Our third conclusion is that this is not the case: the best motifs (in terms of mismatch score) returned by MEME+ are comparable in entropy to the corresponding best motifs of plain MEME. Another way of thinking about this is that the traditional definition of "conservation" (conservation of base identity across aligned sites, instead of aligned species) is maintained with the addition of sequence conservation data.

Of course, as mentioned in the Discussion, adding a better understanding of conservation should improve these results. However, our work has outlined a general framework for adding this data to existing techniques in a simple way. We have shown its utility, and argued that the exploration of further improvements to this approach is warranted.

Appendix A

Appendix

A.1 Total Motif Averages

Factor	Bound	Total	HG PValue	Bin PValue	Entropy	FalsePos	Mismatches
ABF1	37.09333	915.28	0.33048	0	1.15918	0.93494	0.00106
CBF1	16.46667	2,320.38667	0.12244	0.1229	0.23188	0.9408	5.69447
GCN4	40.18667	3,341.08	0.07555	0.04508	0.34416	0.96773	5.02476
GCR1	11.77333	4,606.72	0.60891	0.60911	0.22322	0.96444	3.30644
HAP3	15.54667	3,519.76	0.41525	0.41537	0.17409	0.95133	1.07731
HAP4	27.89333	2,844.58667	0.03838	0.03086	0.2052	0.9618	1.07793
MCM1	24.48	1,364.8	0.05366	0.00006	0.26358	0.9669	0.08153
REB1	74.08	3,461.44	0.05729	0.03464	0.25882	0.92794	5.11886
STE12	31.93333	2,996.22667	0.03114	0.01275	0.27503	0.98341	4.83941
YAP1	30.05333	3,441.32	0.05154	0.05193	0.24174	0.99581	5.16696

Table A.1: MEME+: Average Statistics, All Motifs

A.2 Discovered Motif Lists

For each factor, we show the list of the top 20 motifs discovered, sorted by their (Hypergeometric p-value) score. Motifs discovered with and without conservation are listed side-by-side; each motif is provided with its mismatch score as well, to indicate how close it is to the "real" motif (although this mismatch score is *not* used to sort the motifs in these lists).

Factor	Bound	Total	HG PValue	Bin PValue	Entropy	FalsePos	Mismatches
ABF1	36.36	821.22667	0.32537	0	1.13686	0.90675	0.00106
CBF1	17.42667	2,397.37333	0.12811	0.12856	0.25223	0.95627	5.70826
GCN4	41	3,427.45333	0.06737	0.06284	0.25874	0.97467	5.12889
GCR1	11.56	4,487.30667	0.60591	0.60611	0.18103	0.94889	3.36502
HAP3	16.92	3,757.16	0.38955	0.3898	0.21597	0.98533	1.08585
HAP4	31.34667	3,519.98667	0.08103	0.08168	0.17862	0.98523	1.0772
MCM1	24.06667	1,156.77333	0.0519	0.00002	0.30158	0.95833	0.08259
REB1	74.88	3,497.50667	0.06237	0.04689	0.25852	0.94772	5.18975
STE12	37.17333	3,513.49333	0.01282	0.01304	0.30612	0.98114	4.83652
YAP1	31.90667	3,685.66667	0.12644	0.12692	0.28153	0.99276	5.01991

Table A.2: MEME (Plain): Average Statistics, All Motifs

MEME+ (Conservation)			MEME (Plain)		
Motif	HypG Score	Mismatch Score	Motif	HypG Score	Mismatch score
tCGTgcaaaGTG	0.0000000	0.0006283	TCACtttgtACG	0.0000000	0.000141
gCCctgCGcaaA	0.0000000	0.0010637	GtACcaAAaCAC	0.0000000	0.001124
aAAatTTTTCAg	0.0000002	0.0011856	tAGCgaTAtGgC	0.0000000	0.000983
agcGaTGagctg	0.0000007	0.0010883	aAAAtTTTTCAg	0.0000000	0.001208
GgCGCgCcgACC	0.0000043	0.0009479	CCtCCccTGCCC	0.0000000	0.000994
GgtGAGgtGaGC	0.0000076	0.0009959	caAGCcAAGCag	0.0000000	0.001094
AGaggtTaAGCa	0.0000287	0.0010036	CAgCaatCAcCa	0.0000000	0.000872
cGCaaAgggCGC	0.0000329	0.0008814	aaaatTtCAGca	0.0000000	0.001041
cGaGtTgGcgGA	0.0000401	0.0007892	AgGctTTtGaAA	0.0000000	0.001076
GCcgAcCctCGg	0.0001084	0.0010059	GAgtgCgcTgGc	0.0000002	0.001154
GcgGGgGcAGGG	0.0001099	0.0009764	AcGcAGtTcAGa	0.0000003	0.001031
gCcaccaCAccc	0.0001637	0.0009392	CAgcTTtGAATc	0.0000005	0.001047
ggggGaaaagtG	0.0002372	0.0010568	GCaacggtttcA	0.0000007	0.001003
AccggACTcaAA	0.0003391	0.0011359	tcgGCgGCtatT	0.000001	0.000976
GcACGagcgGCC	0.0004277	0.0008361	gAAAAgtGaaAa	0.0000014	0.001156
GGAAtgaGTaAa	0.0005615	0.0010579	ctTcCTTCCtCa	0.0000019	0.000937
cActgAaAcaaa	0.0010155	0.0010084	gGGTtaaTcAGG	0.0000022	0.000937
TCacattTTcgc	0.0018235	0.0009403	aacatacAtaCa	0.0000109	0.001030
gaAaCagTagAg	0.0023559	0.0011039	cCTtCttGGCTT	0.0000292	0.000986
ggAAacaGCCgc	0.0031745	0.0010532	gCAgCaaCaaCA	0.0000422	0.000798

Table A.3: ABF1 Motifs

MEME+ (Conservation)			MEME (Plain)		
Motif	HypG Score	Mismatch Score	Motif	HypG Score	Mismatch Score
GTCACGTG	0.0000000	1.2538556	CACGTGAC	0.0000000	1.3771077
CACGTGAC	0.0000000	1.4890094	CACGTGAC	0.0000000	1.5022658
TCACGtGg	0.0000000	3.7555067	CACGTGAC	0.0000000	1.4359919
ggCaGCAC	0.0000000	5.8444582	CtCaGCgC	0.0000000	5.5830942
CagtGcGC	0.0000021	5.5199781	GGTgagGG	0.0000000	5.7226663
GGGcgGgG	0.0000174	5.7534491	CgGTgCGG	0.0000000	5.7331878
GGtgCgGt	0.0000558	5.9753679	GGaAgGGc	0.0000001	5.8783685
gGtGGcAa	0.0000861	5.0225057	CAatCAGt	0.0000001	5.8053504
GtCCGTGA	0.0001192	5.2052652	GGtTGCAC	0.0000002	6.2552329
GAtGGGGc	0.0001813	4.5621774	AGCaccAg	0.0000043	5.281844
CCcTGagC	0.0005692	5.7728602	AaatGcGg	0.0000086	5.7118556
agatGaGG	0.000633	5.9167638	aACTTtac	0.0000104	4.6362353
aaCtGCAa	0.0008448	5.7046676	CTCCtcCc	0.0000158	5.2096718
caAGcCAC	0.0009675	4.4642088	GCGGCcAA	0.0000243	5.7092698
aCCaaCAA	0.003547	5.8858861	GaGGgAGC	0.0000299	5.3981832
tAacCGAa	0.0042486	4.9875228	CgGaaaGC	0.0000936	6.1504036
gCAgCGGC	0.0080837	5.7620363	AAAtcCAg	0.0001044	5.585007
GaGCAaAA	0.0194321	6.1320402	gGactGgG	0.0001329	5.4340025
aGAagCac	0.0304537	6.2675328	TGAaTGaG	0.0001741	5.6741139
TggagAag	0.0602283	5.7268902	AaGtGGAA	0.0003632	4.9209064

Table A.4: CBF1 Motifs

MEME+ (Conservation)			MEME (Plain)		
Motif	HypG Score	Mismatch Score	Motif	HypG Score	Mismatch Score
gTGACTC	0.0000000	2.5389899	GAGTCAt	0.0000000	2.7967652
GAGTCAc	0.0000000	2.6839278	gGCGATG	0.0000000	5.0077149
GCCGCCG	0.0000014	4.3111174	GGCGCCG	0.0000000	4.9626583
cgGtGCG	0.0000149	5.5463455	GcGcgGG	0.0000000	5.6009077
gAAgaGC	0.0000378	5.0792734	GGgtTGa	0.0000383	4.1236652
CctGCgt	0.0002895	5.3609233	GGAAGcg	0.0000582	4.3316198
cAaCcCg	0.0003208	4.9041661	tgCGcaC	0.0001986	4.5893244
CtgaTtg	0.0003738	5.1029782	aTATAtA	0.000292	5.0090251
TgCGaGG	0.000557	4.4579924	cCAGCGC	0.0002945	5.2015533
CgaAtcA	0.0009255	4.0678108	CCTTaAG	0.0004087	5.9806072
aGcgGgA	0.0009441	4.467521	GCgCaAA	0.0006238	4.7540503
caAGTGA	0.0017248	3.4549858	AAAGGGA	0.0026221	3.8512306
CcCtctC	0.0022732	4.965524	cGttgGA	0.0032181	4.8869516
AAagGCA	0.0029695	4.5307216	AAAAGAA	0.0033411	5.0409905
ctCAgCa	0.0036912	4.9168871	TaTATAt	0.0042288	4.9263834
GctcGCG	0.0047662	5.1207109	AagtGAA	0.0098064	5.392566
gcgggag	0.0060971	5.2255149	aatGAAA	0.0120353	5.1232835
CCGcaAA	0.0065299	4.7298977	AAAGAAA	0.0132137	4.7552144
TTcCaaG	0.0070612	5.1210328	GcAaAaG	0.0165156	5.5643397
gcgggag	0.0072722	5.2250842	cgcgcgc	0.0192069	5.2536613

Table A.5: GCN4 Motifs

MEME+ (Conservation)			MEME (Plain)		
Motif	HypG Score	Mismatch Score	Motif	HypG Score	Mismatch Score
TTGAG	0.0043499	3.2173474	cgcgc	0.0728085	3.7463096
cGGtg	0.0322537	3.1922068	cgcgc	0.0728085	3.746312
tGCgC	0.0478316	3.5322627	cgcgc	0.0728085	3.7463149
gGgtg	0.1389592	3.4503993	cgcgc	0.0728085	3.7463174
caGgg	0.1437718	3.532605	cgcgc	0.0728085	3.7463191
ggggg	0.1474809	3.3271428	cgcgc	0.0728085	3.7463212
ggggg	0.1600376	3.3905846	cgcgc	0.0728085	3.7463235
GCGGc	0.1606811	3.2456304	cgcgc	0.0728085	3.7463255
ggggg	0.1632118	3.4275735	cgcgc	0.0728085	3.746328
ggggg	0.1632118	3.4521464	cgcgc	0.0728085	3.7463305
gggtg	0.1876178	3.3073524	ggggg	0.118734	3.405019
AaGcT	0.1891834	4.3467173	ggggg	0.118734	3.4362444
ggggg	0.1961073	3.4677918	agGaA	0.1923376	3.1483346
gggtg	0.1993806	3.3296989	ggggg	0.2021465	3.4581064
ggggg	0.2004828	3.304208	ggggg	0.2021465	3.4779795
gggtg	0.2243453	3.37802	ggggg	0.2946543	3.2946072
GCaAG	0.2286403	1.7552734	ggggg	0.3391421	3.5182718
gggtg	0.2618573	3.3576527	ggggg	0.3454571	3.5680929
gggtg	0.2726417	3.4019814	ggggg	0.3454571	3.569564
gaGcg	0.3107719	3.5214411	ggggg	0.4100027	3.2978872

Table A.6: GCR1 Motifs

MEME+ (Conservation)			MEME (Plain)		
Motif	HypG Score	Mismatch Score	Motif	HypG Score	Mismatch Score
CgacTGT	0.0000512	0.7859438	CcGgaCC	0.0000351	1.0499719
TCTTCaC	0.0017302	1.0157905	CcCTtcC	0.0000418	1.0280708
AcCCGaA	0.0021599	1.1627359	AAATGCA	0.0000453	1.0012489
cGtACCC	0.0025422	1.143745	agGGAgG	0.0001135	1.0849801
CttTcTC	0.0035598	1.1292835	AGtTcAA	0.0001683	1.1162281
aGGggGa	0.0045729	1.0031318	CCCggTT	0.0002038	0.9257034
TGCaTtT	0.0050598	0.9684994	gCTGCaC	0.0002085	1.0633881
TcATTGG	0.0091423	0.2505321	CGTaCcC	0.0002711	1.1287432
tTTCCTT	0.0110132	1.1802055	gGGtacc	0.0038762	1.2455337
AaATAaC	0.0181119	1.0115547	AAaTgCG	0.0043234	0.9312306
GCAttta	0.0200317	0.6387078	AAtTtaA	0.0045207	1.0052966
caAGtTT	0.0289353	1.023051	AacTGcA	0.0054226	1.1270701
GctGAAA	0.1301338	1.1083257	GCtgAAa	0.0101985	1.0921387
atatata	0.1549358	0.9817326	AAaGAAA	0.1006442	1.0801467
aCaaagA	0.237374	0.6851173	aaagaAA	0.1240161	1.028764
TgTaCAt	0.240208	1.1002173	aaagaAa	0.1589801	0.9880949
atatata	0.2894488	1.0272125	AAagaAA	0.1735976	1.0685384
aaAGGGG	0.3071024	0.7897033	aaagaAa	0.2094648	1.0022772
gggggggg	0.5182324	1.121442	aaagaAa	0.2458606	1.0104638
gggggggg	0.5182324	1.1214605	aaagaaa	0.2919352	1.0196358

Table A.7: HAP3 Motifs

MEME+ (Conservation)			MEME (Plain)		
Motif	HypG Score	Mismatch Score	Motif	HypG Score	Mismatch Score
TGaTTGG	0.0000000	0.2155934	TGaTTGG	0.0000000	0.1688499
GgtCCAA	0.0000006	1.2779894	CTGCcGC	0.0000028	1.1640292
GGagCGC	0.0000051	1.0727802	CCcGCtt	0.0000029	1.0866546
GgGCGGC	0.0000063	1.061745	tcAATTG	0.0000033	0.6640833
GgGtAAA	0.0000139	1.2200915	TTgAaCc	0.0000064	1.1217152
cgTTTAA	0.0000879	0.9194723	GGActGG	0.0000073	0.6906105
cCCCcgC	0.0002211	1.0714883	tgCTTcC	0.000009	0.8517307
GttCTTT	0.0002792	1.090879	TTtAAcT	0.0000164	1.1990826
TCaATTG	0.0002973	0.7450138	TtgGTtC	0.0000987	0.9088252
GGacGGG	0.0003674	0.7688436	AtccTTC	0.0015499	1.1059885
gCaTcTc	0.0004792	0.9247769	TtcATTT	0.002286	0.9174646
gggggcg	0.0004998	1.1009904	tTCCTtG	0.0051877	0.9994358
ggggggg	0.0030559	1.1061946	gAAAGAA	0.0281187	0.8104999
ggggggg	0.0030783	1.1047471	TgTATaT	0.0453342	1.0418691
ggggggg	0.0037682	1.1021094	cccgccc	0.0839595	1.1222117
ggggggg	0.0038499	1.1028262	cccgccc	0.0839595	1.1222107
ggggggg	0.0042824	1.1044277	gggcggg	0.0839595	1.1225382
ggggggg	0.004533	1.1094042	gggcggg	0.0839595	1.1225395
ccccccc	0.0047919	1.1199402	gggcggg	0.0839595	1.122541
ccccccc	0.0048613	1.1201409	gggcggg	0.0839595	1.1225422

Table A.8: HAP4 Motifs

MEME+ (Conservation)			MEME (Plain)		
Motif	HypG Score	Mismatch Score	Motif	HypG Score	Mismatch Score
CCtaAtgtGG	0.0000000	0.0421805	ccAAATTtAG	0.0000000	0.0666633
aatTtCCcga	0.0000000	0.075271	cgAtTTGAgG	0.0000000	0.0718743
gacGAAAAag	0.0000000	0.0735646	GGAAAtttcc	0.0000000	0.084998
gGCGcGtGtC	0.0000000	0.0960754	gAAaaGGGAA	0.0000000	0.0920103
GAAAtgTgcc	0.0000000	0.082774	gcAACttGgC	0.0000000	0.0739202
tGgTGGCTgg	0.0000000	0.0795302	GtGaaTGCga	0.0000000	0.0838056
AAAgGGGAAA	0.0000000	0.0932764	GGTAAgTaCA	0.0000000	0.0816083
aatcGGGAAA	0.0000000	0.091471	caAAgTGaAa	0.0000000	0.0856098
CtTGtAaATT	0.0000001	0.0752194	AAAgcAGGAa	0.0000000	0.0938164
TccAAcgAaa	0.0000001	0.0809295	aATgaATGCA	0.0000000	0.0849316
GGtaAtgCAa	0.0000001	0.0847265	TCAgatCAAG	0.0000000	0.072793
TTTccactTC	0.0000002	0.0862853	AgATCaGGAA	0.0000000	0.0901984
CatGatacAG	0.0000006	0.0666297	AAgaGggaAA	0.0000000	0.0936776
ccgAgGCatG	0.000001	0.0700293	aaatTaCCCa	0.0000000	0.0834874
GCGGgTAGga	0.0000039	0.0707149	GaAACgCTaA	0.0000000	0.0943082
gAacCctCGA	0.0000043	0.0837106	GatCtTTaAa	0.0000000	0.083775
gAaAagtGCA	0.000019	0.0898458	GaAAcgtGCA	0.0000000	0.0826903
aCTtaAAaaG	0.0000252	0.0630626	GCAggaGCgg	0.0000000	0.0767648
gAtcGgGCAt	0.0000254	0.0971966	GCggCaGgAA	0.0000000	0.0779035
gTTTAcgTtt	0.0000362	0.087075	GGtaACCtAA	0.0000000	0.0875258

Table A.9: MCM1 Motifs

MEME+ (Conservation)			MEME (Plain)		
Motif	HypG Score	Mismatch Score	Motif	HypG Score	Mismatch Score
TACCCGG	0.0000000	2.7538916	GGGTAAC	0.0000000	3.1767838
GGTAAcg	0.0000000	5.4318566	CCCGGat	0.0000000	4.586193
gtaCCCG	0.0000000	2.833123	cACCAAa	0.0000000	5.481346
CaCcgAa	0.0000003	4.7471576	cAgCtCA	0.0000002	4.8586944
GAGGgaG	0.0000201	4.8593045	gGCAGgg	0.0000026	5.1500161
AtgGaTG	0.0001053	5.2588692	CAactGC	0.0000035	5.1656689
GAaGaGG	0.0001577	5.3398974	GCtGgGC	0.0000039	5.3738754
gggggggg	0.0003063	5.1622467	GgTgaGC	0.0000133	5.4252434
gGcgCgG	0.0004239	4.6363894	attTGAG	0.0000152	5.5761332
ccgcgcc	0.0004667	5.2407539	gAAAGGa	0.0000199	5.6751924
ccgcgcc	0.0004667	5.2407739	GgAAGCt	0.0000271	4.8444961
ccgcgcc	0.0004667	5.2407999	cAtTGCA	0.0000853	5.2859548
ccgcgcc	0.0004667	5.240827	GCAgGaA	0.000148	5.3493053
ccgcgcc	0.0004667	5.2408533	GAgGcGG	0.000221	4.6135597
ccgcgcc	0.0004667	5.2408799	caaaTGC	0.0002427	4.7879039
ccgcgcc	0.0004667	5.2409045	CCCGtAC	0.0002839	4.2385961
ccgcgcc	0.0004667	5.2409303	ccccccc	0.0003398	5.2154507
ccgcgcc	0.0004667	5.2409559	ccccccc	0.0003932	5.2278655
ccgcgcc	0.0004667	5.2409899	cgggggc	0.0004667	5.2484559
ccgcgcc	0.0004667	5.2405694	cgggggc	0.0004667	5.2484562

Table A.10: REB1 Motifs

MEME+ (Conservation)			MEME (Plain)		
Motif	HypG Score	Mismatch Score	Motif	HypG Score	Mismatch Score
cTgAAaC	0.0000000	3.3920029	GCAcCag	0.0000000	4.0822095
GCaCCcG	0.0000000	5.0507502	GAAACtc	0.0000000	3.1710765
tCGAgCc	0.0000004	3.9629193	ccctGAA	0.0000000	4.4680393
AAaTTTG	0.0000009	4.7009221	GGAAAaG	0.0000001	3.578633
CGaACtg	0.0000015	3.8541279	GAAgtGG	0.0000001	5.0909535
TGAAACa	0.0000056	1.2385597	aAGTTCg	0.0000001	3.8921054
AAGgGAA	0.0000379	5.649683	gAAtGCc	0.0000002	4.5814121
GAataaG	0.0000634	4.8773301	GCggGCA	0.0000005	4.8027963
aTTTCGC	0.0001079	4.6792229	AatTTTG	0.0000013	4.2264882
AAAtGcA	0.0001324	4.6944774	AAGaGcA	0.0000022	4.4743445
CagcTtC	0.0001483	5.0621322	AAGCTGA	0.0000051	4.1932898
CAcTGaa	0.0002176	5.1291829	AaGtGAA	0.0000079	5.1624696
AcgtcaA	0.0002428	4.9720475	AAatGCA	0.0000155	4.5624419
GGAAAgg	0.000262	3.3520416	TTCAgCt	0.0000214	4.0833908
tgCcCGc	0.0003458	5.1894324	GcatGaG	0.0000243	4.8823451
ccGCGCG	0.0003645	5.5546213	tGGCtTc	0.0000292	4.3904602
atAaAAG	0.0005254	4.0398275	AAaTcaA	0.0000385	4.2780871
AaGcggt	0.0007	5.3289558	GCtcgAa	0.0000428	5.2120536
aCctGCa	0.0010905	4.1360419	CAAgAAG	0.0000711	4.8846896
ctCCtga	0.0018712	4.6022306	tgCTGCA	0.0001013	3.3572671

Table A.11: STE12 Motifs

MEME+ (Conservation)			MEME (Plain)		
Motif	HypG Score	Mismatch Score	Motif	HypG Score	Mismatch Score
GCTGACt	0.0000000	5.5690116	TGCTTaC	0.0000000	4.8165262
ACCcaAA	0.0000012	4.3202396	TaCCTTC	0.0000000	4.3483167
CGGAagt	0.0000069	5.9549575	CGAaTTT	0.0000000	4.6837337
gcGAtGg	0.0000837	5.3790442	aAagaGC	0.0000000	5.7016966
tCccGCc	0.0001521	5.3617703	GGGCtGa	0.0000001	4.6999018
GaACCgG	0.0002597	4.9131201	GCagGgG	0.0000001	5.4915399
GCctgGG	0.0003086	6.1295252	TTCAACC	0.0000002	5.130679
cCCGctc	0.0004963	4.8066117	tGCtTTT	0.0000003	5.5115417
CCGagGa	0.0007387	5.8232162	GCctTGC	0.0000005	5.085115
GGGtgcc	0.0008329	5.8060099	GCgGAAa	0.0000009	4.2507952
TcTGcCC	0.0010139	4.8865377	GCTGaCT	0.0000013	5.7467932
GtTGaAt	0.001701	4.836169	cCTGCGT	0.0000017	5.8055512
GTtTAtA	0.0020359	4.7513683	ccGgTtC	0.0000033	4.6983994
ccCgtGC	0.0041147	5.5777363	GagAgTT	0.0000033	5.6342763
gcgcgcc	0.0042922	5.2527658	tTTCcGc	0.0000113	4.398035
GGcaCGC	0.0047461	6.0850959	TTCGCtT	0.0000341	4.2196881
GCGGAac	0.0074823	5.2923723	gGGaggG	0.0000916	5.2010311
cccgcgc	0.0080749	5.2565116	TTCcgTT	0.000103	4.2865443
cccgcgc	0.0084664	5.256214	GTaTAtG	0.0001108	5.0955053
ggcgcgc	0.0084854	5.2458367	TttACGG	0.0006987	4.9973447

Table A.12: YAP1 Motifs

	-	A	T	G	C
A	0.05252	0.87735	0.02629	0.03212	0.01172
T	0.07136	0.02145	0.85431	0.01289	0.03999
G	0.03684	0.05189	0.01344	0.8894	0.00843
C	0.04125	0.01432	0.04759	0.02541	0.87142
	-	A	T	G	C
A	0.18659	0.74765	0.019	0.03066	0.01609
T	0.2097	0.03143	0.70742	0.02145	0.03001
G	0.16055	0.05858	0.00843	0.74229	0.03016
C	0.1474	0.03175	0.05393	0.01115	0.75577
	-	A	T	G	C
A	0.15016	0.71997	0.03941	0.06855	0.02192
T	0.15836	0.0514	0.67747	0.03143	0.08135
G	0.12711	0.06861	0.04186	0.73226	0.03016
C	0.15532	0.03967	0.06502	0.02858	0.71141

Table A.13: ABF1 Motif Conservation

A.3 Conservation Matrices

We show the full set of conservation matrices used to re-discover each factor's motif in our test set.

	-	A	T	G	C
A	0.0479	0.89879	0.02025	0.02663	0.00642
T	0.0639	0.01679	0.87002	0.00946	0.03982
G	0.03801	0.03341	0.0104	0.91123	0.00695
C	0.04106	0.01113	0.03773	0.01556	0.89452
	-	A	T	G	C
A	0.1766	0.76584	0.01919	0.02876	0.00961
T	0.19162	0.02726	0.72973	0.01784	0.03354
G	0.15881	0.04146	0.015	0.76397	0.02075
C	0.153	0.02332	0.04881	0.00891	0.76595
	-	A	T	G	C
A	0.13193	0.76158	0.03514	0.05429	0.01706
T	0.14347	0.04506	0.71508	0.02412	0.07228
G	0.11394	0.05527	0.03456	0.77432	0.02191
C	0.13305	0.0333	0.05436	0.0211	0.75819

Table A.14: CBF1 Motif Conservation

	-	A	T	G	C
A	0.03775	0.90502	0.02051	0.02805	0.00866
T	0.05166	0.0169	0.87973	0.01058	0.04113
G	0.02807	0.03703	0.01013	0.918	0.00677
C	0.03163	0.01093	0.04143	0.01638	0.89963
	-	A	T	G	C
A	0.17888	0.75958	0.01836	0.03021	0.01297
T	0.19178	0.02954	0.72381	0.01901	0.03586
G	0.15472	0.04376	0.01462	0.76445	0.02246
C	0.15469	0.02618	0.04905	0.00876	0.76132
	-	A	T	G	C
A	0.13902	0.74988	0.03236	0.0593	0.01944
T	0.14964	0.04534	0.69431	0.02743	0.08327
G	0.12221	0.05945	0.03703	0.76221	0.0191
C	0.14054	0.03272	0.05559	0.02183	0.74934

Table A.15: GCN4 Motif Conservation

	-	A	T	G	C
A	0.04662	0.89841	0.01971	0.02799	0.00729
T	0.05885	0.01449	0.87912	0.0083	0.03925
G	0.03352	0.02794	0.00897	0.92506	0.00451
C	0.0379	0.00895	0.03456	0.01229	0.9063
	-	A	T	G	C
A	0.17806	0.76179	0.01867	0.03006	0.01143
T	0.18473	0.02996	0.73673	0.01655	0.03203
G	0.15961	0.03687	0.01455	0.76661	0.02236
C	0.1381	0.02454	0.04569	0.00895	0.78272
	-	A	T	G	C
A	0.13045	0.75868	0.0342	0.05697	0.01971
T	0.14139	0.03822	0.71403	0.02584	0.08052
G	0.11051	0.05137	0.03575	0.78112	0.02125
C	0.12474	0.02565	0.05348	0.01897	0.77715

Table A.16: GCR1 Motif Conservation

	-	A	T	G	C
A	0.04864	0.89644	0.01969	0.02796	0.00728
T	0.06215	0.01633	0.87359	0.00819	0.03975
G	0.03575	0.03575	0.00978	0.91218	0.00654
C	0.04079	0.00944	0.03974	0.01571	0.89432
	-	A	T	G	C
A	0.18201	0.75893	0.01762	0.02899	0.01245
T	0.19043	0.02957	0.72698	0.01837	0.03466
G	0.15477	0.04549	0.01411	0.76287	0.02277
C	0.1557	0.02407	0.04496	0.00735	0.76791
	-	A	T	G	C
A	0.12825	0.76203	0.03416	0.05794	0.01762
T	0.14258	0.0428	0.71171	0.02244	0.08047
G	0.11582	0.05523	0.03575	0.77152	0.02168
C	0.13063	0.03138	0.05332	0.02198	0.76268

Table A.17: HAP3 Motif Conservation

	-	A	T	G	C
A	0.04431	0.90482	0.01516	0.02704	0.00868
T	0.05283	0.01271	0.88792	0.00849	0.03805
G	0.02977	0.03528	0.00885	0.91945	0.00665
C	0.03961	0.00967	0.03319	0.01608	0.90145

	-	A	T	G	C
A	0.18251	0.75798	0.01732	0.03027	0.01192
T	0.19324	0.02855	0.72956	0.01588	0.03277
G	0.1564	0.04409	0.01326	0.76529	0.02096
C	0.15509	0.02464	0.04602	0.00646	0.76779

	-	A	T	G	C
A	0.11881	0.77094	0.03135	0.06051	0.0184
T	0.13095	0.0391	0.719	0.02644	0.0845
G	0.10905	0.0584	0.03308	0.77741	0.02207
C	0.12729	0.03105	0.05351	0.02036	0.76779

Table A.18: HAP4 Motif Conservation

	-	A	T	G	C
A	0.05213	0.90131	0.01476	0.02269	0.0091
T	0.06575	0.01318	0.8717	0.01099	0.03837
G	0.04165	0.03661	0.00635	0.90903	0.00635
C	0.04538	0.0074	0.03435	0.01598	0.89689

	-	A	T	G	C
A	0.1846	0.76771	0.01363	0.02269	0.01137
T	0.19387	0.02413	0.73592	0.01537	0.03071
G	0.16142	0.03661	0.01266	0.76657	0.02274
C	0.15687	0.0221	0.03558	0.0074	0.77804

	-	A	T	G	C
A	0.14158	0.75978	0.03288	0.04873	0.01703
T	0.15664	0.03728	0.7173	0.02085	0.06794
G	0.1299	0.05048	0.03409	0.76909	0.01644
C	0.15075	0.03068	0.04293	0.01843	0.75722

Table A.19: MCM1 Motif Conservation

	-	A	T	G	C
A	0.05344	0.88962	0.0205	0.02731	0.00913
T	0.06842	0.01686	0.86196	0.01125	0.04152
G	0.04208	0.03953	0.01024	0.90046	0.00769
C	0.04916	0.01264	0.0479	0.01768	0.87263
	-	A	T	G	C
A	0.19091	0.74419	0.01936	0.03186	0.01368
T	0.21076	0.03031	0.70504	0.01686	0.03703
G	0.17071	0.05227	0.01661	0.7349	0.02552
C	0.17759	0.02649	0.05671	0.00886	0.73035
	-	A	T	G	C
A	0.13411	0.75442	0.03867	0.05231	0.0205
T	0.14351	0.04376	0.7028	0.02694	0.08299
G	0.11594	0.065	0.03571	0.76037	0.02298
C	0.12344	0.03531	0.06553	0.02649	0.74923

Table A.20: REB1 Motif Conservation

	-	A	T	G	C
A	0.04647	0.90584	0.0159	0.02495	0.00684
T	0.049	0.01419	0.88777	0.00875	0.0403
G	0.03222	0.03333	0.00892	0.92104	0.00448
C	0.03775	0.00974	0.03991	0.0162	0.8964
	-	A	T	G	C
A	0.16196	0.78243	0.01816	0.02722	0.01024
T	0.17628	0.02615	0.74526	0.01963	0.03268
G	0.15095	0.04554	0.01447	0.76458	0.02446
C	0.14549	0.02267	0.04852	0.00866	0.77466
	-	A	T	G	C
A	0.14497	0.74393	0.03288	0.05779	0.02043
T	0.14147	0.0403	0.71806	0.02507	0.07511
G	0.11988	0.05885	0.02889	0.77124	0.02113
C	0.13579	0.03129	0.05283	0.02051	0.75958

Table A.21: STE12 Motif Conservation

	-	A	T	G	C
A	0.04337	0.90342	0.01954	0.02712	0.00654
T	0.05829	0.01593	0.88015	0.01063	0.03499
G	0.03444	0.03552	0.00972	0.91382	0.00649
C	0.03962	0.01046	0.03858	0.0167	0.89465

	-	A	T	G	C
A	0.1636	0.77452	0.01954	0.03037	0.01196
T	0.17691	0.02864	0.74353	0.01699	0.03393
G	0.15377	0.04519	0.01294	0.76439	0.02369
C	0.15209	0.024	0.04691	0.00733	0.76967

	-	A	T	G	C
A	0.12894	0.76261	0.03146	0.05745	0.01954
T	0.13878	0.04135	0.71599	0.02652	0.07736
G	0.11615	0.05379	0.03552	0.77622	0.01832
C	0.13439	0.03129	0.05316	0.02087	0.7603

Table A.22: YAP1 Motif Conservation

	-	A	T	G	C
A	0.09314	0.74758	0.04915	0.07907	0.03106
T	0.09135	0.04977	0.74434	0.03163	0.08291
G	0.09507	0.13873	0.04942	0.67805	0.03873
C	0.0906	0.05451	0.13906	0.04103	0.67479
	-	A	T	G	C
A	0.25279	0.55532	0.0659	0.08886	0.03712
T	0.25058	0.06887	0.55231	0.03977	0.08846
G	0.24988	0.16746	0.07027	0.46537	0.04702
C	0.25394	0.07378	0.16491	0.04812	0.45926
	-	A	T	G	C
A	0.22899	0.51293	0.08777	0.10895	0.06135
T	0.22493	0.09092	0.5106	0.06323	0.11031
G	0.23333	0.16787	0.08841	0.4389	0.07149
C	0.22497	0.0904	0.1696	0.07248	0.44254

Table A.23: ABF1 Background Conservation

	-	A	T	G	C
A	0.09014	0.75487	0.04722	0.07808	0.02969
T	0.08772	0.04755	0.7532	0.02995	0.08157
G	0.08941	0.13992	0.04737	0.68561	0.03768
C	0.08745	0.05267	0.14013	0.03878	0.68097
	-	A	T	G	C
A	0.24759	0.55988	0.06593	0.08938	0.03723
T	0.24498	0.06729	0.55945	0.03878	0.0895
G	0.2469	0.16868	0.06953	0.46839	0.0465
C	0.25091	0.07271	0.16747	0.04686	0.46206
	-	A	T	G	C
A	0.2082	0.52958	0.08847	0.11191	0.06184
T	0.20349	0.09091	0.52923	0.06235	0.11402
G	0.21199	0.17344	0.08854	0.4539	0.07214
C	0.20634	0.08931	0.17702	0.07077	0.45656

Table A.24: CBF1 Background Conservation

	-	A	T	G	C
A	0.07292	0.76869	0.04795	0.08033	0.03011
T	0.07098	0.04833	0.76656	0.03098	0.08315
G	0.07086	0.14306	0.04814	0.6994	0.03854
C	0.07049	0.05276	0.14301	0.03996	0.69378
	-	A	T	G	C
A	0.25161	0.55558	0.06598	0.08958	0.03725
T	0.24951	0.06741	0.55432	0.03901	0.08976
G	0.24936	0.16876	0.0699	0.46536	0.04662
C	0.25358	0.07289	0.16801	0.04732	0.4582
	-	A	T	G	C
A	0.22015	0.5209	0.08733	0.11034	0.06128
T	0.21491	0.08946	0.52044	0.0621	0.11309
G	0.22365	0.17093	0.08737	0.4464	0.07166
C	0.2167	0.08882	0.1746	0.06982	0.45006

Table A.25: GCN4 Background Conservation

	-	A	T	G	C
A	0.08364	0.76209	0.04621	0.07877	0.02929
T	0.0817	0.04624	0.76021	0.02986	0.08199
G	0.08305	0.13991	0.04626	0.69345	0.03733
C	0.08126	0.0513	0.13944	0.03858	0.68941
	-	A	T	G	C
A	0.24085	0.56555	0.06574	0.09094	0.03691
T	0.23767	0.06748	0.56541	0.03911	0.09033
G	0.23914	0.17073	0.06947	0.47449	0.04616
C	0.24451	0.07263	0.1687	0.04769	0.46647
	-	A	T	G	C
A	0.20773	0.52904	0.08796	0.11333	0.06193
T	0.20323	0.09007	0.52867	0.06275	0.11529
G	0.211	0.1733	0.08803	0.45548	0.07219
C	0.20508	0.08929	0.17689	0.07094	0.45781

Table A.26: GCR1 Background Conservation

	-	A	T	G	C
A	0.08458	0.75981	0.04708	0.07886	0.02967
T	0.08318	0.04735	0.75767	0.03016	0.08165
G	0.08448	0.13939	0.04694	0.69149	0.0377
C	0.0826	0.05218	0.13949	0.0391	0.68662
	-	A	T	G	C
A	0.25171	0.55705	0.06508	0.08947	0.03669
T	0.24927	0.06683	0.55689	0.03861	0.0884
G	0.24986	0.1674	0.06838	0.46859	0.04578
C	0.25463	0.07201	0.16557	0.04671	0.46108
	-	A	T	G	C
A	0.2111	0.52708	0.08759	0.11237	0.06187
T	0.20638	0.09023	0.52652	0.06265	0.11422
G	0.21378	0.17162	0.08726	0.45543	0.0719
C	0.20804	0.08937	0.17461	0.07102	0.45697

Table A.27: HAP3 Background Conservation

	-	A	T	G	C
A	0.08801	0.76054	0.04506	0.07785	0.02854
T	0.08545	0.04601	0.7581	0.02902	0.08142
G	0.08635	0.13828	0.04498	0.6937	0.03669
C	0.08467	0.05016	0.13743	0.03853	0.68921
	-	A	T	G	C
A	0.24913	0.56094	0.06388	0.09002	0.03603
T	0.2465	0.06591	0.56042	0.038	0.08915
G	0.24915	0.16788	0.06733	0.47036	0.04528
C	0.25464	0.071	0.16543	0.046	0.46293
	-	A	T	G	C
A	0.20343	0.53437	0.08674	0.11373	0.06173
T	0.19967	0.08982	0.53147	0.06282	0.11623
G	0.20712	0.17371	0.08677	0.45932	0.07308
C	0.19984	0.08952	0.17643	0.07144	0.46277

Table A.28: HAP4 Background Conservation

	-	A	T	G	C
A	0.09632	0.74662	0.048	0.07887	0.03018
T	0.09345	0.04831	0.74587	0.03076	0.08161
G	0.09497	0.14055	0.04824	0.67807	0.03818
C	0.0927	0.05348	0.14032	0.03949	0.67401
	-	A	T	G	C
A	0.25326	0.55437	0.06553	0.08978	0.03706
T	0.25049	0.06707	0.55454	0.03898	0.08892
G	0.25089	0.16849	0.06875	0.46541	0.04645
C	0.25623	0.07288	0.16741	0.04745	0.45603
	-	A	T	G	C
A	0.22592	0.51538	0.08693	0.11028	0.06148
T	0.22015	0.08947	0.51579	0.06194	0.11265
G	0.22903	0.16896	0.08686	0.44394	0.07122
C	0.22326	0.08868	0.17338	0.07019	0.44448

Table A.29: MCM1 Background Conservation

	-	A	T	G	C
A	0.09492	0.75119	0.04669	0.07743	0.02977
T	0.09285	0.04735	0.74972	0.03003	0.08005
G	0.09477	0.13903	0.04652	0.68202	0.03766
C	0.09166	0.05196	0.1389	0.03879	0.67868
	-	A	T	G	C
A	0.2723	0.54227	0.06259	0.08714	0.03571
T	0.27152	0.06496	0.5408	0.03688	0.08585
G	0.27086	0.16367	0.0658	0.45424	0.04542
C	0.27746	0.06922	0.16086	0.04482	0.44764
	-	A	T	G	C
A	0.21233	0.52753	0.08806	0.11125	0.06083
T	0.20753	0.09011	0.52854	0.06169	0.11213
G	0.21628	0.17168	0.08742	0.45234	0.07228
C	0.20959	0.08928	0.1755	0.07084	0.45478

Table A.30: REB1 Background Conservation

	-	A	T	G	C
A	0.07966	0.7684	0.04495	0.0785	0.02849
T	0.07861	0.04515	0.76582	0.02905	0.08137
G	0.07834	0.13915	0.04451	0.70214	0.03587
C	0.0785	0.05	0.13948	0.03831	0.69371

	-	A	T	G	C
A	0.2308	0.57404	0.06584	0.09233	0.03699
T	0.22883	0.06713	0.57329	0.03939	0.09137
G	0.23161	0.17103	0.06891	0.48228	0.04616
C	0.23549	0.07274	0.1703	0.04817	0.4733

	-	A	T	G	C
A	0.19729	0.53619	0.08868	0.11481	0.06303
T	0.1934	0.09101	0.53595	0.06317	0.11646
G	0.20006	0.17527	0.0881	0.46486	0.0717
C	0.19633	0.09046	0.17895	0.07228	0.46198

Table A.31: STE12 Background Conservation

	-	A	T	G	C
A	0.08559	0.76228	0.04533	0.07848	0.02832
T	0.08412	0.04581	0.76014	0.02873	0.08121
G	0.08408	0.13897	0.04561	0.69488	0.03645
C	0.08237	0.05033	0.13762	0.0383	0.69139

	-	A	T	G	C
A	0.21369	0.58592	0.06815	0.09402	0.03822
T	0.21534	0.06903	0.58291	0.03994	0.09277
G	0.21233	0.17542	0.07178	0.49241	0.04806
C	0.21916	0.07469	0.17381	0.0489	0.48344

	-	A	T	G	C
A	0.19703	0.53651	0.08931	0.11449	0.06266
T	0.19313	0.09133	0.53547	0.06294	0.11713
G	0.19877	0.17526	0.08975	0.46382	0.07239
C	0.19329	0.09008	0.1795	0.07186	0.46528

Table A.32: YAP1 Background Conservation

A.4 Discovery Times

One of the points we have argued is that, even when both methods discover the correct motif *eventually*, adding conservation often improves how quickly the correct motif is found. This notion of "discovery time" is a reasonable metric to use; in most probabilistic search motif discovery methods, the algorithm cannot be run to exhaustively enumerate all "interesting" motifs. Instead, the search is stopped after a period of time and the standing results are ranked by an independent metric (here, the hypergeometric p-value). Therefore, a method which returns the correct result sooner would raise confidence that the correct motif has been covered in a fixed set of results.

To indicate this metric, we show two time series for each factor, for discovery with and without conservation. The series' values are the mismatch score for motifs *in the order they were discovered.* The minimum point of each series is the closest that the algorithm came to discovering the "correct" motif.

(In these graphs, the red lines are the mismatch scores of MEME+, and the blue lines are the mismatch scores of MEME Plain; the X axis indicates the order of discovery, and the Y axis indicates mismatch scores for each of those motifs).

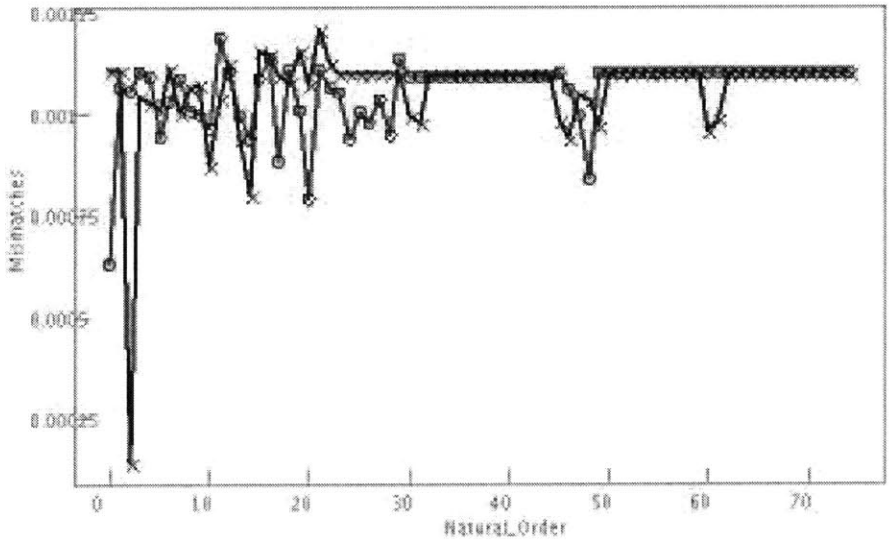

Figure A-1: ABF1 Mismatch Discovery Time

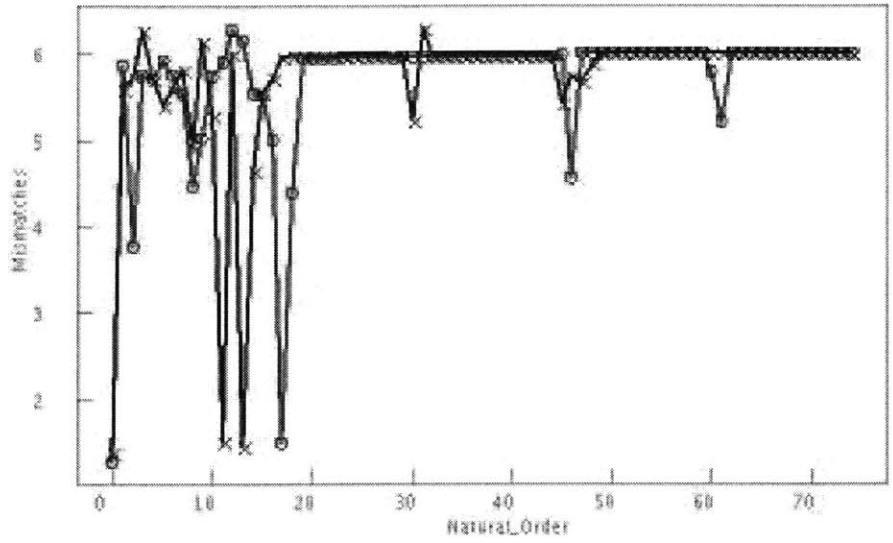

Figure A-2: CBF1 Mismatch Discovery Time

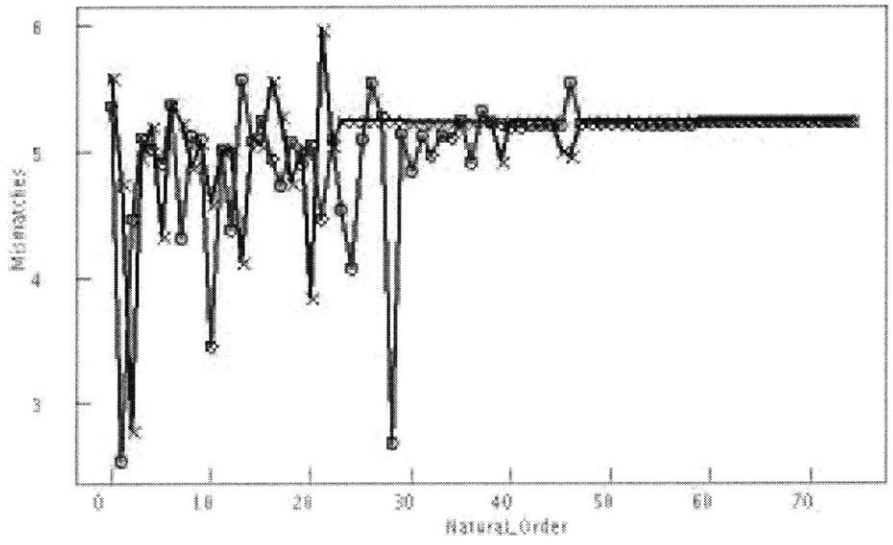

Figure A-3: GCN4 Mismatch Discovery Time

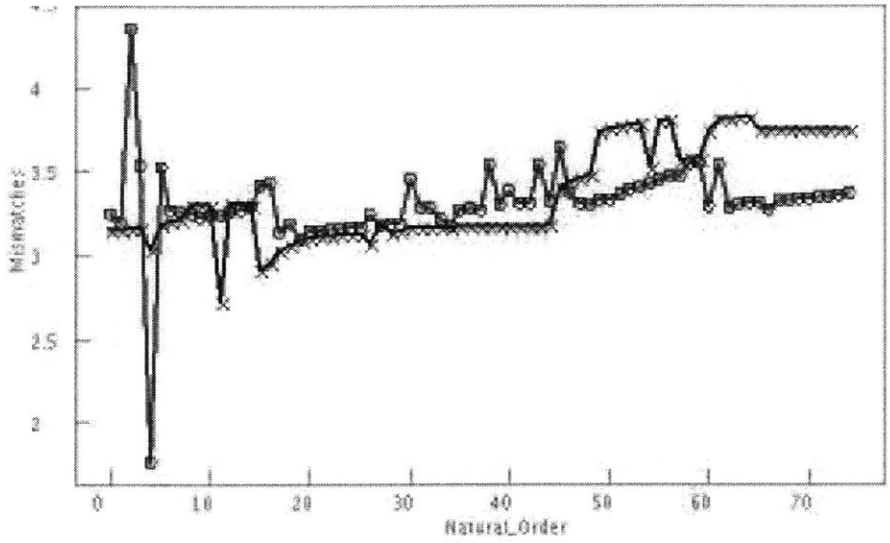

Figure A-4: GCR1 Mismatch Discovery Time

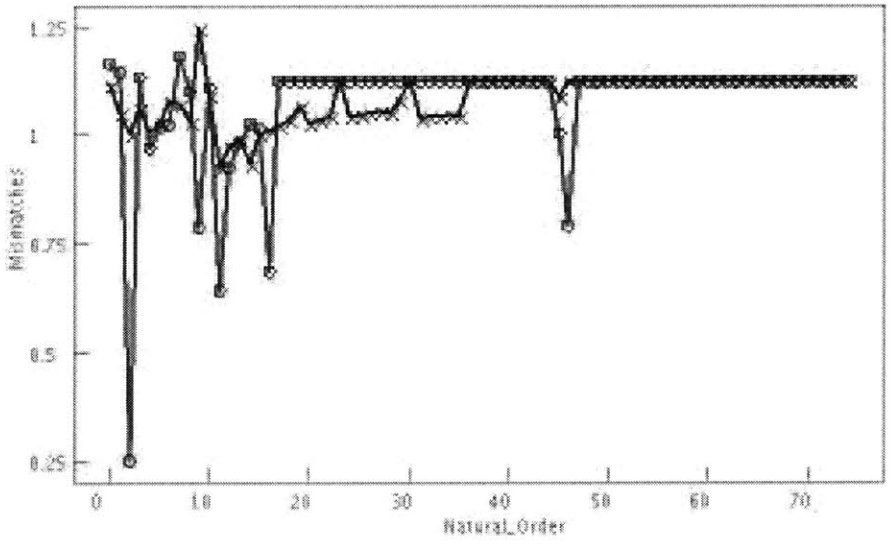

Figure A-5: HAP3 Mismatch Discovery Time

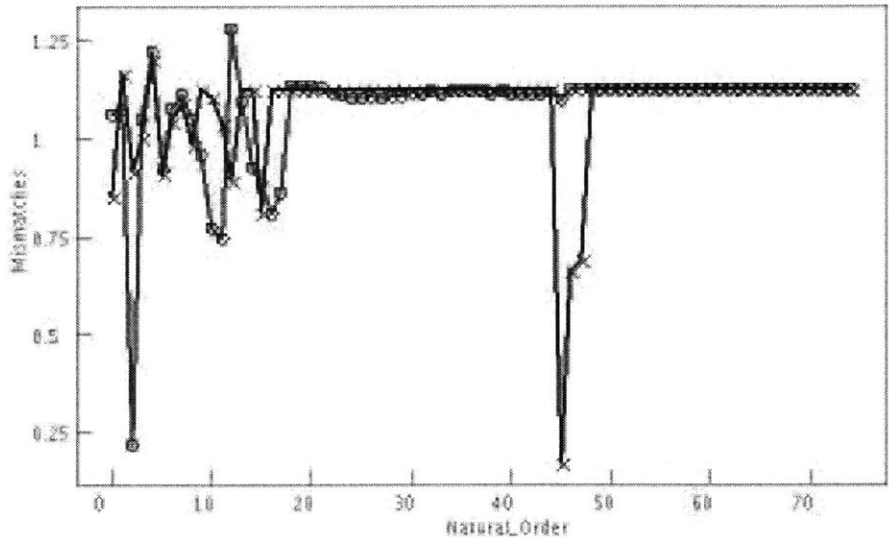

Figure A-6: HAP4 Mismatch Discovery Time

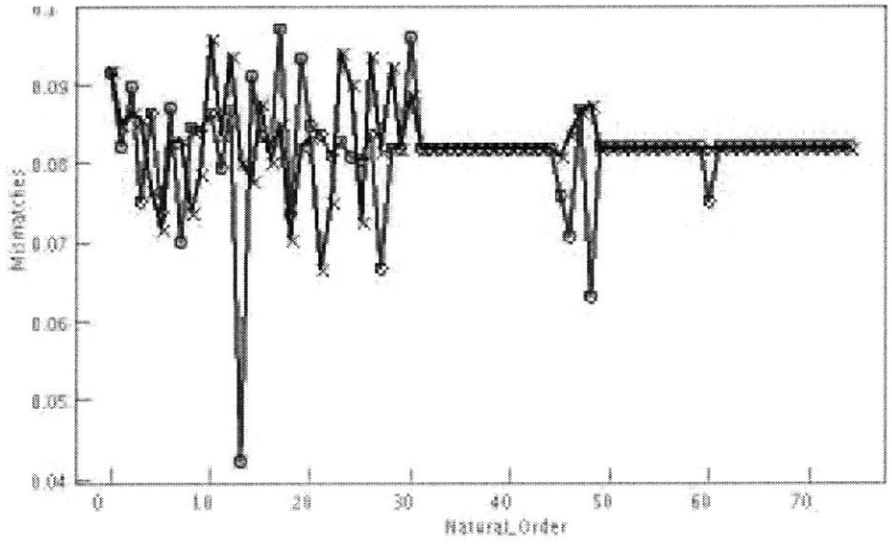

Figure A-7: MCM1 Mismatch Discovery Time

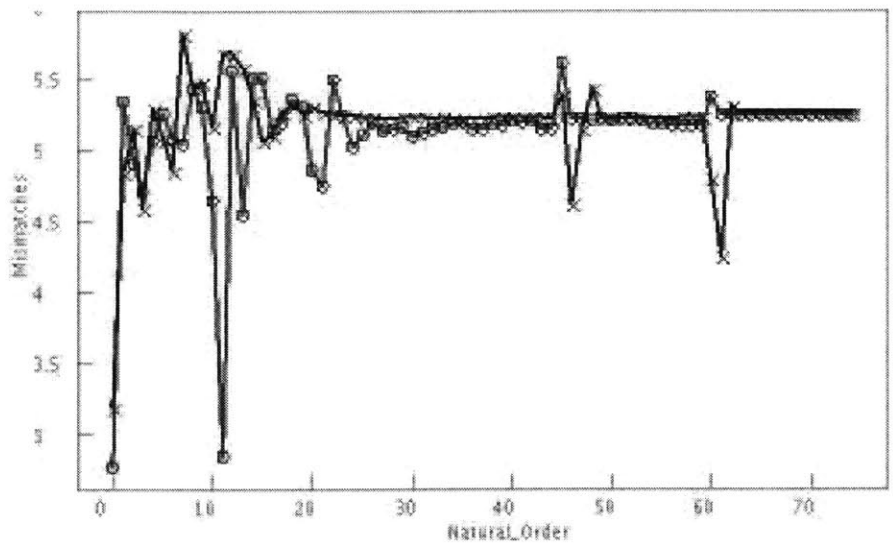

Figure A-8: REB1 Mismatch Discovery Time

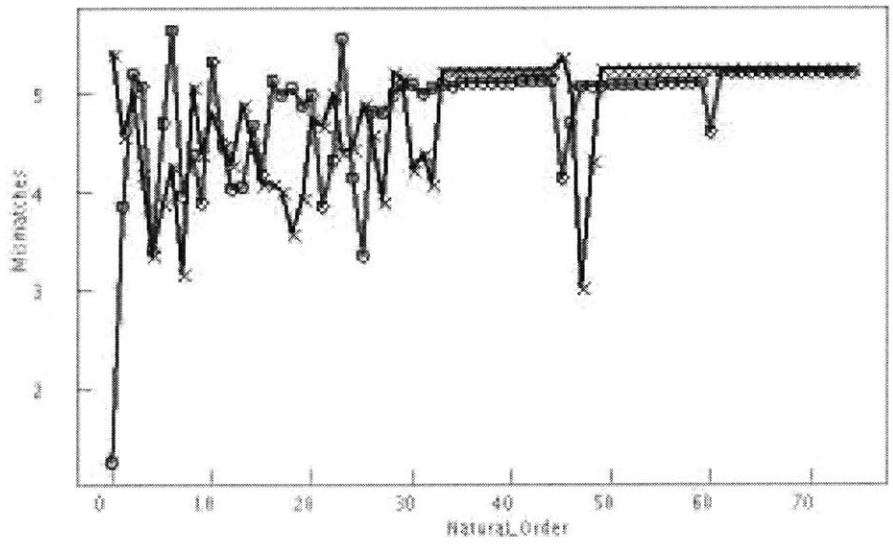

Figure A-9: STE12 Mismatch Discovery Time

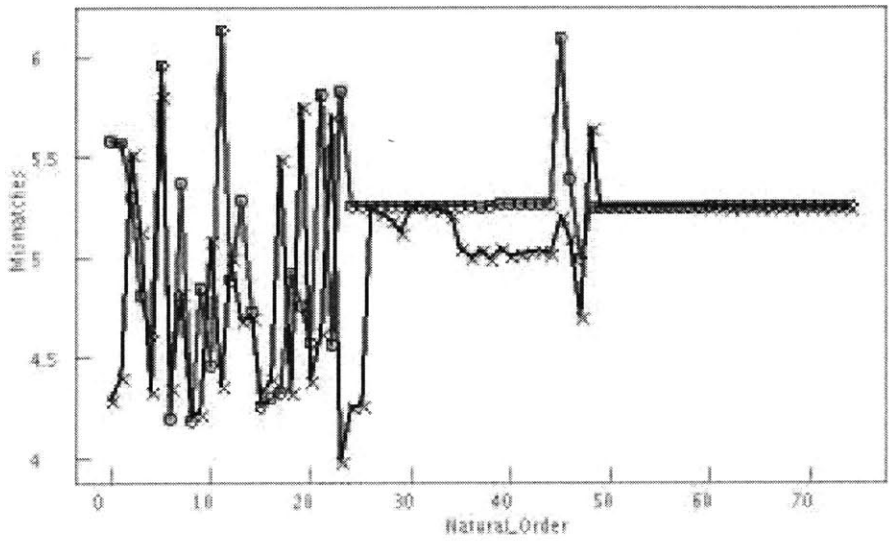

Figure A-10: YAP1 Mismatch Discovery Time

A.5 Discovered Motif Entropies

For each factor we plot the entropy of each discovered motif against its mismatch score (circles are MEME+, crosses are MEME Plain). We note that several graphs appear to have tightly bunched clousters of motifs (often from the Plain MEME algorithm); these are stretches of similar, uniformative motifs discovered after the initial motifs reachable from the seed appear to be exhausted. We believe this to be an effect of sequential motif masking during repeated motif discovery.

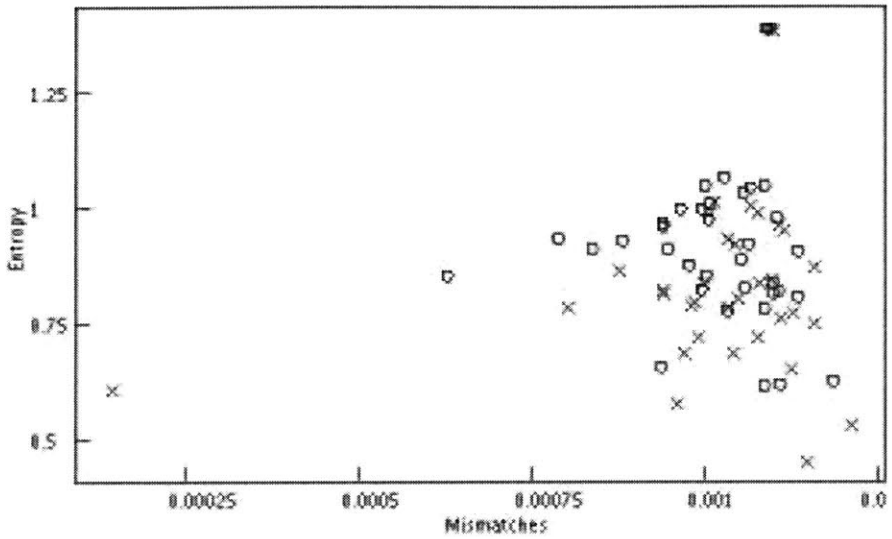

Figure A-11: ABF1 Entropies vs. Mismatch Scores

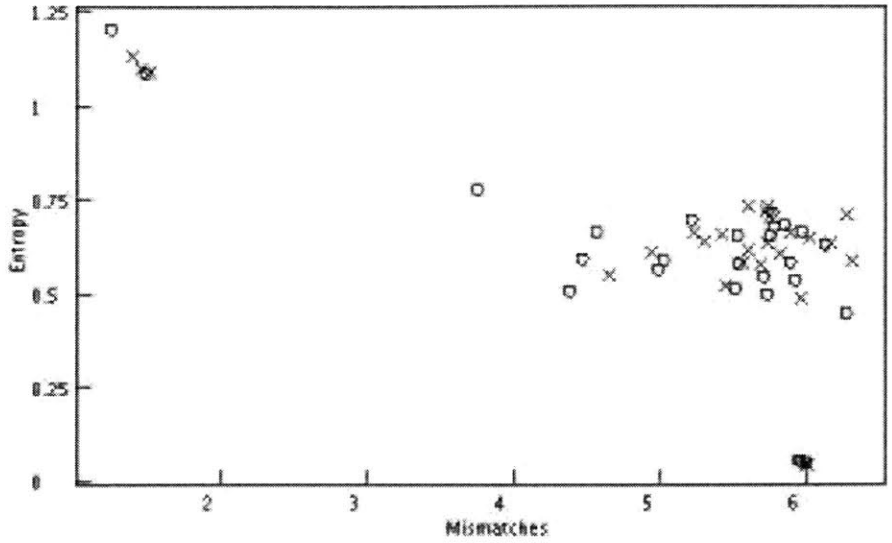

Figure A-12: CBF1 Entropies vs. Mismatch Scores

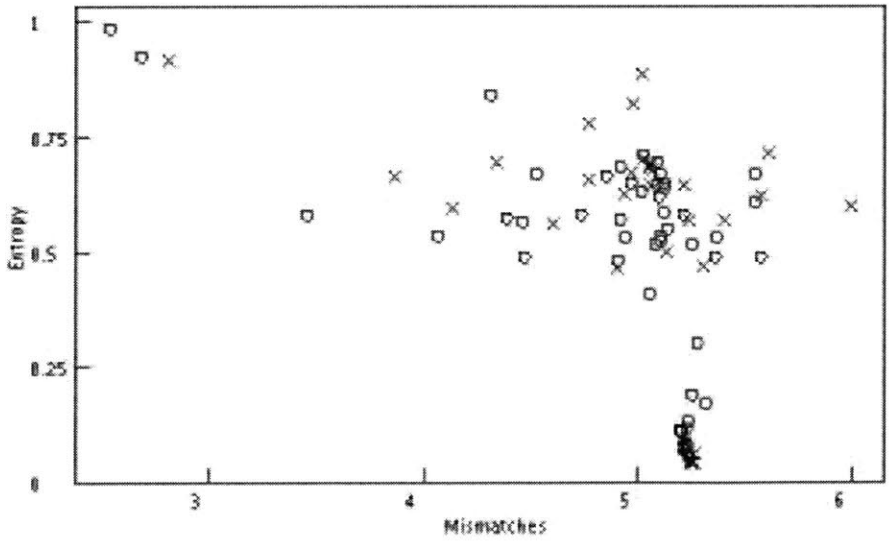

Figure A-13: GCN4 Entropies vs. Mismatch Scores

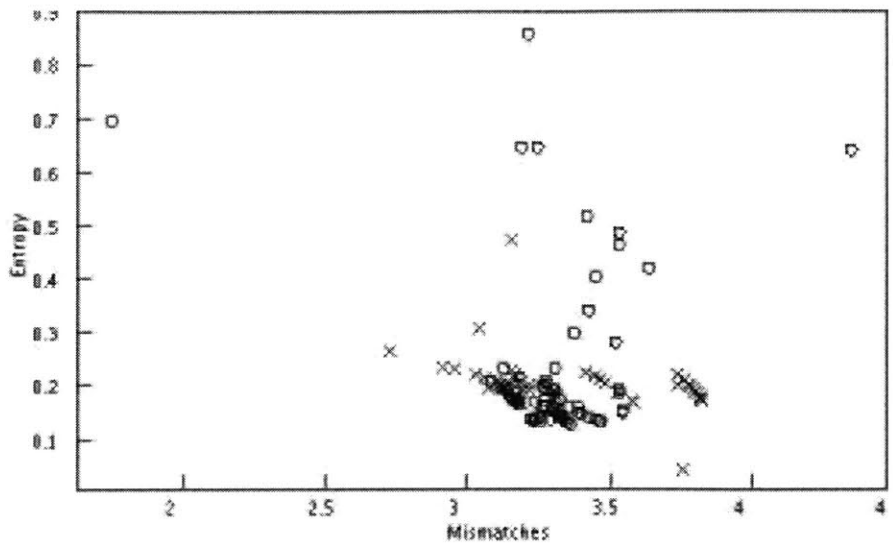

Figure A-14: GCR1 Entropies vs. Mismatch Scores

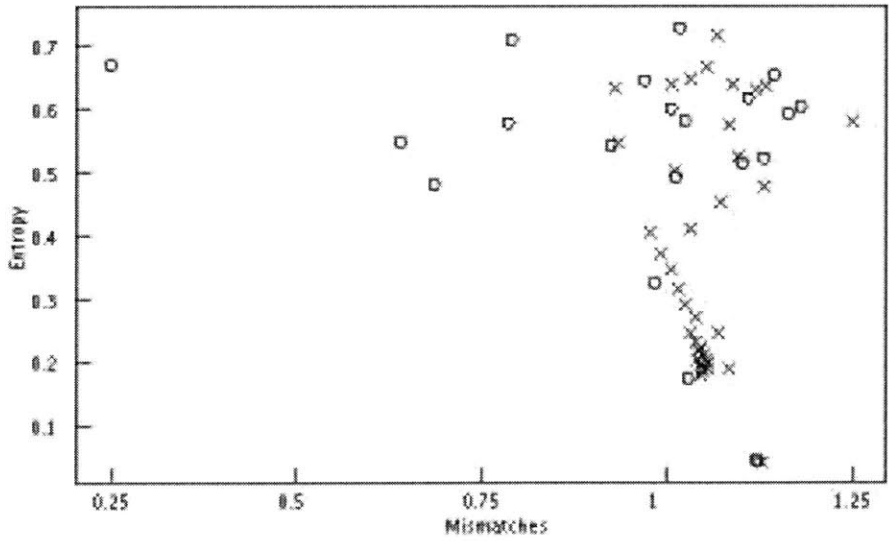

Figure A-15: HAP3 Entropies vs. Mismatch Scores

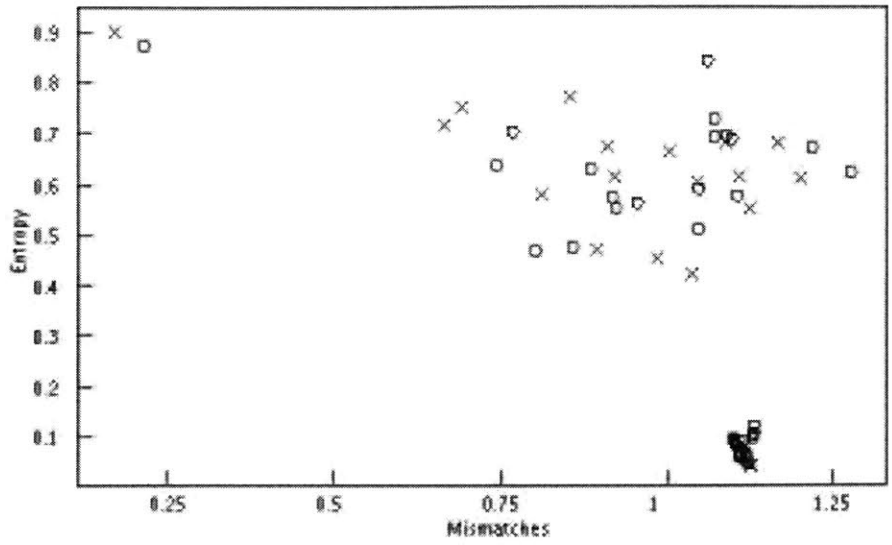

Figure A-16: HAP4 Entropies vs. Mismatch Scores

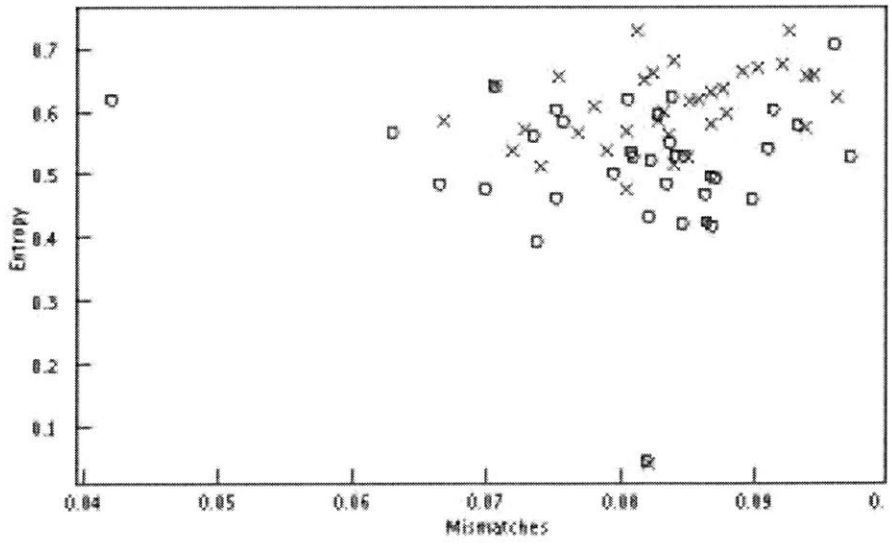

Figure A-17: MCM1 Entropies vs. Mismatch Scores

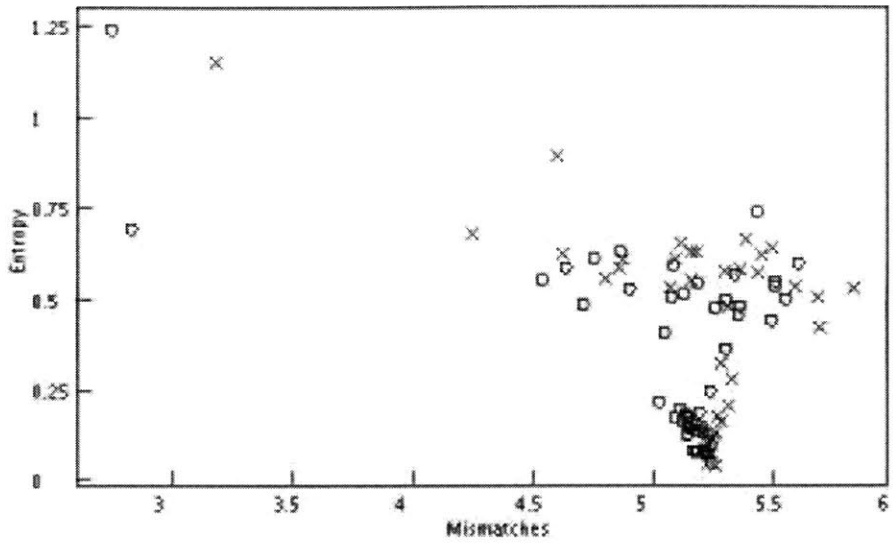

Figure A-18: REB1 Entropies vs. Mismatch Scores

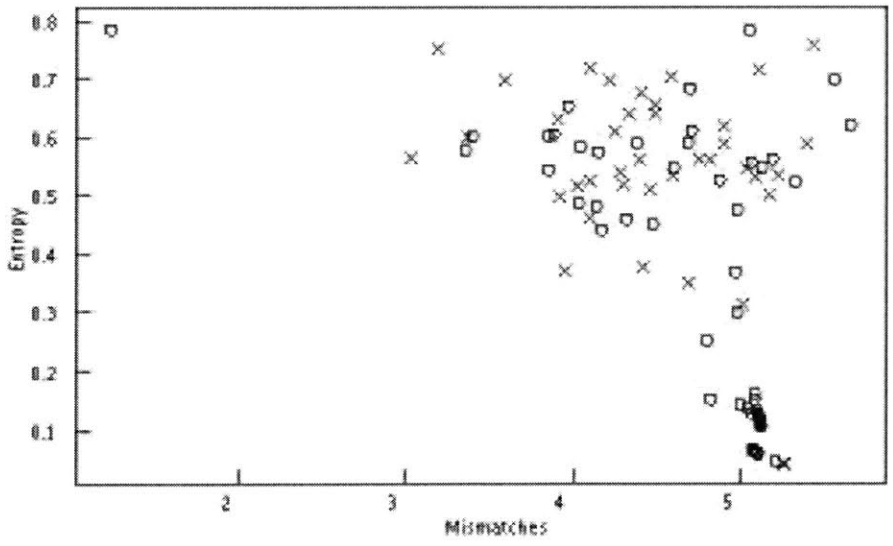

Figure A-19: STE12 Entropies vs. Mismatch Scores

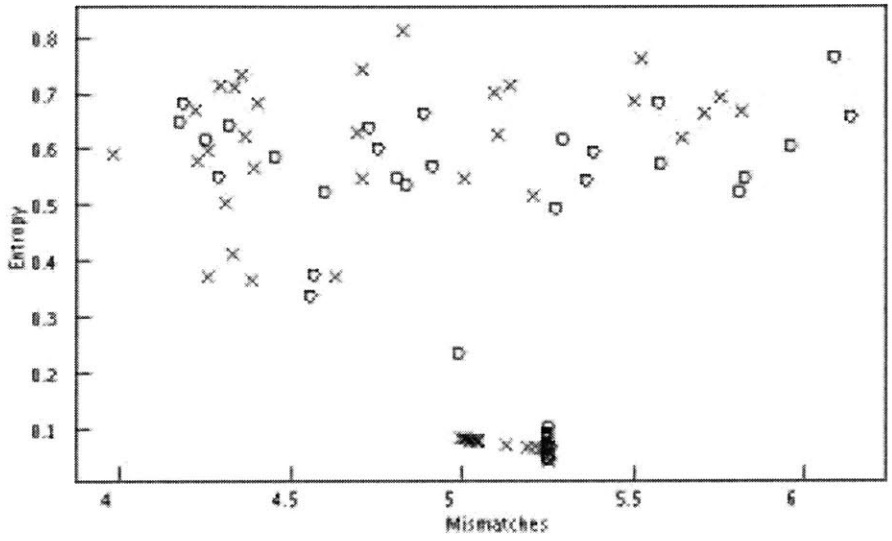

Figure A-20: YAP1 Entropies vs. Mismatch Scores

A.6 Known Motif ROC Curves

Before presenting the results of our two motif discovery algorithms, we some of the equivalent statistics for the known motifs. Some of these statistics have been quoted already in Table 2.1; the given bound and total counts have hypergeometric and binomial scores lower than any of the discovered motifs (what would be displayed as 0.0000 in the tables which follow). However, we may also calculate ROC curves for the known motifs, which are shown in Table A-21.

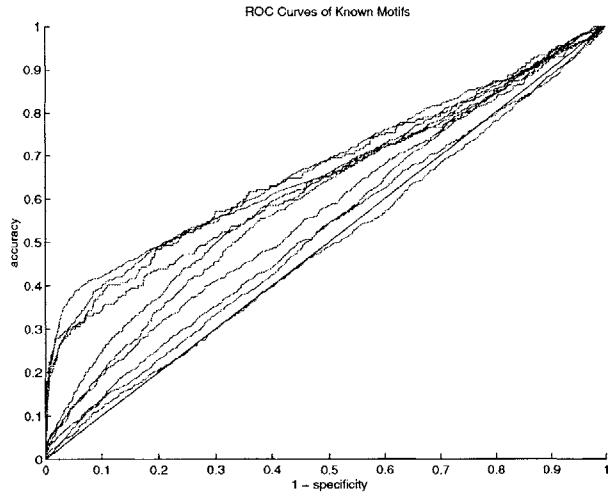

Figure A-21: ROC Curves of Known Motifs

A.7 Discovered Motif ROC Curves

Our final set of exhaustive statistics is the set of ROC curves for the discovered motifs. As before, red curves are MEME+ motifs, and blue curves are derived from Plain MEME motifs. Each graph also has the diagonal marked in a straight green line. It is interesting to note that adding conservation does not appear to dramatically improve the ability of a discovered motif to "generalize." However, in those situations where informative motifs may be found (most notably ABF1), both techniques appear to discover such a motif.

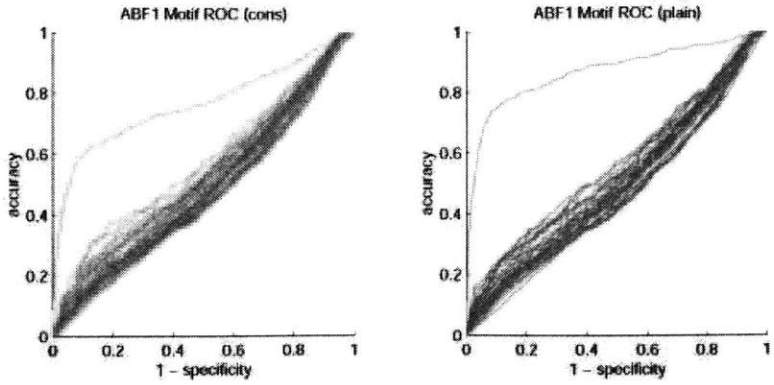

Figure A-22: ABF1 Discovered Motif ROC Curves

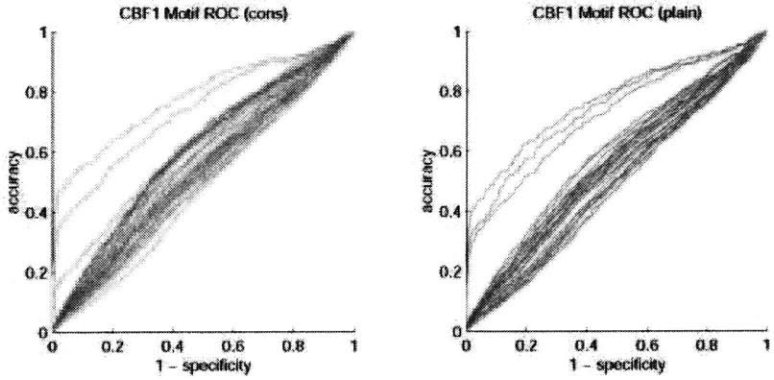

Figure A-23: CBF1 Discovered Motif ROC Curves

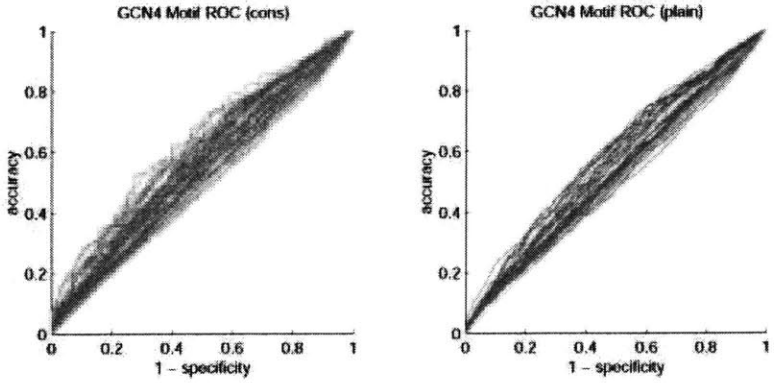

Figure A-24: GCN4 Discovered Motif ROC Curves

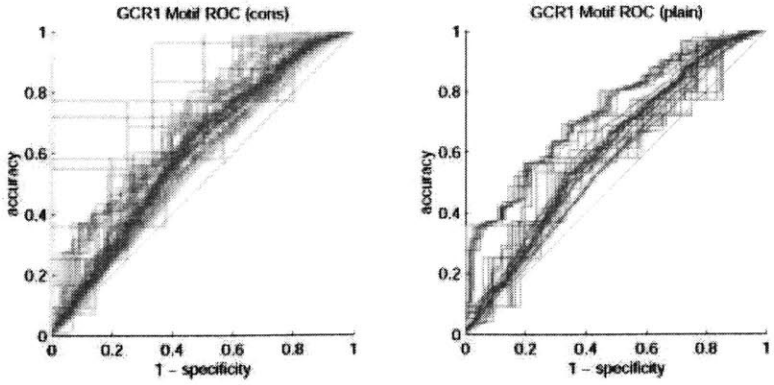

Figure A-25: GCR1 Discovered Motif ROC Curves

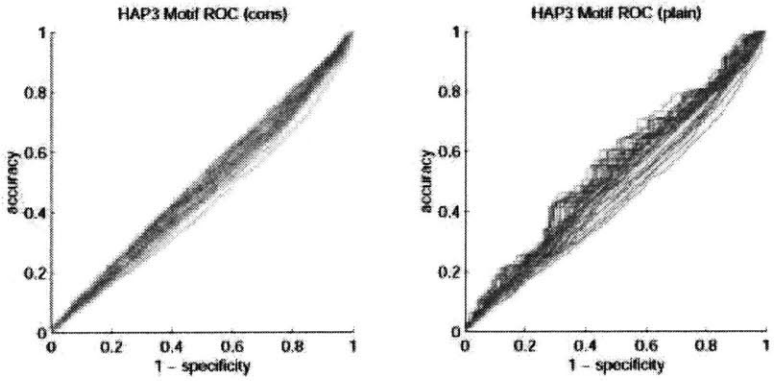

Figure A-26: HAP3 Discovered Motif ROC Curves

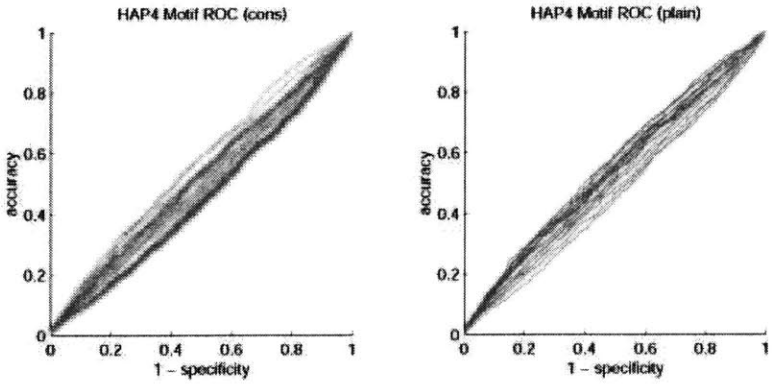

Figure A-27: HAP4 Discovered Motif ROC Curves

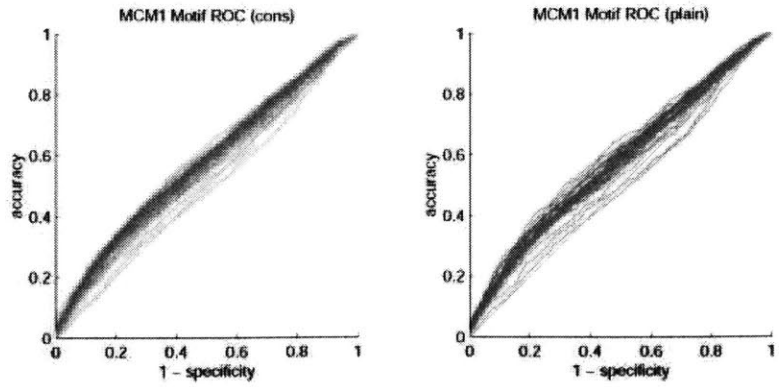

Figure A-28: MCM1 Discovered Motif ROC Curves

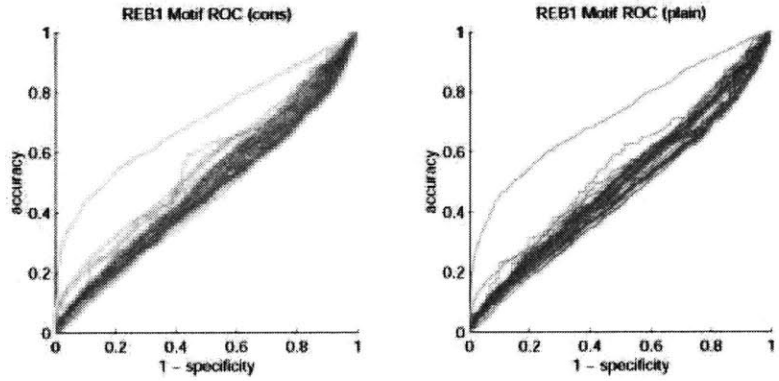

Figure A-29: REB1 Discovered Motif ROC Curves

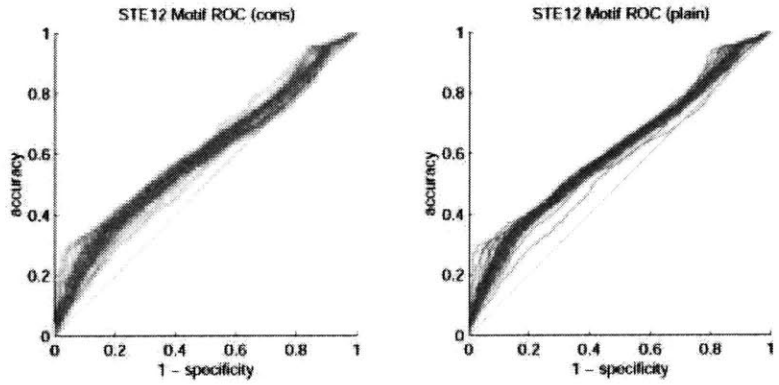

Figure A-30: STE12 Discovered Motif ROC Curves

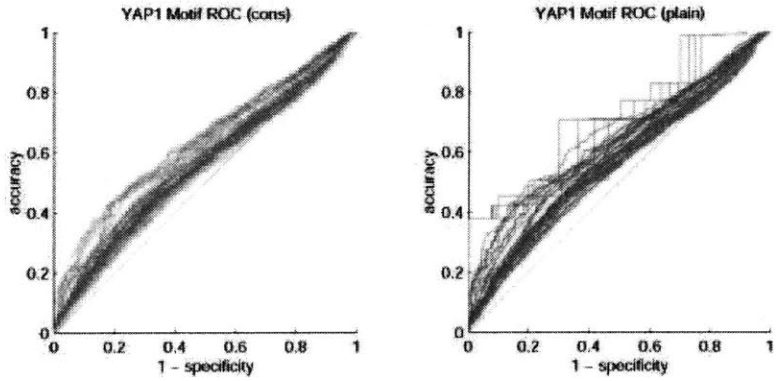

Figure A-31: YAP1 Discovered Motif ROC Curves

Bibliography

[1] Richard Durbin, Sean R. Eddy, Anders Krogh, and Graeme Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids.* Cambridge University Press, 1999.

[2] Andrew F. Neuwald and Philip Green. Detecting patterns in proteins sequences. *Journal of Molecular Biology*, 239:698–712, 1994.

[3] Charles E. Lawrence and A. A. Reilly. An expectation maximization (EM) algorithm for the identification and characterization of common sites in aligned biopolymer sequences. *PROTEINS: Structure, Function, and Genetics*, 7:41–51, 1990.

[4] Timothy Bailey and Charles Elkan. Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Machine Learning Journal*, 21:51–83, 1995.

[5] Charles E. Lawrence, Stephen F. Altschul, Mark S. Boguski, Jun S. Liu, Andrew F. Neuwald, and John C. Wootton. Detecting subtle sequence signals: A Gibbs sampling strategy for multiple alignment. *Science*, 262:208–214, 1993.

[6] Eric P. Xing, Michael I. Jordan, Richard M. Karp, and Stuart Russell. A hierarchical Bayesian Markovian model for motifs in biopolymer sequences.

[7] Yoseph Barash, Gill Bejerano, and Nir Friedman. A simple hyper-geometric approach for discovering putative transcription factor binding sites.

[8] X. Liu, D. L. Brutlag, and J. S. Liu. Bioprospector: Discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. In *Pacific Symposium on Biocomputing*, 2001.

[9] Panayiotis V. Benos, Martha L. Bulyk, and Gary D. Stormo. Additivity in protein-DNA interactions: how good an approximation is it? *Nucleic Acids Research*, 30(20):4442–4451, 2002.

[10] Panayiotis V. Benos, Alan S. Lapedes, and Gary D. Stormo. Probabilistic code for DNA recognition by proteins of the EGR family. *Journal of Molecular Biology*, 323:701–727, 2002.

[11] Gene Yeo and Christopher B. Burge. Maximum entropy modeling of short sequence motifs with applications to rna splicing signals. In *Proceedings of the seventh annual international conference on Computational molecular biology*, pages 322–331. ACM Press, 2003.

[12] Saurabh Sinha and Martin Tompa. A statistical method for finding transcription factor binding sites. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, 2000.

[13] Saurabh Sinha. Discriminative motifs. In *Proceedings of the Sixth Annual International Conference on Computational Molecular Biology*, pages 291–298, 2002.

[14] Pavel Pevzner and Sing-Hoi Sze. Combinatorial approaches to finding subtle signals in DNA sequences. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, pages 269–278, 2000.

[15] Emily Rocke and Martin Tompa. An algorithm for finding novel gapped motifs in DNA sequences, 1999.

[16] Jeremy Buhler and Martin Tompa. Finding motifs using random projections. In *Proceedings of the Fifth Annual International Conference on Computational Molecular Biology*. ACM, 2001.

[17] Laurent Marsan and Marie-France Sagot. Extracting structured motifs using a suffix tree: Algorithms and application to promoter consensus identification. In *Proceedings of the Fourth Annual International Conference on Computational Molecular Biology*, pages 210–219. ACM, 2000.

[18] Timothy L. Bailey and Charles Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. In *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pages 28–36. AAAI Press, 1994.

[19] Gert Thijs, Kathleen Marchal, Magali Lescot, Stephane Rombauts, Bart De Moor, Pierre Rouze, and Yves Moreau. A Gibbs sampling method to detect overrepresented motifs in the upstream regions of coexpressed genes. *Journal of Computational Biology*, 9(2):447–464, 2002.

[20] Frederick P. Roth, Jason D. Hughes, Preston W. Estep, and George M. Church. Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation. *Nature Biotechnology*, 16:939–945, 1998.

[21] Saurabh Sinha and Martin Tompa. YMF: a program for discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic Acides Research*, 31(13):3586–3588, 2003.

[22] Joseph L. DeRisi, Viswanath R. Iyer, and Patrick O. Brown. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, 278:680–686, 1997.

[23] E. Segal, R. Yelensky, and D. Koller. Genome-wide discovery of transcriptional modules from DNA sequence and gene expression. *Bioinformatics*, 2003.

[24] X. Shirley Liu, Douglas L. Brutlag, and Jun S. Liu. An algorithm for finding protein-DNA binding sites with applications to chromatin-immunoprecipitation microarray experiments. *Nature Biotechnology*, 20:835–839, 2002.

[25] Ken Takusagawa. Negative information for motif discovery. In *Proceedings of the 2004 Pacific Symposium on Biocomputing*, pages 360–371. World Scientific, 2004.

[26] Jun S. Liu, Andrew F. Neuwald, and Charles E. Lawrence. Bayesian models for multiple local sequence alignment and Gibbs sampling strategies. *Journal of the American Statistical Association*, 90(432):1156–1170, 1995.

[27] Kathleen Marchal, Gert Thijs, Sigrid De Keersmaecker, Pieter Monsieurs, Bart De Moor, and Jos Vanderleyden. Genome-specific higher-order background models to improve motif detection. *Trends in Microbiology*, 11(2):61–66, 2003.

[28] Gert Thijs, Magali Lescot, Kathleen Marchal, Stephane Rombauts, Bart De Moor, Pierre Rouze, and Yves Moreau. A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling. *Bioinformatics*, 21(12):1113–1122, 2001.

[29] J. W. Thomas and et. al. Comparative analyses of multi-species sequences from targeted genomic regions. *Nature*, 424:788–793, 2003.

[30] Paul F. Cliften, LaDeana W. Hillier, Lucinda Fulton, Tina Graves, Tracie Miner, Warren R. Gish, Robert H. Waterston, and Mark Johnston. Surveying *Saccharomyces* genomes to identify functional elements by comparative DNA sequence analysis. *Genome Research*, 11:1175–1186, 2001.

[31] M.S. Gelfand, E.V. Koonin, and A.A. Mironov. Prediction of transcription regulatory sites in *archaea* by a comparative genomic approach. *Nucleic Acids Research*, 28(3):695–705, 2000.

[32] Wyeth W. Wasserman, Michael Palumbo, William Thompson, James W. Fickett, and Charles E. Lawrence. Human-mouse genome comparisons to locate regulatory sites. *Nature Genetics*, 26:225–228, 2000.

[33] Laura Elnitski, Ross C. Hardison, Jia Li, Shan Yang, Diana Kolbe, Pallavi Eswara, Michael J. O'Connor, Scott Schwartz, Webb Miller, and Francesca

Chiaromonte. Distinguishing regulatory DNA from neutral sites. *Genome Research*, 13:64–72, 2003.

[34] Kelly A. Frazer, Laura Elnitski, Deanna M. Church, Inna Dubchak, and Ross C. Hardison. Cross-species sequence comparisons: A review of methods and available resources. *Genome Research*, 13:1–12, 2003.

[35] Inna Dubchak, Michael Brudno, Gabriela G. Loots, Lior Pachter, Chris Mayor, Edward M. Rubin, and Kelly A. Frazer. Active conservation of noncoding sequences revealed by three-way species comparisons. *Genome Research*, 10:1304–1306, 2000.

[36] Roded Sharan, Ivan Ovcharenko, Asa Ben-Hur, and Richard M. Karp. CREME: a framework for identifying cis-regulatory modules in human-mouse conserved segments. *Bioinformatics*, 19:i283–i291, 2003.

[37] Lee Ann McCue, William Thompson, C. Stephen Carmack, and Charles E. Lawrence. Factors influencing the identification of transcription factor binding sites by cross-species comparison. *Genome Research*, 12:1523–1532, 2002.

[38] Manolis Kellis, Nick Patterson, Matthew Endrizzi, Bruce Birren, and Eric S. Lander. Sequencing and comparison of yeast species to identify genes and regulatory elements. *Nature*, 423:241–254, 2003.

[39] E. Wingender, P. Dietze, H. Karas, and R. Knuppel. TRANSFAC: A database on transcription factors and their DNA binding sites. *Nucleic Acids Res.*, 24:238–241, 1996.

[40] Bing Ren, Francois Robert, John J. Wyrick, Oscar Aparicio, Ezra G. Jennings, Itamar Simon, Julia Zeitlinger, Jorg Schreiber, Nancy Hannett, Elenita Kanin, Thomas L. Volkert, Christopher J. Wilson, Stephen P. Bell, and Richard A. Young. Genome-wide location and function of DNA binding proteins. *Science*, 290:2306–2309, 2000.

[41] Tong Ihn Lee, Nicola J. Rinaldi, Francois Robert, Duncan T. Odom, Ziv Bar-Joseph, Georg K. Gerber, Nancy M. Hannett, Christopher T. Harbison, Craig M. Thompson, Itamar Simon, Julia Zeitlinger, Ezra G. Jennings, Heather L. Murray, D. Benjamin Gordon, Bing Ren, John J. Wyrick, John-Bosco Tagne, Thomas L. Volkert, Ernest Fraenkel, David K. Gifford, and Richard A. Young. Transcriptional regulatory networks in *Saccharomyces cerevisiae*. *Science*, 298:799–804, 2002.

[42] Ziv Bar-Joseph, Georg K. Gerber, Tong Ihn Lee, Nicola J. Rinaldi, Jane Y. Yoo, Francois Robert, D. Benjamin Gordon, Ernest Fraenkel, Tommi S. Jaakkola, Richard A. Young, and David K. Gifford. Computational discovery of gene modules and regulatory networks. *Nature Biotechnology*, 21(11):1337–1342, 2003.

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

[44] Jacques van Helden, Alma F. Rios, and Julio Collado-Vides. Discovering regulatory elements in non-coding sequences by analysis of spaced dyads. *Nucleic Acids Research*, 28(8):1808–1818, 2000.

[45] Robert C. Edgar and Kimmen Sjolander. Simultaneous sequence alignment and tree construction using hidden Markov models. In *Pacific Symposium on Biocomputing*, 2002.