COMPUTATIONAL IDENTIFICATION OF DISCRIMINATIVE SEQUENCE MOTIFS WITH DYNAMIC SEARCH SPACES

by

Rahul Karnik

A dissertation submitted to The Johns Hopkins University in conformity with the requirements for the degree of Doctor of Philosophy.

Baltimore, Maryland

September, 2012

Abstract

Regulatory regions in mammalian genomes play important roles both in development and in the maintenance of cellular homestasis. Mutations in these regulatory regions are implicated in several disease phenotypes. Understanding the precise role of these regions requires detailed maps of where regulatory proteins bind to DNA. Experimentally determined genome-wide maps of protein binding are available at fairly coarse resolution, but cannot pinpoint the exact locations in the DNA where the proteins bind. Computational methods can identify the specific putative binding locations within the broader loci and build a model of the DNA sequences to which the protein binds. Yet state-of-the-art computational approches to identify specific DNA binding motifs often yield motifs of weak predictive power. Here we present a novel computational algorithm called INSPECTOR, designed to find specific or predictive motifs, in contrast to over-represented sequence elements. Key distinguishing features of this algorithm are that it uses a dynamic search space to find discriminative motifs and that it models binding motifs using a full PWM (position weight matrix) rather than k-mers or regular expressions. We demonstrate that INSPECTOR finds motifs corresponding to known binding specificities in several mammalian ChIP-seq datasets, but that motifs found by INSPECTOR classify the ChIP-seq signals better than motifs from existing algorithms. We also show that INSPECTOR outperforms a technology-specific algorithm in finding predictive motifs from protein-binding microarray (PBM) datasets. Finally we apply this algorithm to detect motifs from expression datasets in *C. elegans* using a dynamic expression similarity metric rather than fixed expression clusters and find novel predictive motifs.

Advisor:	Michael A. Beer, Ph.D.
Primary Reader:	Michael A. Beer, Ph.D.
Secondary Reader:	Joel S. Bader, Ph.D.
Thesis Committee:	Michael A. Beer, Ph.D.
	Joel S. Bader, Ph.D.
	Jeffry Corden, Ph.D.
	Hongkai Ji, Ph.D.
	Sarah J. Wheelan, M.D., Ph.D.

Acknowledgments

I am immensely grateful for the people in my life, without whom this dissertation would not be possible.

First, I would like to thank my advisor, Mike Beer, for his encouragement, support, patience, and inspiration. He was my mentor and my teacher as I navigated the long road to my dissertation and degree.

I would also like to thank my thesis committee members, Dr. Bader, Dr. Corden, Dr. Ji, and Dr. Wheelan, for their time, guidance, and support during my dissertation process.

My labmates, Dongwon Lee, Mahmoud Ghandi, Donavan Cheng, and Jun Kyu Rhee, were very helpful and supportive of my work, and always had useful and constructive feedback. I appreciate their contributions to my work.

Another person I would like to thank is my undergraduate advisor, Dr. Malcolm A. Campbell, for his mentorship and support, long after my graduation from Davidson College.

I would like to thank my mother, Seema Karnik, who raised me so lovingly and with such great sacrifice, is responsible for so many of my successes, and taught me to value education above all else. Thank you Amu, for all that you have done and continue to do.

My family is next: my aunt, Arati Ranadive, who always loved me and believed in my ability, even when I did not; my sister, Parul Karnik, whose adoration and devotion to my well-being pushes me to be the best brother I can; my in-laws, Geeta Unnikrishnan, Unnikrishnan Ittiampurath, Ammoomma, Rema Devi Purushothaman, and Ramesh Chandran Bhaskarannair, who have been such a source of support during difficult times in recent years; my brothers-in-law, Arvind Nair and Alok Thakur, for their companionship, both intellectual and otherwise; and the rest of my family, too many to mention, who are all a part of my success.

My dog Maggi has quickly become a member of our family, and makes me feel loved so often. Thank you, Maggi, for adopting us.

Last but certainly not least, I am grateful beyond words to my wife, Shalini Unnikrishnan. Her sacrifice and her patience, her encouragement and her competence, her love and her empathy, are an integral part of my success. Dear Shalu, you truly complement my personality, and you make me a better man.

Thank you all so very much, and I hope to make you proud.

Dedication

I would like to dedicate this dissertation to the memory of my father, Vilas Raghunath Karnik, whose example will always inspire, encourage, and challenge me. Baba, I wish you were here to see me attain this goal, but I know I shall always have your blessing and support, wherever you may be.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Multiple mechanisms exist to modulate protein levels in a cell and create a dynamic cellular phenotype from a static genotype. One such mechanism is transcriptional regulation. Transcription factors (TFs) bind to cis-regulatory elements in the vicinity of genes and enhance or inhibit the transcription of those genes into messenger RNA. Therefore, identifying the DNA binding specificities of transcription factors is necessary to decipher the regulatory network in the cell, such that we can identify disease causing mutations in these elements or engineer synthetic organisms to perform specific biochemical functions.

Several technological platforms can be used to identify the binding specificites of the DNA-binding domains of transcription factors to cis-regulatory elements in DNA. We can measure binding in an *in vivo* environment using chromatin immunoprecipitation, followed by measuring binding to a microarray [1] or by deep sequencing [2]. We can also measure binding *in vitro* using protein binding microarrays, or PBMs [3]. Finally, we can often use the expression levels of genes as a proxy for the level of binding of a regulatory factor to their promoters, and thus a set of co-regulated genes is functionally equivalent to a set of genes whose upstream regions are bound by a common transcription factor. Due to the limitations of these technologies, the binding data from these experiments is generally analyzed by motif-finding algorithms that identify the binding specificity of the transcription factor being interrogated to a much higher resolution than the raw data obtained from the technological platform [2–4].

Despite these significant advances in technology, there is still a substantial gap in our ability to generate PWMs which accurately describe binding specificities from these experiments. For instance, Zhu et al [5] used PBMs to investigate the binding specificities of 246 candidate DNA-binding proteins in yeast. Of these, predictive motifs were found in only 89 cases, or 36% of the factors assayed. Similarly, 23 transcription factors from *C. elegans* were assayed using ChIP-seq as part of the modENCODE project [6]. Specific motifs were found for only 8 (35%) of these factors. Considering the number of experiments being done with these technologies [6–9], it would be of great benefit to be able to extract maximal useful information from them.

Generally, motif-finding algorithms search a set of sequences for a shared cis-regulatory element. We term this set of sequences the *search space*. Early motif-finding algorithms optimized for over-represented sequence motifs, which are sequence patterns found more often in the search space than would be predicted by some null or background sequence model. Successful and popular algorithms of this class include ALIGNACE [4] and

MEME [10] which use Gibbs sampling and expectation maximization respectively to search for the optimal sequence motif. In contrast, a discriminative approach searches for specific motifs, or sequence patterns that are present in a positive set of sequences at a higher frequency than in a negative set of sequences. Discrimination reflects the power of the sequence pattern to classify sequences as being part of the search space or not. Recently, a few algorithms of this type have been developed [11, 12]. By necessity, a discriminative objective function is more expensive to compute; it requires scoring not only the sequences in the positive search space, but also those sequences in the negative set to establish the *absence* of the motif in those sequences. Consequently, previous algorithms have used a simplified word-based sequence model for the motif while performing discriminative motif finding. We believe that our algorithm, which we call INSPECTOR, is unique in using a full position-weight matrix as its sequence model, which we show performs better than existing models.

Even if using a discriminative approach, most existing motif finding algorithms consider the positive and negative sets of sequences to be fixed. The positive set is usually defined through one of three methods. First, we might have direct binding data for the protein or TF, and the set of sequences whose binding score is above a threshold is used as the positive set. This threshold can result from the rank or from computation of a p-value given some null binding model. This approach is common with ChIP-chip, ChIP-seq and PBM experiments [2, 5, 13]. Second, we might use a proxy for the binding score, such as the degree of correlation with a shared expression pattern, and choose a threshold for this correlation to define the search space. This method is often used with genome-wide expression profiling data, by clustering genes according to their expression patterns and then running a motif-finding algorithm on the upstream regions of genes in the individual clusters [4, 14]. Lastly, we might simply use prior biological annotation that makes it likely that some sequences are likely to be regulated by the same TF or set of TFs by virtue of being involved in the same biological process. In all these cases, however, the optimal boundary between the positive and negative sets is generally not obvious. Instead of using a fixed set of positive sequences, a dynamic approach allows the boundary between the positive and negative sequences to evolve during the search procedure. A dynamic threshold can be applied when searching for motifs in any set of continuous enrichment data, such as ChIP-seq peak intensity. INSPECTOR uses a dynamic search space optimizing the predictive power of the motif being found, and we show that this flexibility improves the motifs found on PBM and expression datasets.

1.1 Thesis organization

This thesis is organized as follows:

- Chapter 2 explains the significance of TF binding sites and gives an overview of existing motif-finding algorithms, including some general concepts.
- Chapter 3 describes the INSPECTOR algorithm.
- Chapter 4 showcases how INSPECTOR can be used to analyze ChIP-seq data and

how its performance compares to other discriminative motif finders.

- Chapter 5 covers the analysis of PBM data with INSPECTOR.

- Chapter 6 details the use of INSPECTOR to do genome-wide discovery of regulatory elements using expression data.

- Chapter 7 describes the application of INSPECTOR to discover regulatory elements from orthologous sequences in nematodes.

- Chapter 8 summarizes the results from previous chapters and suggests further avenues for research.

Chapter 2

Background

2.1 Importance of cis-regulatory elements

The regulation of transcription is one of several key mechanisms giving rise to diverse cellular phenotypes. Precise spatial and temporal control of transcription can lead to the elegant orchestration of complex processes such as organismal development, while conversely transcriptional misregulation can cause disease phenotypes including cancer [15], inflammatory diseases [16] and Alzheimers disease [17].

A key step in transcriptional regulation is the binding of a transcription factor to a regulatory element a transcription factor binding site (TFBS) in the promoter of a gene (Fig. 2.1). In the case of an activator, the binding event can then either immediately or upon some further cellular signal activate transcription by recruiting additional specific co-activators, chromatin remodelers and general transcription machinery. In the case of a

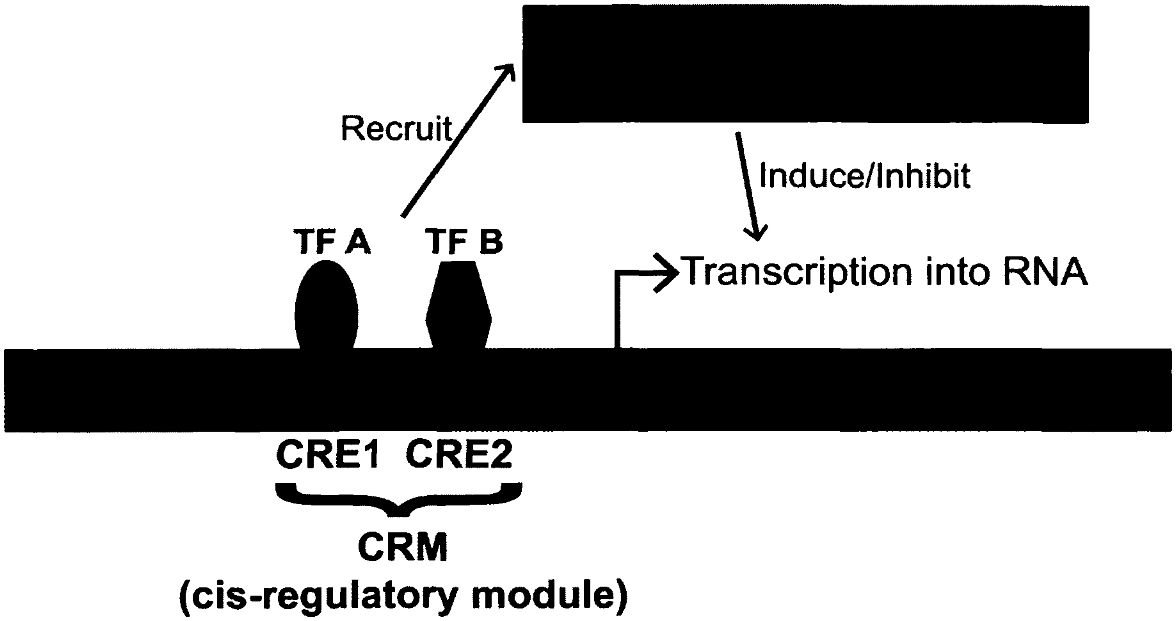


Figure 2.1: **How transcription factors regulate transcription.** Transcription factors bind to cis-regulatory elements in the proximal promoters of genes. They recruit general transcription machinery and chromatin remodelers, and can either induce or repress transcription.

repressor, the transcription factor might preclude the binding of an activating transcription factor or chromatin remodeler. In either case, the DNA sequence to which the transcription factor binds is a crucial element of the process. Knowing the sequence profile to which a transcription factor binds can yield a putative list of its target genes, and eventually lead to the construction of a transcriptional regulatory network [13].

In subsequent chapters, we shall describe experimental technologies that can be used to identify cis-regulatory elements. All these technologies require computational analysis of the data by motif-finding algorithms to find the DNA binding specifities of the TFs being investigated.

2.2 Motif-finding algorithms

The main innovation described in this thesis is the motif-finding algorithm INSPECTOR. INSPECTOR improves existing discriminative motif-finding algorithms by introducing a new dynamic approach to defining the search space and using a full PWM sequence model for motifs. Baseline methods are described here for reference, and subsequent chapters demonstrate how the innovations in INSPECTOR lead to substantially better performance.

2.2.1 Gibbs sampling and expectation maximization

The earliest motif finding algorithms were based on either Gibbs sampling or expectation maximization. ALIGNACE (based on Gibbs sampling) and MEME (based on expectation maximization) are the two most popular algorithms.

ALIGNACE [4] is a slight modification of the Gibbs sampling algorithm as originally described for multiple local sequence alignment [18]. It begins by choosing a random position within a randomly chosen sequence in the search space and initializes a PWM sequence model. This PWM is used to scan all the sequences in the search space and score sites according to the equation below:

$$L = \frac{\Pr(S|\theta)}{\Pr(S|\theta_0)} \tag{2.1}$$

where L is the site score or likelihood ratio, S is the site being scored, θ is a PWM sequence model, and θ_0 is some background distribution. Sites that score above a certain threshold are added to the PWM with a probability that increases according to their score. The process continues iteratively until the objective function — the maximum a posteriori (MAP) score — converges. The MAP score can be represented as

$$MAP = x \log\left(\frac{x}{E}\right) \tag{2.2}$$

where x is the number of instances of the motif in the search space and E is the number of instances that would be predicted by some background model.

MEME [10] considers all k-mers occurring in the search space as being derived either from a PWM representing the optimal motif or from a background sequence model. It initializes from a k-mer that it considers to be over-represented in the search space. It scores all the other k-mers using a log-odds ratio similar to (2.1) and finds high scoring k-mers (E-step). The PWM is then updated to include these high-scoring k-mers (M-step).

The main difference between ALIGNACE and MEME is that ALIGNACE adds sites to the motif model in a stochastic manner, while MEME adds k-mers (which represent sites) in a greedy fashion. Both ALIGNACE and MEME have been used successfully in the past, especially with small sets of genes.

2.2.2 Enumerative motif-finding

A significant drawback of both Gibbs sampling and expectation maximization is the possibility of getting stuck in local minima. The commonly employed workaround is to execute the search multiple times with different random restarts, with the hope that one of the restarts finds the globally optimum motif.

Later algorithms tried to avoid the local minima problem entirely by using a enumerative approach. Instead of a PWM, these algorithms used words or k-mers. All words or k-mers of a certain length k were evaluated using the objective function. The most successful algorithm of this type is WEEDER [19]. It uses suffix trees to search for over-represented k-mers and their mutations and performed very well in a comparative study with 13 other motif-finding algorithms [20].

2.2.3 Discriminative motif finding

The main failing of the algorithms described thus far is that these algorithms only look at the search space or positive set of sequences. The negative set or background is generally summarized using a Markov model (0th order for ALIGNACE, up to 5th order for MEME, 6th or 8th order for WEEDER). Real biological sequences tend to be poorly modeled using these background models, and therefore motifs that are non-specific to the positive set can often end up scoring quite well.

AMADEUS was one of the first discriminative motif finders [11]. It uses an enumerative approach, but its objective function is discriminative, i.e. it evaluates sequence patterns for not only their over-representation in the search space, but also for their absence in the negative set. The objective function is based on the hypergeometric distribution. AMADEUS performed well on a compendium of metazoan target sets [11] relative to

ALIGNACE, MEME and WEEDER. It can also consolidate the k-mers it finds into PWMs and optimize them, though this PWM construction ends up being the most computationally expensive step of the algorithm.

DREME is another algorithm that uses a discriminative approach [12]. It was designed specifically to handle large datasets such as ChIP-seq, where there might be several thousand peaks that are listed as bound. It can use either a true negative set (say a list of unbound peaks) or a scrambled version of the positive set. Like AMADEUS, it uses enumeration, but uses IUPAC regular expressions in addition to simple k-mers. The motifs generated are not true PWMs, but do allow for degeneracy (up to 3 bases) at each position.

Name	Feature space	Algorithm	Objective function
ALIGNACE	PWM	Gibbs sampling	Over-representation
MEME	PWM	Expectation maximization	Over-representation
WEEDER	k-mers	Enumeration, suffix-trees	Over-representation
AMADEUS	k-mer s	Enumeration	Discrimination
DREME	k-mers	Enumeration	Discrimination

Table 2.1: **Summary of motif-finding algorithms**

2.3 Summary

The identification of cis-regulatory elements is crucial to understanding the regulatory network in cells. In this chapter, we have looked at some of the existing motif-finding algorithms, particularly those that exemplify the general principles that have been employed thus far (Table 2.1). In the next chapter, we shall examine the INSPECTOR algorithm in detail, and show how INSPECTOR is novel in combining discriminative motif-finding with a full PWM sequence model. In subsequent chapters, we shall show how INSPECTOR performs relative to the state-of-the-art algorithms described here on different kinds of datasets.

Chapter 3

The INSPECTOR algorithm

INSPECTOR is a heavily modified version of the Gibbs sampling algorithm AlignACE [21]. The key innovations in INSPECTOR are:

1. Dynamic search spaces;
2. Dynamic thresholding of sequence score;
3. Discriminative objective function.

These innovations and their predicted effects are explained below.

3.1 Dynamic search spaces

The search space is the set of sequences that are considered the positive set, and are expected to contain an instance of the DNA sequence element to which the DNA binding protein or TF binds. As mentioned above, current algorithms typically use a fixed set of such sequences.

INSPECTOR attempts to learn the optimal search space instead of using a fixed search space. Given a motif, it redefines the search space such that the motif best discriminates between the search space and the negative set of sequences (Fig. 3.1), as measured by the objective function discussed below. We believe that the dynamic nature of the search space is appropriate, given the widely varying binding specificities of DNA binding proteins in general and TFs in particular.

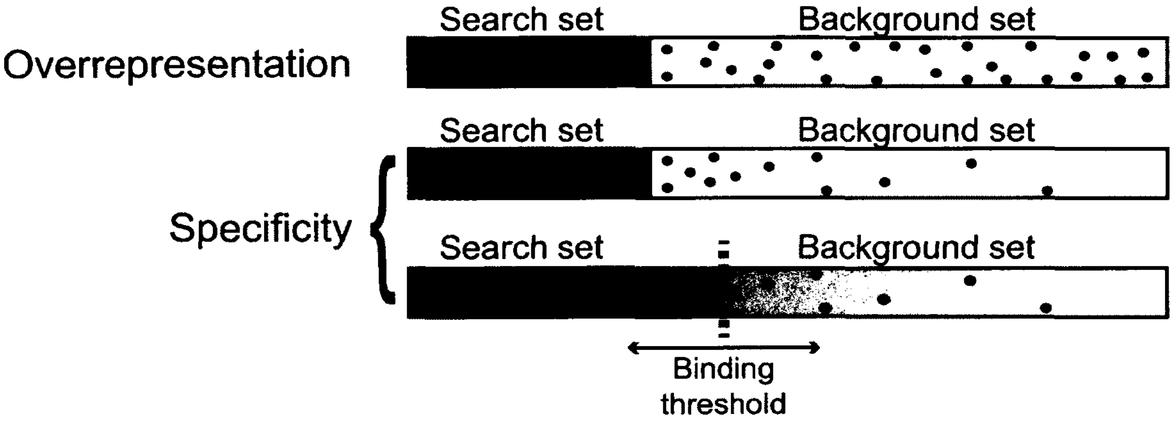


Figure 3.1: **Over-representation vs. specificity.** An over-represented motif is found in the search space more often than expected according to some background sequence model. It is not necessarily predictive and is often found in the search space at frequencies close to its freqency in the background sequences. A specific motif is found in a much higher frequency in the search space than in the background sequences. Since binding data is actually continuous, a dynamic search space threshold finds the optimal search space such that the motif is most discriminative.

3.2 Dynamic threshold for sequence score

In both ALIGNACE and INSPECTOR, sequence positions are scored according to the following equation:

$$L = \frac{\Pr(S|\theta)}{\Pr(S|\theta_0)} \tag{3.1}$$

where L is the site score or likelihood ratio, S is the site being scored, θ is a PWM sequence model, and θ_0 is some background distribution. This odds-ratio is then converted into a probability using a Bayesian framework. We compute the probability of the PWM model θ given that the site S currently being scored as follows:

$$\begin{aligned}
\Pr(\theta|S) &= \frac{\Pr(S|\theta)\Pr(\theta)}{\Pr(S)} \\
&= \frac{\Pr(S|\theta)\Pr(\theta)}{\Pr(S|\theta)\Pr(\theta) + \Pr(S|\theta_0)\Pr(\theta_0)} && (\because \Pr(\theta) + \Pr(\theta_0) = 1) \\
&= \frac{\frac{\Pr(S|\theta)}{\Pr(S|\theta_0)}\Pr(\theta)}{\frac{\Pr(S|\theta)}{\Pr(S|\theta_0)}\Pr(\theta) + \Pr(\theta_0)} \\
&= \frac{L\Pr(\theta)}{L\Pr(\theta) + 1 - \Pr(\theta)} && (3.2)
\end{aligned}$$

In ALIGNACE, a fixed threshold is used to decide whether the site score is good enough for the site to be considered an instance of the current PWM, and thus included in the computation of the PWM for the next iteration. Instead of using a fixed site score threshold, INSPECTOR adjusts the sequence threshold to maximize the discriminative power of the motif, as measured by the objective function discussed below.

In addition, ALIGNACE has a parameter called *expect*, which is the prior for number of instances of the motif in the search space. This parameter is used to calculate $\Pr(\theta)$ in (3.2):

$$\Pr(\theta) = \frac{ew + x(1-w)}{T} \tag{3.3}$$

where e is *expect*, x is the actual number of motif instances according to the current model, w is a weight assigned to the prior and T is the total number of positions available in the search space. INSPECTOR does away with the *expect* parameter; it assumes a "one-instance-per-sequence" model and therefore replaces e with the number of sequences in the current search space, i.e.

$$\Pr(\theta) = \frac{s_1 w + x(1-w)}{T} \tag{3.4}$$

where s_1 is the size of the current search space.

3.3 Model components and objective function

In order to support the dynamic thresholds for site score and search space membership, the INSPECTOR motif model consists of not just a PWM, but also two additional components:

1. **Search space threshold**

   The minimum binding score that a sequence must have to be included in the search space or positive set. The set of sequences that are above this threshold is the set s_1. In the case of Chip-seq this is read depth; for PBM, this score is the binding intensity of the oligo; in the case of expression similarity, this threshold is a correlation measure.

2. **Site score threshold**

   The minimum site score that a site in a sequence must achieve to be considered an instance of the motif. The set of sequences that have a site scoring above this threshold is the set s_2.

INSPECTOR uses an objective function called a *specificity score*, which measures the enrichment for sequences to be in both s_1 and s_2. Given x sequences that are in the intersection of the above sets, and N total sequences in the positive and negative sets, the specificity score is defined using the hypergeometric distribution as

$$\text{Specificity score} = -\log\left(\sum_{i=x}^{\min(s_1,s_2)} \frac{\binom{s_1}{i}\binom{N-s_1}{s_2-i}}{\binom{N}{s_2}}\right). \tag{3.5}$$

The specificity score is similar to the group specificity score used by Hughes et al [4].

We calculate the specificity score in log space using the following equation, which allows us to handle the case of large N where we would otherwise encounter numeric overflows:

$$\log(x+y) = \log(x) + \log[1 + \exp(\log(y) - \log(x))]. \tag{3.6}$$

3.4 Weighted PWM

ALIGNACE and most other motif finders use all instances of a motif found within the search space to compute the PWM, with an equal contribution from each motif. INSPECTOR uses a weighted PWM approach that weights the contribution of each instance to the PWM according to the binding score of the sequence containing that instance.

Say that we have n instances of a motif. Let **w** be the vector of binding scores of the sequences normalized to be between 0 (low) and 1 (high). Assume that the PWM had k columns and that $I_{i,j,b}$ is the indicator variable of having the base b at position j in instance i of the motif. Then the probability $f_{j,b}$ of having base b at position j in the motif in an unweighted PWM is

$$f_{j,b} = \frac{\sum_{i=1}^{n} I_{i,j,b}}{n}. \tag{3.7}$$

The equivalent calculation in the weighted PWM model is:

$$f_{j,b} = \frac{\sum_{i=1}^{n} I_{i,j,b} w_i}{\sum_{i=1}^{n} w_i}. \tag{3.8}$$

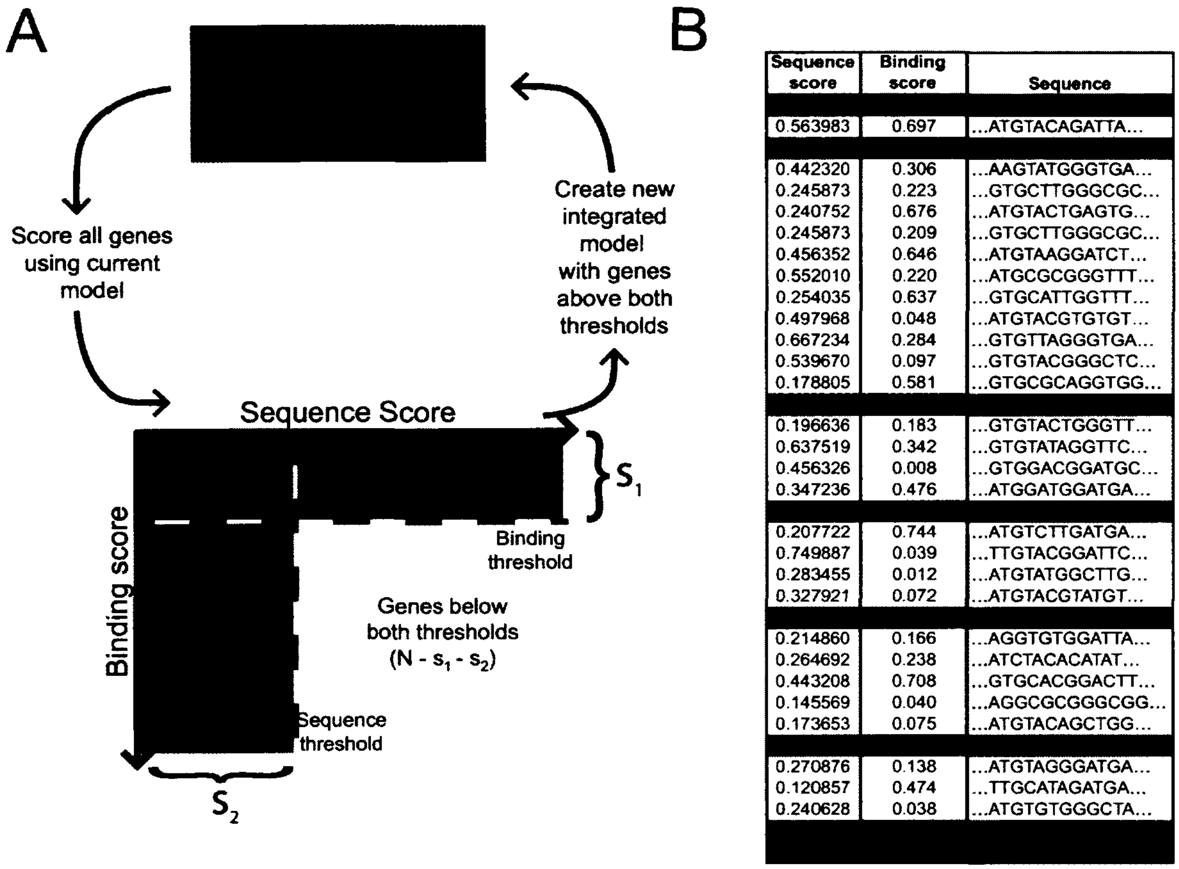
3.5 The background model

ALIGNACE uses a single nucleotide frequency model to calculate the probability of a given site in Eq.(3.1) above, which is equivalent to a 0th order Markov model. Later algorithms have shown that the use of a higher order background model can prove beneficial [22]. In keeping with this trend, INSPECTOR can use a background Markov model with order up to 5. In practice, we tend to use a 3rd order Markov model. Increasing the order of the background model did not result in consistent improvement in performance (see Chap. 4 for an example). Since our objective function penalizes motifs according to their actual frequency in the negative/background set, it is likely that the background model, which is a summary statistic of the background set, is not as important to performance.

3.6 Algorithm

The INSPECTOR algorithm iteratively optimizes the PWM and the thresholds in the model (Fig. 3.2). INSPECTOR initializes the model by choosing a random site from the positive search space. Similar to ALIGNACE, convergence is measured by improvement in the objective function; Inspector stops iterating after a series of *minpass* iterations without improvement have occurred, where *minpass* is a parameter (default 100). INSPECTOR then alternately adjusts the binding score threshold and sequence score threshold to maximize the specificity score, given the current PWM. Sites are re-scored using these new thresholds. With the new thresholds, the iteration process is repeated, again until *minpass* iterations without improvement occur, and the current model is output. After the first motif is found, subsequent searches are performed with new random starts, terminating early if the current model is similar (COMPAREACE score greater than a threshold, default 0.9) to a motif previously found with a higher specificity score. The number of such restarts is

$S_1/(w * k)$, where S_1 is the size of the search space (in base pairs), w is the number of columns in the PWM motif model, and k is a sampling parameter.


A
B
Score all genes using current model
Create new integrated model with genes above both thresholds
Sequence Score
Binding score
s_1
Binding threshold
Genes below both thresholds ($N - s_1 - s_2$)
Sequence threshold
s_2

Sequence score	Binding score	Sequence
0.563983	0.697	...ATGTACAGATTA...
0.442320	0.306	...AAGTATGGGTGA...
0.245873	0.223	...GTGCTTGGGCGC...
0.240752	0.676	...ATGTACTGAGTG...
0.245873	0.209	...GTGCTTGGGCGC...
0.456352	0.646	...ATGTAAGGATCT...
0.552010	0.220	...ATGCGCGGGTTT...
0.254035	0.637	...GTGCATTGGTTT...
0.497968	0.048	...ATGTACGTGTGT...
0.667234	0.284	...GTGTTAGGGTGA...
0.539670	0.097	...GTGTACGGGCTC...
0.178805	0.581	...GTGCGCAGGTGG...
0.196636	0.183	...GTGTACTGGGTT...
0.637519	0.342	...GTGTATAGGTTC...
0.456326	0.008	...GTGGACGGATGC...
0.347236	0.476	...ATGGATGGATGA...
0.207722	0.744	...ATGTCTTGATGA...
0.749887	0.039	...TTGTACGGATTC...
0.283455	0.012	...ATGTATGGCTTG...
0.327921	0.072	...ATGTACGTATGT...
0.214860	0.166	...AGGTGTGGATTA...
0.264692	0.238	...ATCTACACATAT...
0.443208	0.708	...GTGCACGGACTT...
0.145569	0.040	...AGGCGCGGGCGG...
0.173653	0.075	...ATGTACAGCTGG...
0.270876	0.138	...ATGTAGGGATGA...
0.120857	0.474	...TTGCATAGATGA...
0.240628	0.038	...ATGTGTGGGCTA...

Figure 3.2: **The INSPECTOR algorithm.** (A) A schematic of the INSPECTOR algorithm. The PWM model is initialized with a random sequence and position in the search space. The model is iteratively refined and the motif and binding score thresholds are adjusted at convergence to maximize specificity. (B) An example of sequences scored using the model. Each sequence has a motif score and a binding score. The binding score is compared to the binding score threshold (here, 0.8) to determine if a sequence is in the search space. The sequence score is compared to the sequence score threshold (here, 0.8) to determine if the sequence has an instance of the motif. The sequences are color-coded according to the set to which they belong as defined in (A).

3.7 Gapped motifs and number of columns

One advantage that ALIGNACE provides over competing algorithms is the ability to find gapped motifs. The PWM model is not necessarily contiguous bases and can actually include gaps. For example, the width of the motif might be 15 bases, but only 10 of these bases might be informative and therefore included in the PWM. This gapped PWM model is useful for TFs that bind as a dimer to two sets of constrained DNA bases separated by unconstrained bases. For example, the yeast TF Gal4 binds to the sequence pattern CGGnnnnnnnnnnnGGC.

INSPECTOR takes the gapped motif concept one step further: it has the ability to add or remove columns to the PWM randomly during the motif search to see if it improves the specificity score. We use this setting for genome-wide searches using expression data, since we are looking for novel motifs that may have an unpredictable number of informative columns.

3.8 Column sampling

Like ALIGNACE, INSPECTOR implements a feature called column sampling that looks for informative columns adjacent to existing motif instances to address the phase shift problem [18]. The phase shift problem occurs when a Gibbs sampler find a locally optimal solution that is a shifted version of the global optimum, i.e. the pattern found overlaps with, but is not identical to, the optimal motif. The column sampling procedure is described below.

Let w be the initial motif width and k be the number of columns in the PWM. We begin with $w = k$, though this can change after column sampling occurs. After every iteration, INSPECTOR creates a set of extended motif instances of width $w_s = 3w$. For each column in this span, INSPECTOR computes the information content and the new width w' of the motif if this column were to be added. It then weights the information content by the following function suggested in [23], which is the number of ways of picking k columns, given that the width of the motif is w'.

$$\rho(w') = \binom{w' - 2}{k - 2}^{-1} \tag{3.9}$$

As ρ decreases with w', this weighting scheme results in columns that are further away from existing motif instances being penalized and preferring shorter motifs overall. Finally, INSPECTOR creates a ranked list of columns in the span w_s and chooses the k best columns to create a new PWM.

3.9 Motif comparison

INSPECTOR uses a similarity measure based on the Pearson correlation coefficient to create a non-redundant set of motifs. Two motifs are passed to the motif comparison function as sets of motif instances or sites.The similarity measure is calculated as follows:

1. Extract aligned instances of each PWM from the original sequences along with 10 bp of sequence on ether side of each instance. Let these be A_1 and A_2 for the first and second motif respectively.

2. Within A_1, find the 6 columns (possibly with gaps) with the highest information content for the first motif and create a PWM M_1 for these columns.

3. For each possible set i of 6 columns in A_2 with the same gap configuration as M_1, create a PWM $M_{2,i}$.

4. For each set i, calculate the Pearson correlation coefficient C_i between M_1 and $M_{2,i}$.

5. The similarity measure is the maximum of all C_i.

The similarity measure described above is much like that used for motif comparison in ALIGNACE and COMPAREACE [4].

3.10 Performance heuristics

Given the definition of the specificity score, it is obvious that each iteration of the INSPECTOR algorithm requires the scanning of N total sequences for motif instances, unlike over-representation based algorithms such as ALIGNACE and MEME which would scan the search space alone. Since each random start of the INSPECTOR algorithm is independent, however, we can parallelize the searches and essentially reduce running time by a factor equaling the number of parallel instances of INSPECTOR (Fig. 3.3).

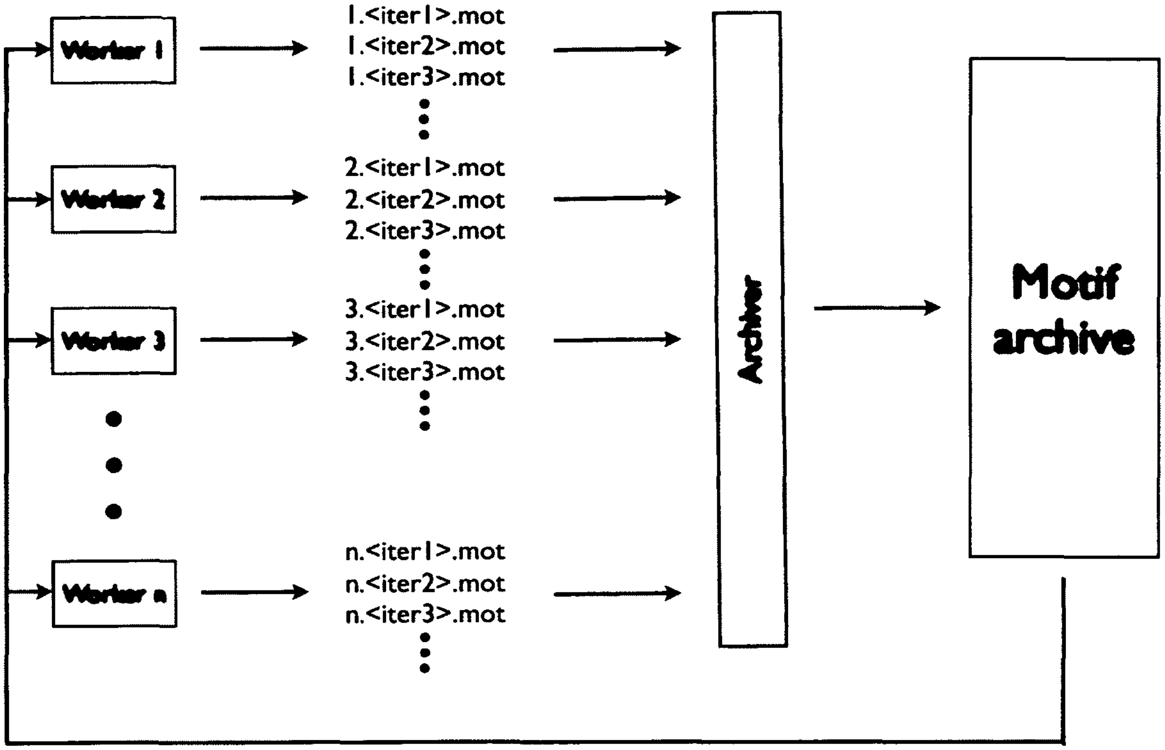


Figure 3.3: **INSPECTOR uses multiple worker processes to parallelize the motif search process.** The worker threads output the motifs found, which are collected by an archiver process that creates a non-redundant archive of motifs. The motif archive is fed back into the worker processes for early termination of searches that are similar to a motif that has already been found.

Another heuristic used to increase scanning performance is the maintenance of a list of the highest scoring site in each of the input sequences. Since only the highest scoring site determines membership in the set of sequences containing an instance of the motif, we do not need to scan every position of every sequence. Instad, we only scan the position that was the highest scoring site in each sequence in the previous iteration. If the PWM has changed considerably from the previous iteration ($\Pr(\theta|S) < 0.8$), INSPECTOR then triggers a full scan of all positions in all sequences.

3.11 Ways to run INSPECTOR

INSPECTOR can run in three modes:

1. **Subset mode**

   Appropriate for any situation with sequences that have been classified in a binary fashion into positive and negative sets. In subset mode, INSPECTOR takes a sequence file in FASTA format, along with a list of sequence names that is a subset of those in the sequence file. This list of sequences is used as the fixed search space. With a fixed search space, the weighted PWM calculation is equivalent to the unweighted PWM since all sequences in the search space have equal weight. However, the sequence score threshold is calculated dynamically. We have used this mode for ChIP-seq datasets.

2. **Score mode**

   To be used whenever we have some continuous measure of binding strength for each sequence. In score mode, INSPECTOR takes a sequence file and a file with sequence names and binding scores. All sequences in the sequence file should have corresponding binding scores in the score file. In this mode, INSPECTOR uses a dynamic search space and weighted PWM calculation as described above, along with dynamic sequence score thresholds. We have used this mode for PBM datasets.

3. **Expression mode**

   Finds co-regulated genes with a shared cis-regulatory element in their upstream sequences. In expression mode, INSPECTOR takes a sequence file with upstream sequences (or other regulatory regions) and a file with expression profiles. All sequences in the sequence file should have corresponding expression profiles.

3.12 Performance

If N_p is the number of base pairs in the positive set or search space, and N_n is the number of base pairs in the negative set, each motif search performed by INSPECTOR takes $O(N_p + N_n)$ time. The upper bound on the number of such searches needed is $O(N_p)$, and can be much less if the motif is found in one of the early random starts performed. Memory requirements are $O(N_p + N_n)$, and are on the order of a few hundred MB.

In practice, we have run Inspector in 3-4 hours on ChIP-seq datasets with about 30,000 peaks (positive + negative set), with each peak of length 300 bp. In expression mode, we have run Inspector on sequence datasets with 15,000 upstream sequences and expression datasets across about 100 conditions: these searches take 4-5 days.

3.13 Software availability

Source code for Inspector and documentation describing installation and operating instructions are available from our website: http://www.beerlab.org/inspector.

3.14 Summary

INSPECTOR is a novel motif-finding algorithm that uses a discriminative objective function and a dynamic search space along with a full PWM sequence model. In the following chapters, we apply the INSPECTOR algorithm to three different kinds of datasets—ChIP-seq, PBM and expression—and demonstrate the superior performance of INSPECTOR relative to existing state-of-the-art motif-finding algorithms.

Chapter 4

Motif finding with ChIP-seq

The availability of next-generation sequencing has resulted in a proliferation of ChIP-seq datasets. This technique has been applied to find binding sites for transcription factors [2], transcriptional co-activators [24], general transcriptional machinery [25], and chromatin states [26]. ChIP-seq is one of the main techniques being used to understand regulatory regions of the genome in the ENCODE (Encyclopedia of DNA Elements) [7] and modENCODE projects [6, 8]. There is thus a pressing need to develop and improve computational techniques that can extract useful information from ChIP-seq datasets. We show that INSPECTOR is able to find highly predictive motifs from ChIP-seq data, and that it improves on the existing algorithms being used for such analysis.

4.1 Background

Chromatin immunoprecipitation (ChIP) is an experimental technique that measures *in vivo* binding of proteins to genomic DNA (Fig. 4.1). The first step in a ChIP-seq experiment is the addition of a nuclear protein extract to genomic DNA, followed by cross-linking to reversibly stabilize protein-DNA interactions. The DNA is then sheared by sonication and immunoprecipitated with an antibody specific to the protein of interest. After washing away unbound DNA, the cross-linking is reversed and the protein is digested away. The end result of this process is DNA that represents the broad regions (300-1000 bp) that were bound by the protein of interest.

To identify these DNA regions, we can use either hybridization to a microarray (ChIP-chip) [1] or deep sequencing (ChIP-seq) [2]. As deep sequencing has become more affordable, ChIP-seq has become the method of choice, as it is easier to simply sequence the DNA fragments than design microarrays that can ensure adequate coverage of large genomes. Once the fragments are sequenced, we can map them to known genomic locations and get read counts across the genome. As a control to identify sequencing biases that can affect the intensity readout, the procedure is repeated with input genomic DNA and the immunoprecipitation step is skipped.

At this point, a peak-calling program such as PeakSeq [27] or MACS [28] is used to convert mapped reads into peaks that represent actual binding locations on the order of 100-200 bp. In order to find the exact DNA sequences (6-20 bp) that were bound by the protein, we can use a motif finding algorithm on some set of the peaks that we consider

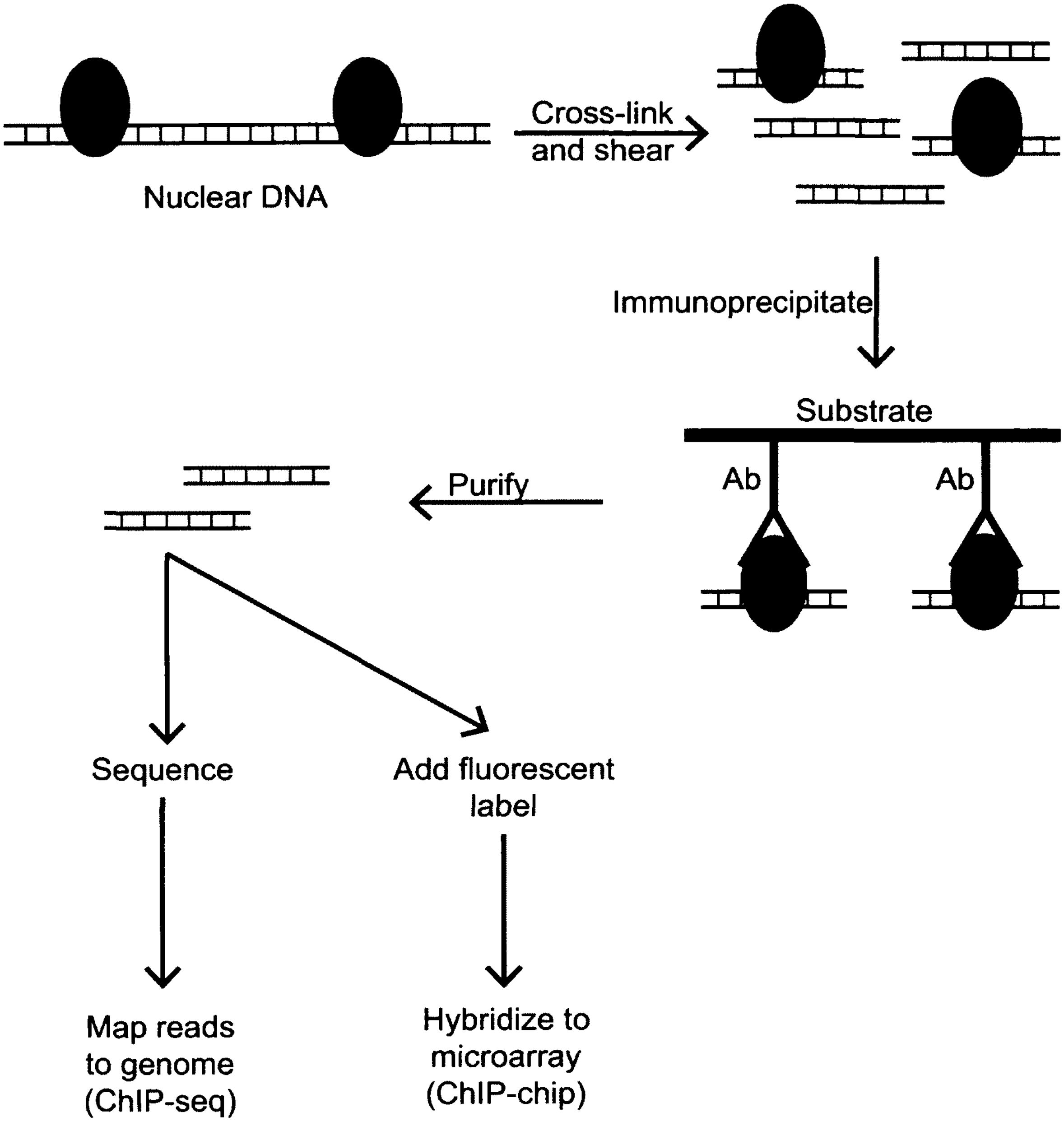


Figure 4.1: **Using ChIP to find genomic binding locations of proteins.**

as definitively bound. This set of peaks can be based on rank ("top n peaks"), but in experimental design that include a control, we can calculate a false discovery rate (FDR) for each peak. The FDR can be used to standardize the stringency with which peaks are classified as as bound.

Traditional motif finding algorithms such as MEME and WEEDER can identify over-represented sequence motifs from ChIP datasets, but are limited in their ability to handle large numbers of sequences [12]. Best practice when using these algorithms is to choose a small number (200-500) of the highest scoring peaks (lowest p-value or FDR). While this approach works reasonably well to identify sequence patterns corresponding to the sites that are bound strongest, it underestimates the sequence diversity to which the protein can actually bind.

Increasingly, the trend in ChIP-seq analysis is to use discriminative motif finders such as AMADEUS and DREME. These algorithms can use the full set of ChIP-seq peaks (on the order of 1000s) and still find motifs in a useful timeframe. There is however room for improvement. ChIP-seq was done for 23 worm TFs as part of the modENCODE project, but predictive motifs were found for only 8 of these TFs. Assuming that the datasets themselves are of good quality, a better algorithm may find more predictive motifs.

4.1.1 Human transcription factors

In order to compare INSPECTOR with the motif-finding algorithms DREME and AMADEUS, we used ChIP-seq datasets for three human TFs:

1. CTCF

   CTCF (CCCTC-binding factor) is a zinc-finger protein that has multiple regulatory functions including transcriptional regulation and insulation [29]. While most TFs have binding sites in proximal promoters, the vast majority of CTCF binding sites are found quite far away from coding sequence, possibly indicating that its mechanism of action involves long-range genomic interactions. Therefore, expression-based methods such as those described in Chap. 6 would be unlikely to find CTCF binding sites.

2. NRSF

   Neuron-restrictive silencing factor (NRSF) (also called REST) is a repressor of neuronal gene expression in mammals [30,31]. It binds to a 21 bp DNA sequence element called the neuron-restrictive silencing element, or NRSE, which was initially identified by deletion analysis of the promoter of the neuronal protein SCG10 [31] as having the consensus sequence TTYAGHACCRCGGASAGHSCC. NRSF was the first mammalian protein whose genomic binding locations were determined by ChIP-seq [2].

3. ER

   The estrogen receptor (ER) is a nuclear hormone receptor with two mammalian forms, ER-α and ER-β [32]. ER can function as a homodimer or a heterodimer, and activates transcription in its target genes when activated by its ligand, estradiol (E2). The E2-ER complex binds to estrogen response elements (EREs), classically represented by the palindromic consensus sequence CAGGTCAnnnTGACCTGA.

4.2 Methods

4.2.1 Human ChIP-seq data

We used publicly available ChIP-seq data for three human transcription factors:

1. CCCTC-binding factor (CTCF) binding in the Gm12878 cell line [33];

2. neuron-restrictive silencer factor (NRSF/REST) binding in the Jurkat T cell line [2];

3. the estrogen receptor (ER-α) in the MCF-7 cell line [25].

For each dataset, we downloaded the raw data from the Gene Expression Omnibus (GEO) database. We reprocessed the raw data using the MACS algorithm [28] with default parameters and used a stringent false discovery rate threshold of less than 0.01% to designate high confidence peaks as the positive set. This process resulted in positive sets with 5444, 2417 and 1225 peaks for CTCF, NRSF and ER respectively. A 300 bp window of genomic sequence around each peak was used for analysis.

4.2.2 Mouse ChIP-seq data

We used previously published ChIP-seq data for thirteen mouse transcription factors in mouse embryonic stem cells [34]. For the mouse ESC datasets, we used the set of bound sequences defined by [34] as the positive set. These datasets had been previously analyzed with the DREME algorithm using this exact positive set [12], and therefore we used the same parameters so as to compare with DREME's performance.

4.2.3 Worm data

We analyzed ChIP-seq data for *C. elegans* transcription factors from the modENCODE project [6]. We used the peaks designated in [35] as appropriate for downstream analysis, so as to allow for optimal comparison with the results from that study.

4.2.4 Negative sets

We have the freedom to select negative sets larger than the positive sets, which usually lead to more predictive motifs. For each positive set, we generated a negative set with twice the number of sequences as the positive set. The sequences in the negative set were chosen randomly from the genome and matched in length, repeat fraction and GC content with the sequences in the positive set using a previously published procedure [36].

4.2.5 Analysis with motif finding algorithms

We analyzed the human ChIP-seq datasets with DREME, AMADEUS and INSPECTOR. The mouse datasets were analyzed with INSPECTOR, and the results compared to previously published results using DREME. The *C. elegans* datasets were analyzed using INSPECTOR alone, since for most of the 26 TFs assayed, there was no gold standard with which to compare the results.

DREME was run using both the positive and negative datasets. DREME has a mode in which a scrambled version of the sequences in the positive set are used as the negative set, but in order to match the inputs to the other algorithms, we used a true negative set. DREME requires parameter specifying the number of columns in the motif, and we used a range from 6 to 20 columns ("-mink 6 -maxk 20"). The highest scoring motif reported regardless of length was used for comparison.

AMADEUS was run using the combined positive and negative sets as the background and the positive set as the search space. AMADEUS does not have a way of doing a single run with different numbers of columns in the motif, so we performed independent runs with 6 through 20 columns. The top ranking motif from each run was evaluated and the highest scoring motif used for comparison.

INSPECTOR was run in fixed search space mode, with the positive set as the search space and the positive and negative sets as background. We used a 3rd order background model. We tried background models of order 0 through 5 on the human ChIP-seq datasets, and the impact of the background model on the predictive power was minimal and showed no consistent trend (Fig. 4.2).

4.2.6 Motif evaluation

We scanned the positive and negative sets with the motif being evaluated using ScanACE and ranked the sequences according to the highest scoring site in each se-

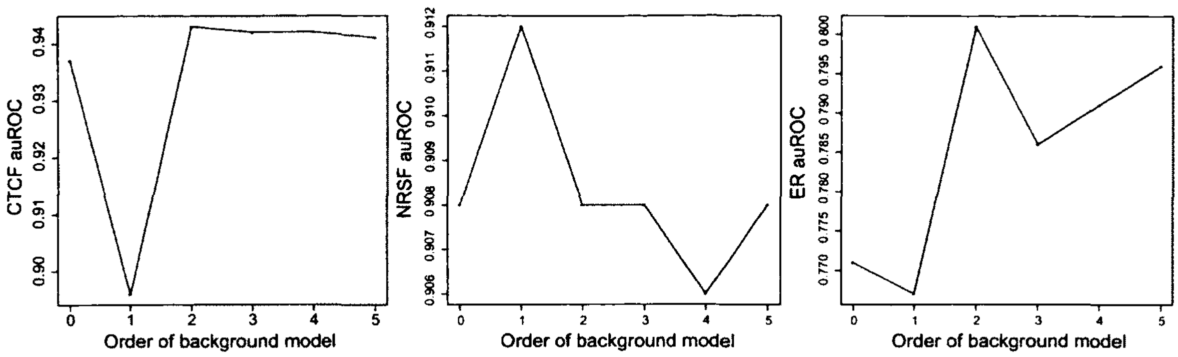


Figure 4.2: **Effect of background model on INSPECTOR ChIP-seq results** The graphs show the impact on auROC for three datasets of the order of the background model. There was very little change in auROC with the use of higher-order background models.

quence. We used this ranking to plot a receiver operator characteristic (ROC) curve and used the area under the ROC curve (auROC) as a measure of how well a motif was able to discriminate between the positive and negative sets.

4.3 Results

4.3.1 CTCF

We searched for motifs in ChIP-seq data for CTCF using DREME, AMADEUS and INSPECTOR. The top motif found by each algorithm was evaluated and the auROC values compared.

The motif found by INSPECTOR was by highly predictive, with an auROC value of 0.941. No motif in JASPAR was more predictive on this dataset: the canonical motif (JASPAR ID MA0139.1) itself had an AUC of 0.938. In comparison, the motifs found by

DREME and AMADEUS were much less predictive, with auROC values of 0.73 and 0.88 respectively. The ROC curves for all three motifs and the motifs themselves are shown in Fig. 4.3.

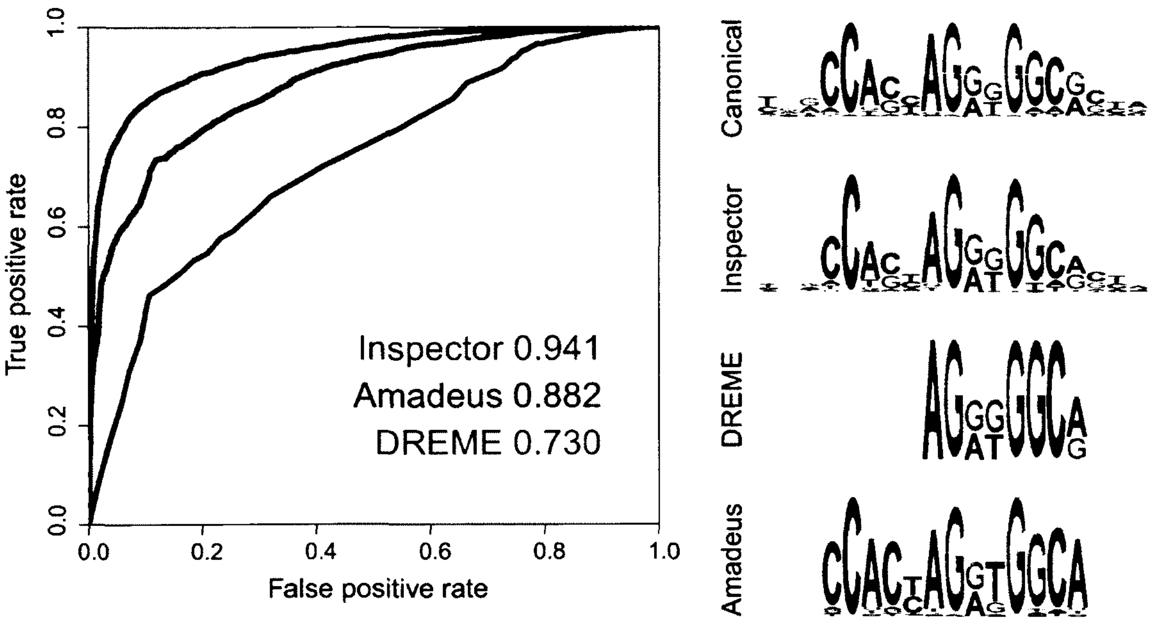


Figure 4.3: **INSPECTOR finds a more predictive CTCF motif than DREME or AMADEUS** On the left are the ROC curves for the top motifs found by each algorithm, with INSPECTOR in green, DREME in red, and AMADEUS in brown. The auROC values are listed on the ROC plot. On the right are the actual motifs found, with the canonical motif at the top.

We examined the actual motifs found from this dataset by the three algorithms. It seems that DREME finds the core CTCF motif, but not the adjacent bases. AMADEUS does find the whole motif, but has several positions where the PWM is more stringent than the canonical motif, and this stringency presumably prevents it from finding weaker instances of the motif. INSPECTOR finds the motif closest to the canonical motif, including slightly informative positions at the edges of the motif.

4.3.2 NRSF

We analyzed ChIP-seq data for NRSF using DREME, AMADEUS and INSPECTOR as described for CTCF above. The INSPECTOR motif was once again the most predictive with an auROC of 0.906. While the auROC for the AMADEUS and DREME motifs were lower (0.886 and 0.84 respectively) the INSPECTOR motif was more discriminative by a smaller margin than in the case of CTCF. However, when comparing the ROC curves themselves, it is clear that the motif found by INSPECTOR shows much higher discriminative power than the other two motifs. The ROC curves and motifs are shown in Fig. 4.4.

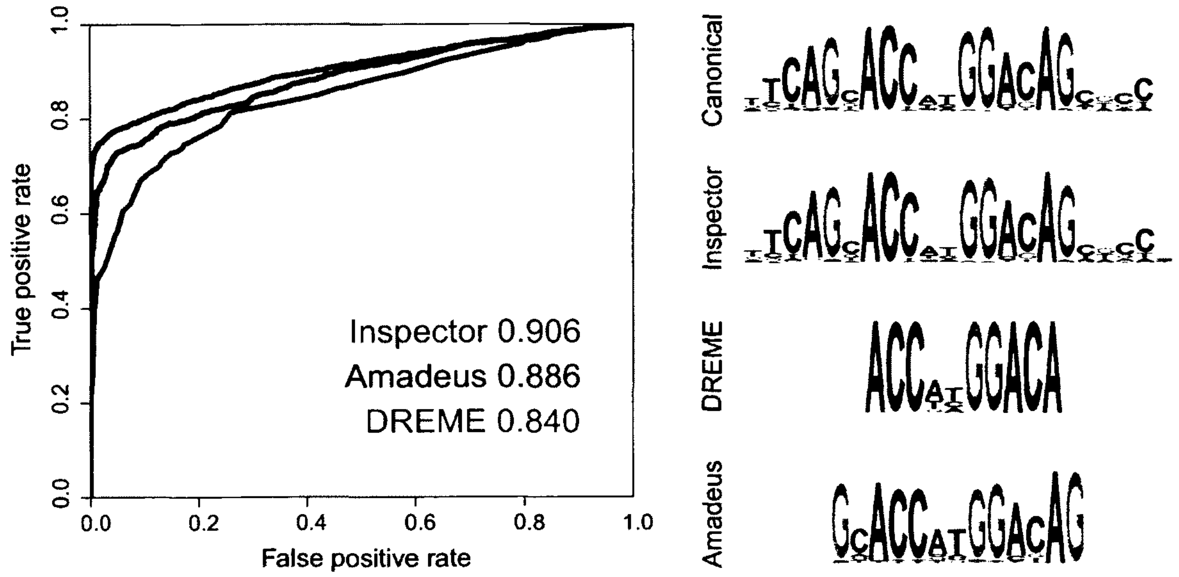


Figure 4.4: **INSPECTOR finds a more predictive NRSF motif than DREME or AMADEUS** On the left are the ROC curves for the top motifs found by each algorithm, with INSPECTOR in green, DREME in red, and AMADEUS in brown. The auROC values are listed on the ROC plot. On the right are the actual motifs found, with the canonical motif at the top.

As with CTCF, the motif found by INSPECTOR is almost exactly identical to the canonical NRSF motif from JASPAR (MA0138.1). Both the DREME and the AMADEUS motifs are shorter than the canonical and INSPECTOR motifs.

4.3.3 ER

We also analyzed ChIP-seq data for the estrogen receptor (ER) using the three algorithms. Once again, INSPECTOR found a more predictive motif (auROC=0.796) compared to DREME (0.745) or AMADEUS (0.760). Examination of the ROC curves shows that the difference lies in the important part of the ROC curve, i.e. at low false positive rates. The INSPECTOR motif recovers 59% of the positive regions at a false positive rate of 10%, compared to about 45% for the AMADEUS and DREME motifs.

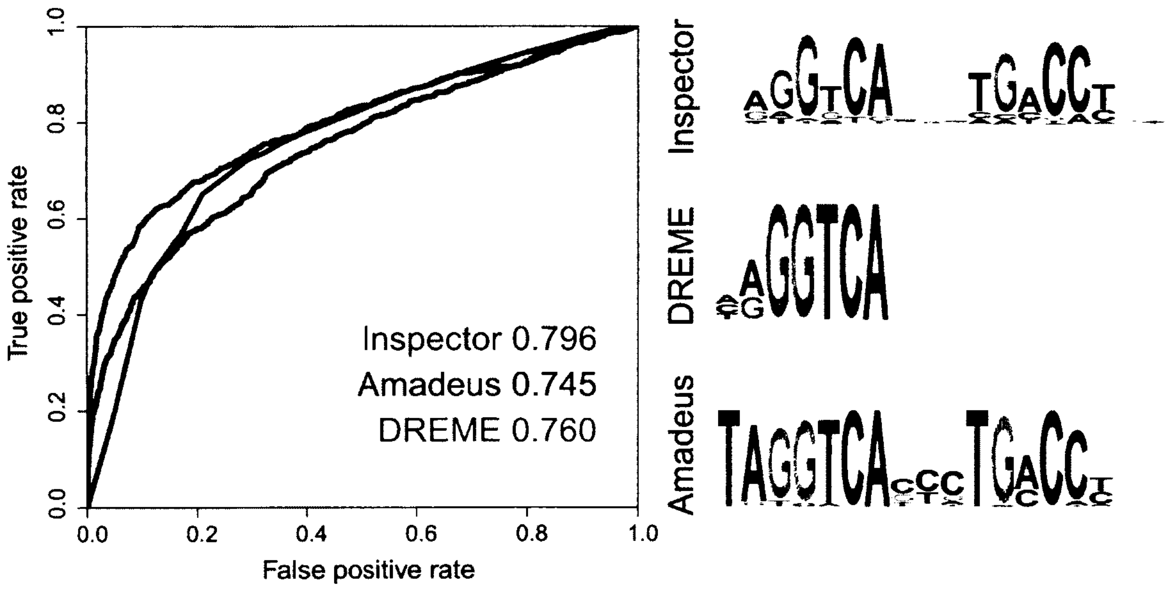


Figure 4.5: **INSPECTOR finds a more predictive ER motif than DREME or AMADEUS** On the left are the ROC curves for the top motifs found by each algorithm, with INSPECTOR in green, DREME in red, and AMADEUS in brown. The auROC values are listed on the ROC plot. On the right are the actual motifs found.

Both INSPECTOR and AMADEUS find the full motif (AGGTCAnnnTGACCT), while DREME only found the half motif (AGGTCA). As in the case of CTCF, the AMADEUS motif seems to be too stringent in certain positions, such as the thymine to the left of the first half-motif, and assigns a high probability to cytosines for the middle three nucleotides. The INSPECTOR motif seems to be highly symmetric; corresponding positions in each half motif are equally degenerate. This quality is not shared by the AMADEUS motif.

4.3.4 Summary of human ChIP-seq results

We note that INSPECTOR consistently finds longer and more discriminative motifs than either DREME or AMADEUS, although all three algorithms are trying to optimize discriminative objective functions. It is not clear if there is a systematic bias towards shorter motifs inherent in the DREME and/or AMADEUS algorithms. As these algorithms are word-based, we speculate that finding high-scoring exact matches to long k-mers or regular expressions is not likely and that these long words get filtered out at an early stage in the algorithm. On the other hand, as a entirely PWM-based algorithm, INSPECTOR is able to detect these longer motifs even with mismatches as instances of the PWM. Alternatively, it is possible that as k increases, the number of cases per k-mer decreases, reducing the sensitivity of DREME and AMADEUS to longer motif signals.

We wanted to ensure that the better discriminative power of the motifs found by INSPECTOR was not due to overfitting, as a PWM based motif model has more parameters than a motif model based on words or regular expressions. We performed 5-fold crossvalidation on the three datasets using INSPECTOR. The motifs found were almost identical to those found using each complete dataset, and the auROC for retrieval of the test set was the same as that for the whole dataset in each case.

4.3.5 Mouse transcription factors

We next ran Inspector on 13 TF ChIP-seq data sets generated in mouse embryonic stem cells [34], and compared our motifs to the previously published motifs found using DREME [12]. Motifs found by Inspector were similar, but not identical, to those found by DREME in each dataset. We evaluated the ability of the top motif in each dataset to recover positive sequences from each dataset as above using auROC. In 10 of the 13 datasets, the top motif found by Inspector had higher auROC than the top motif found by DREME (Fig. 4.6). In the three remaining cases (Zfx, Klf4, Esrrb), the auROCs were almost identical. For some cases, the auROC values for the best single motifs are low, indicating more combinatorial regulation by those factors.

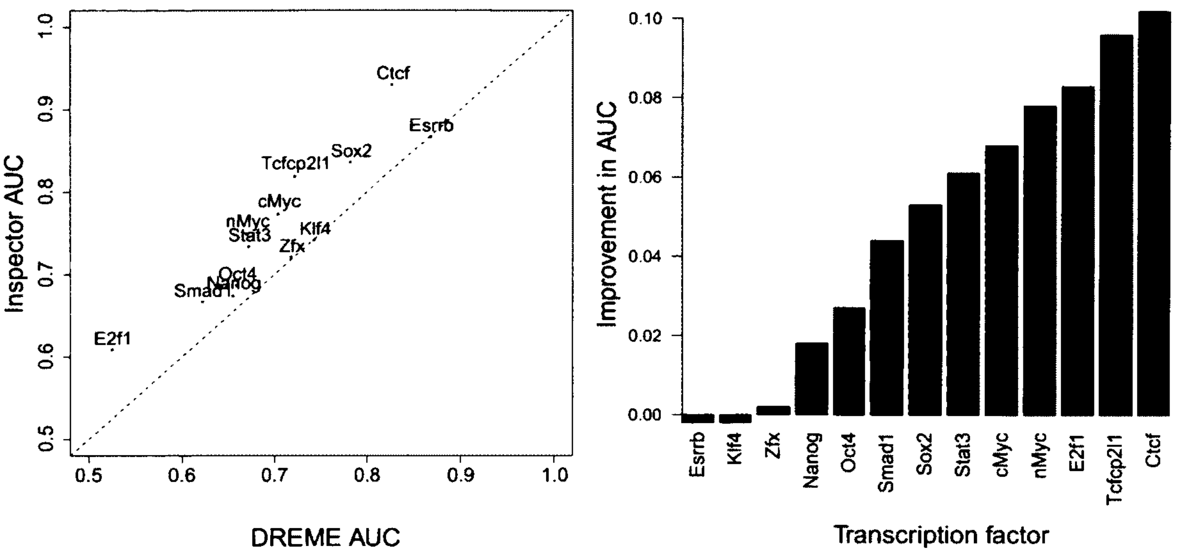


Figure 4.6: **Mouse ChIP-seq results** INSPECTOR outperforms DREME when run on ChIP-seq data for 13 transcription factors from mouse embryonic stem cells. The left panel shows a plot of the AUC for the top motif reported by INSPECTOR against the AUC for the top motif reported by DREME, while the right panel shows the improvement in AUC for the INSPECTOR motif relative to the DREME motif.

4.3.6 Worm transcription factors

We ran INSPECTOR on the ChIP-seq data for 23 worm *C. elegans* from the modENCODE project [35]. We were able to find discriminative binding specificities for four factors, ELT-3, GEI-11, LIN-15B, and PQM-1, that were not reported in the original analysis (Fig. 4.7). The motif that we found for LIN-15B was also one of the top motifs in a genome-wide search for cis-regulatory elements using expression data, increasing the likelihood that it is functional.

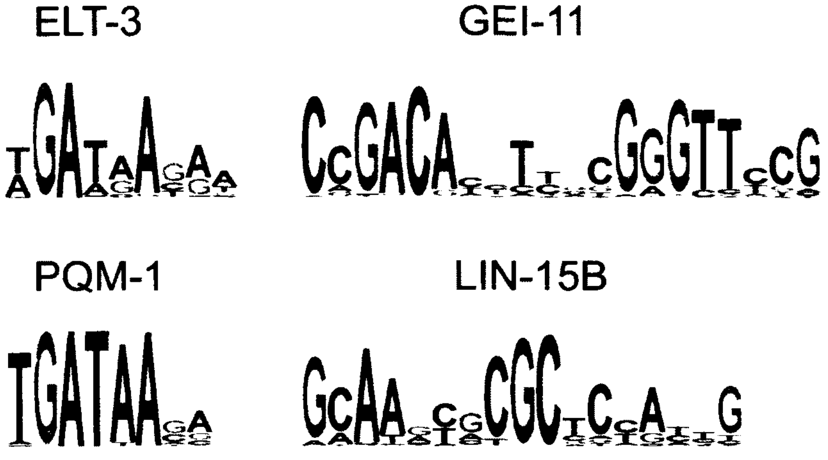


Figure 4.7: *C. elegans* **modENCODE ChIP-seq results** Binding specificities for 4 *C. elegans* transcription factors as learned by INSPECTOR from ChIP-seq data from the modENCODE project.

4.4 Summary and Discussion

We ran INSPECTOR and two other discriminative motif-finding algorithms, DREME and AMADEUS, on three human ChIP-seq datasets. Even when using a fixed search space, the use of a full PWM model results in INSPECTOR finding more predictive motifs than those found by the other algorithms. The improvement in discriminative power was even more evident when looking at the critical part of the ROC curve, i.e. sensitivity at low false-positive rates.

We also ran INSPECTOR on a set of 13 mouse ChIP-seq datasets in mouse embryonic stem cells. We compared the motifs found with those found by DREME on the same datasets. Once again, the motifs found by INSPECTOR predicted the data better than those found by DREME.

The results above demonstrate the power of the full PWM motif model as compared to a k-mer or regular expression motif model. To reiterate, the dynamic search space mode was not used here, and yet INSPECTOR found better motifs than other state-of-the-art motif finders that use simpler motif models. The TFs that were assayed had long motifs, and perhaps these motifs challenge the k-mer/regular expression models more than shorter motifs would. To test this idea, we plotted the improvement in auROC (Inspector vs. DREME) for the mouse ChIP-seq datasets against the length of the motif (Fig. 4.8).

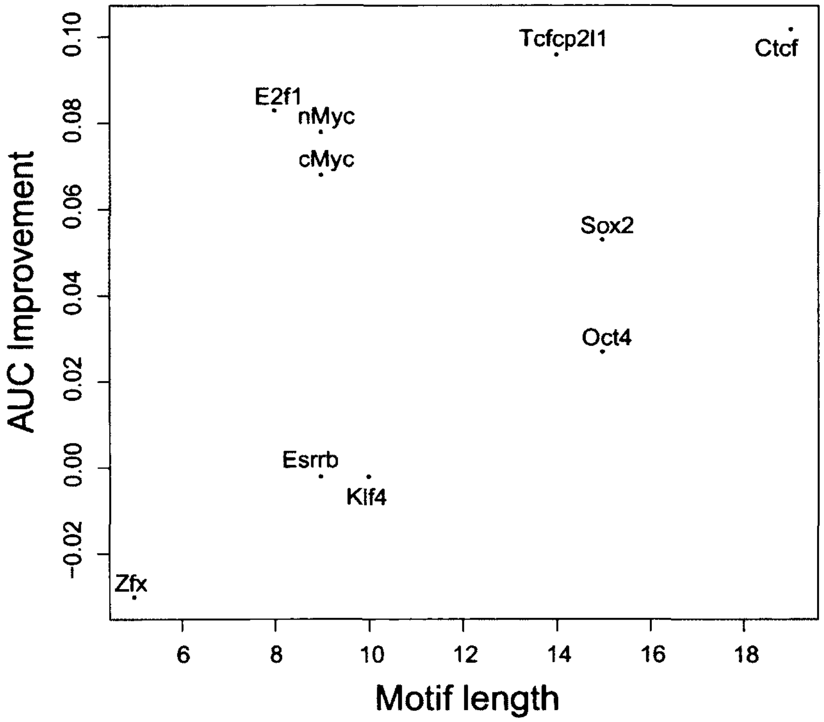


Figure 4.8: **auROC improvement vs. motif length.** INSPECTOR's improvement in auROC over DREME is not correlated with motif length on the mouse ChIP-seq datasets.

Finally, we analyzed ChIP-seq datasets from *C. elegans* generated by the modENCODE project. Motifs for 8 of the 26 TFs assayed had been previously published; we present here

4 new motifs found by INSPECTOR. One of these motifs was also found in a genome-wide search using expression data, which makes it likely to be functional.

Chapter 5

Motif finding with protein-binding microarrays

In the last chapter, we analyzed data generated by chromatin immunoprecipitation. An important advantage of ChIP is that the measured binding recapitulates the *in vivo* binding of a protein. At the same time, the presence of other proteins in the nuclear extract adds noise to the binding signal at the sequence level. Canonical instances of a binding motif are often unbound due to the absence of a required cofactor or due to chromatin states that prevent binding. At the same time, we see indirect binding, where the protein of interest binds to another protein, which in turn binds to DNA, in the absence of a binding site for the former protein. Both of these results are valid biological signals and demonstrate the complex regulatory mechanisms that affect the formation of protein-DNA complexes, but do not strictly reflect the sequence preference of the protein being assayed.

Protein binding microarrays (PBMs) directly measure the *in vitro* DNA binding specificity of a protein without the complication of co-factors and do not require an antibody to the protein of interest. [3]. PBMs were used to measure the binding specificities of 246 candidate yeast TFs [5]. High confidence motifs were obtained for only 89 of these 246 TFs, suggesting that there exists some scope for improvement in the analysis of the data from the PBM microarrays. Therefore, we decided to re-analyze the PBM data from this previous study using INSPECTOR.

Our results show that INSPECTOR is able to find motifs that predict the PBM binding scores for probes better than those reported by SEED-AND-WOBBLE, an algorithm specifically designed to construct motifs from PBM datasets [3].

5.1 Background

5.1.1 Protein binding microarray design

PBMs are an attempt to measure the *in vitro* DNA binding specificity of proteins in an unbiased fashion [3]. The microarrays are designed using deBruijn sequences of order 10 (of total length 4^{10}), which contain every possible DNA 10-mer exactly once. The deBruijn sequence is then computationally segmented into subsequences of length 35, overlapping by 9 bases, yielding a total of approximately 44,000 probe sequences. Each probe sequence thus contains 26 distinct 10-mers, with an overlap of 9 bases between adjacent k-mers. The specific deBruijn sequences were chosen to also contain all possible 10-mers spanning 11 bp with a single gap at any position to attempt maximal coverage of binding sites longer than 10 bp. Since each 10-mer is represented once, all 8-mers are represented 16 times and non-palindromic 8-mers are represented 32 times on the array after accounting for reverse complements.

Each of these probe sequences is attached to a constant 24 nt section, which is end-attached to the substrate. Primer extension using a primer complementary to the constant section, dNTPS and a small quantity of fluorescently labeled dUTP converts the single-stranded oligonucleotides into dsDNA. The dUTP is used to estimate the amount of dsDNA at each probe spot during data normalization.

The protein of interest is expressed with an epitope tag, purified, and applied to the dsDNA microarray. The microarray is incubated with an fluorescent antibody specific to the epitope tag, and the intensity of the fluorescence from the antibody represents the amount of protein bound at each spot on the microarray. Generally, two different microarray designs derived from different deBruijn sequences are used to assay each protein, thereby allowing for correction due to probe effects, either from sequence context, from the position of k-mers within the probe, or due to position of the probe within the microarray design.

5.1.2 Analysis of PBM data using SEED-AND-WOBBLE

SEED-AND-WOBBLE is an algorithm specifically designed for the analysis of PBM datasets [3]. For each k-mer, it designates a foreground set consisting of all probes on the microarray that contain the k-mer, and all other probes form the background set. The probes on the microarray are also ranked according to their normalized intensity score. SEED-AND-WOBBLE computes an enrichment score according to the following equation:

$$\text{Enrichment score} = \frac{1}{B+F}\left[\frac{\rho_B}{B} - \frac{\rho_A}{A}\right], \tag{5.1}$$

where B is the size of the foreground set, F is the size of the background set, ρ_B is the sum of the ranks of the background set, and ρ_A is the sum of the ranks of the foreground set. If two microarray designs were used, the average of the two enrichment scores is used for each k-mer was used. The enrichment score is a discriminative function and therefore SEED-AND-WOBBLE belongs to the discriminative class of motif-finding algorithms.

In order to construct a PWM from enriched k-mers, SEED-AND-WOBBLE first calculates enrichment scores for all 8-mers with 3 gap positions, including cases where the gaps are at the ends of the 8-mer. The highest scoring such 8-mer is designated the *seed.* For each position of the seed 8-mer, the effect on the enrichment score of changing the current base to one of the other three bases is used to calculate the base probabilities at that position. Doing so for all positions of the seed yields the desired PWM.

As all of the data from [5] was analyzed with SEED-AND-WOBBLE and the motifs for 89 TFs published, we used these motifs as a benchmark for INSPECTOR's performance on the PBM datasets.

5.2 Methods

5.2.1 Dataset and initial processing

We downloaded PBM data from the UniPROBE database [9] for the yeast transcription factors assayed in [5]. Though the original study assayed 243 transcription factors and predicted DNA-binding proteins, the published data is for only the 89 proteins for which the authors were able to find a motif that satisfied their criteria.

We used the normalized 60-mer binding scores for motif finding with INSPECTOR. Unlike the 8-mer enrichment scores used by SEED-AND-WOBBLE, the 60-mer scores include the flanking anchor sequence which is common to all the 60-mer probes. We assumed that there could be motif occurrences at the junction of the probe sequence and flanking anchor, and we wanted to ensure that we were scoring that part of the sequence.

For each experiment, we fitted the normalized 60-mer scores to a log-normal distribution. We assessed the fit to this null distribution using Q-Q plots and observed that the fit was good, with deviation only at the tails of the datasets (Figure 5.1). Using the log-normal null distribution, we computed a p-value p for each probe binding score and used $(1 - p)$ as the binding score for analysis with INSPECTOR.

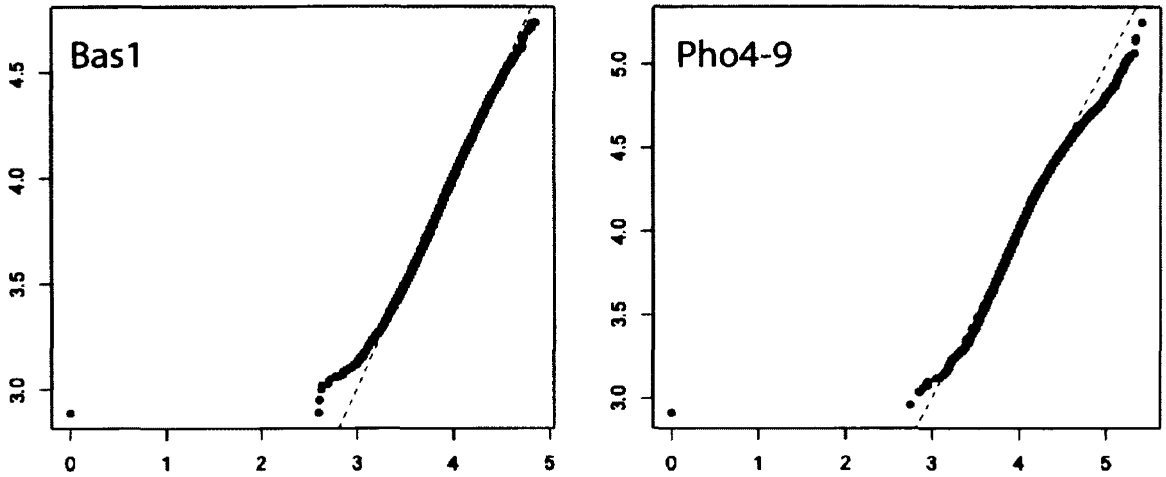


Figure 5.1: **Q-Q plots showing fit of PBM data to log-normal distribution**. Q-Q plots for two experiments, Bas1 and Pho4-9, are shown.

5.2.2 INSPECTOR analysis

For PBM data, we used the 60-mer probe sequences and the $(1 - p)$ score as described above as the inputs to the INSPECTOR algorithm, which was run in "score" mode. As described earlier, the "score" mode uses a dynamic search space and weighted PWMs. We used a zeroeth order Markov model for the background; since the design of the PBM ensures equal coverage of all 10-mers, using a higher order background model would yield no additional information.

5.2.3 Motif evaluation

Probes with a p-value less than 0.1 were designated as the positive set, and probes with a p-value greater than 0.5 were designated as the negative set (other thresholds were tested as well, as described in the Results section). We scanned the positive and negative sets with the motif being evaluated using SCANACE and ranked the sequences according to the highest scoring site in each sequence. We used this ranking to plot a receiver operator characteristic (ROC) curve and used the area under the ROC curve (auROC) as a measure of how well a motif was able to discriminate between the positive and negative sets.

5.3 Results

We ran INSPECTOR using the dynamic search space mode using the set of 60-mer probes and their corresponding normalized binding scores. As a benchmark, we used the motifs reported in the original study as found by SEED-AND-WOBBLE [3, 5]. We evaluated the top motif for each factor as described above, using the auROC as a measure of how well a motif predicted the binding data. We excluded 41 of the 132 experiments where either auROC was less than 0.75, thereby eliminating any experiments where neither algorithm found a sufficiently predictive motif.

As shown in Fig. 5.2, the motifs found by INSPECTOR outperformed those found by SEED-AND-WOBBLE in 76 of the 91 experiments (83%). This performance improvement was consistent regardless of the p-value threshold used to define the positive set, or the set of bound probes. With p-value thresholds of 0.05 and 0.01, motifs found by INSPECTOR were more predictive in 83/104 (80%) and 93/119 (79%) experiments respectively (Fig. 5.3.

Since auROC measures the quality of classification, we checked to see how well it

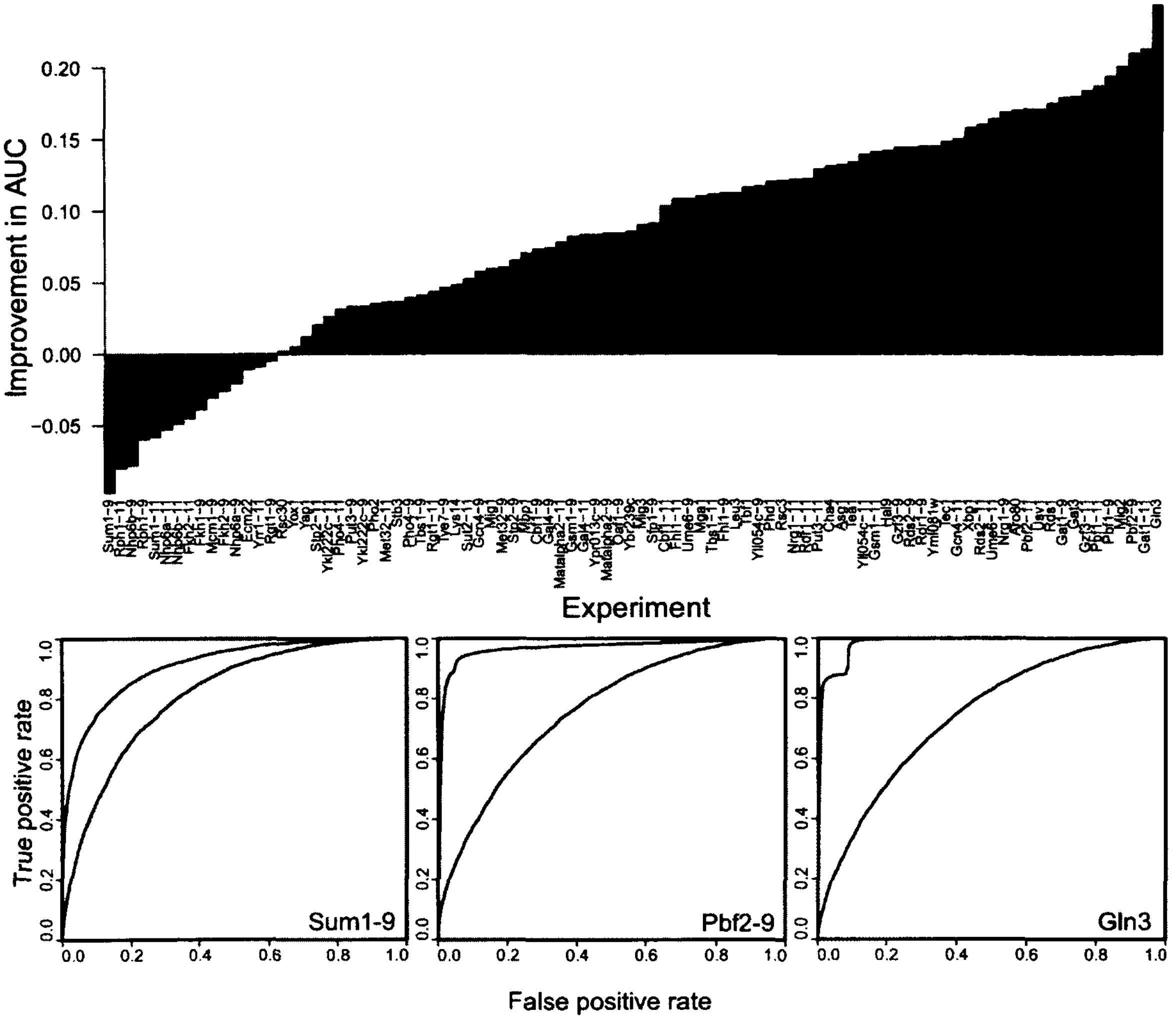


Figure 5.2: **Motifs found by INSPECTOR perform better at retrieval of bound probes than the motifs found by SEED-AND-WOBBLE.** The barchart shows the improvement in the area under the receiver-operator characteristic curve (auROC), and the top motif found by INSPECTOR performs better than the SEED-AND-WOBBLE motif in the majority of cases where either motif has an auROC of 0.75 or better. Three representative ROC curves are shown, two (Gln3 and Pbf2-9) in which INSPECTOR outperforms SEED-AND-WOBBLE and one in which SEED-AND-WOBBLE is better (Sum1-9). The red curve is the ROC for the SEED-AND-WOBBLE motif and the green curve is the ROC for the best INSPECTOR motif.

correlated with the ability of INSPECTOR motifs to predict the binding scores of all probes on the array. We calculated the correlation coefficient between the motif score (score of

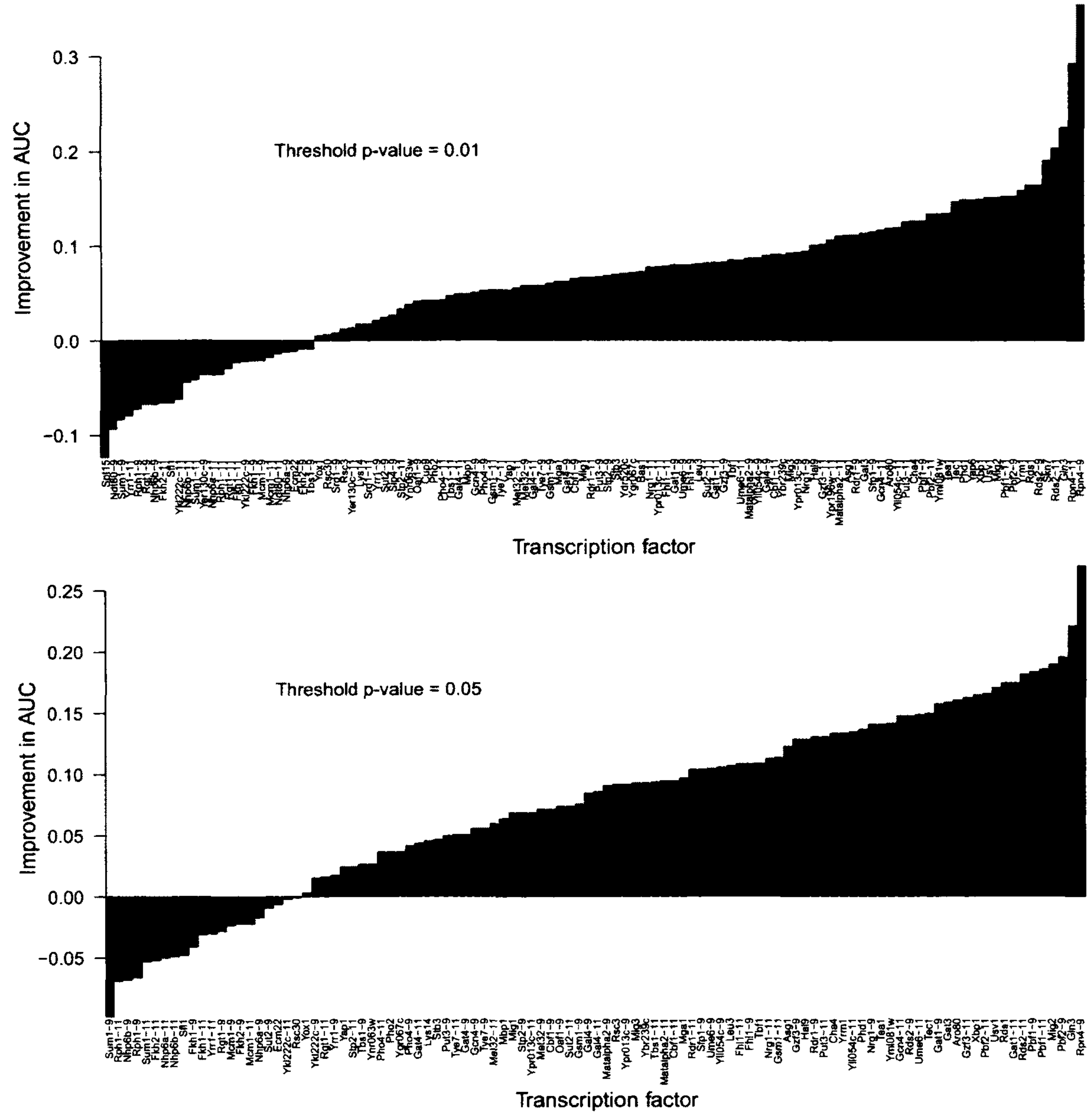


Figure 5.3: **INSPECTOR motifs predict PBM binding data better regardless of p-value threshold** The barcharts shows the improvement in the area under the receiver-operator characteristic (ROC) curve, and the top motif found by INSPECTOR performs better than the SEED-AND-WOBBLE motif regardless of the p-value threshold used to define the positive set of bound probes. The top chart shows the auROC improvement with a threshold of 0.01 and the bottom chart is with threshold 0.05.

the highest scoring site) and the binding score of each probe. We then plotted the auROC against the correlation coefficient; our results show that auROC is a good predictor of the correlation coefficient for each experiment (Fig. 5.4).

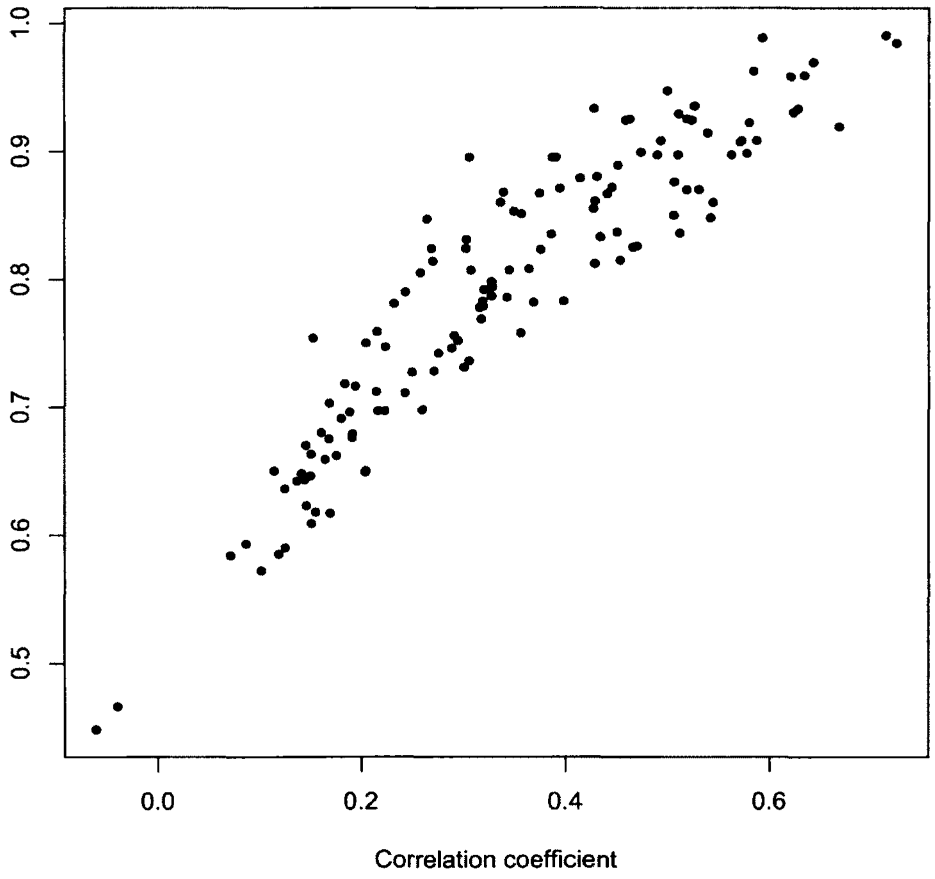


Figure 5.4: **auROC and correlation coefficient are highly correlated.** auROC is highly correlated with the correlation coefficient between motif scores and binding scores ($R^2 = 0.93$). Each point represents an experiment.

5.4 Summary and Discussion

We use INSPECTOR to search for motifs in PBM datasets that had previously been analyzed with SEED-AND-WOBBLE, an algorithm specifically designed for PBM data analysis. In spite of its general nature, INSPECTOR shows marked improvement in finding predictive motifs from PBM datasets relative to the technology-specific SEED-AND-WOBBLE. We postulate that running INSPECTOR on the data for the remaining 157 yeast TFs would yield even more useful information from these experiments.

In addition, PBM has been used to interrogate the DNA binding specificity of several metazoan TFs and the data deposited into UniPROBE [9]. It would be useful to analyze these datasets with INSPECTOR to find predictive motifs for metazoan TFs as well.

Chapter 6

Motif finding using expression data

In earlier chapters, we have identified cis-regulatory elements using experimental techniques that assume prior knowledge of the regulating TF. In the absence of such knowledge, we can use an alternative approach that uses expression data to find gene targets.

Previous studies have attempted to detect regulatory elements by identifying sets of coregulated genes and searching for shared sequence motifs in the upstream regions of these sets of genes [4, 14, 21]. These efforts have involved clustering of genes based on their expression profiles followed by the use of motif-finding algorithms on the upstream regions of each cluster of genes. Given the biological and experimental noise inherent in expression profiles, these methods are limited in their ability to tease apart regulatory programs that results in similar expression profiles, as genes are assigned incorrectly to clusters and lower the sensitivity of the subsequent motif-finding step. INSPECTOR can use dynamic search spaces to perform a genome-wide search for regulatory elements without starting with any predefined coregulatory clusters, thus increasing sensitivity over the two-step approach. INSPECTOR tries to identify genes that have similar expression profiles and a shared sequence motif in their upstream regions, and iteratively refines the model of both the expression profile and the sequence motif. This approach, which we call *dynamic expression clustering*, was used to search for regulatory elements in genome-wide datasets that combined sequence and expression data.

6.1 Background

In the absence of binding data and known TFs, initial efforts to find cis-regulatory elements used similarity of expression profiles to detect regulation by the same TF or set of TFs [21, 37]. The underlying principle was the concept of the *regulon*; genes that were targets of the same TF were likely to show similar patterns of expression (Fig. 6.1).

mRNA expression data was clustered to find sets of genes that were co-regulated. For each gene in a cluster, proximal promoter or sequence upstream of the start codon up to a certain length was extracted from the genome. Motif finding programs were then run on these upstream sequences to find sequence patterns that were over-represented, which were presumptive regulatory elements (Fig. 6.2).

Clustering of expression profiles and subsequent motif finding was successful in finding predictive motifs. The motifs found were used in Bayesian framework to learn combinato-

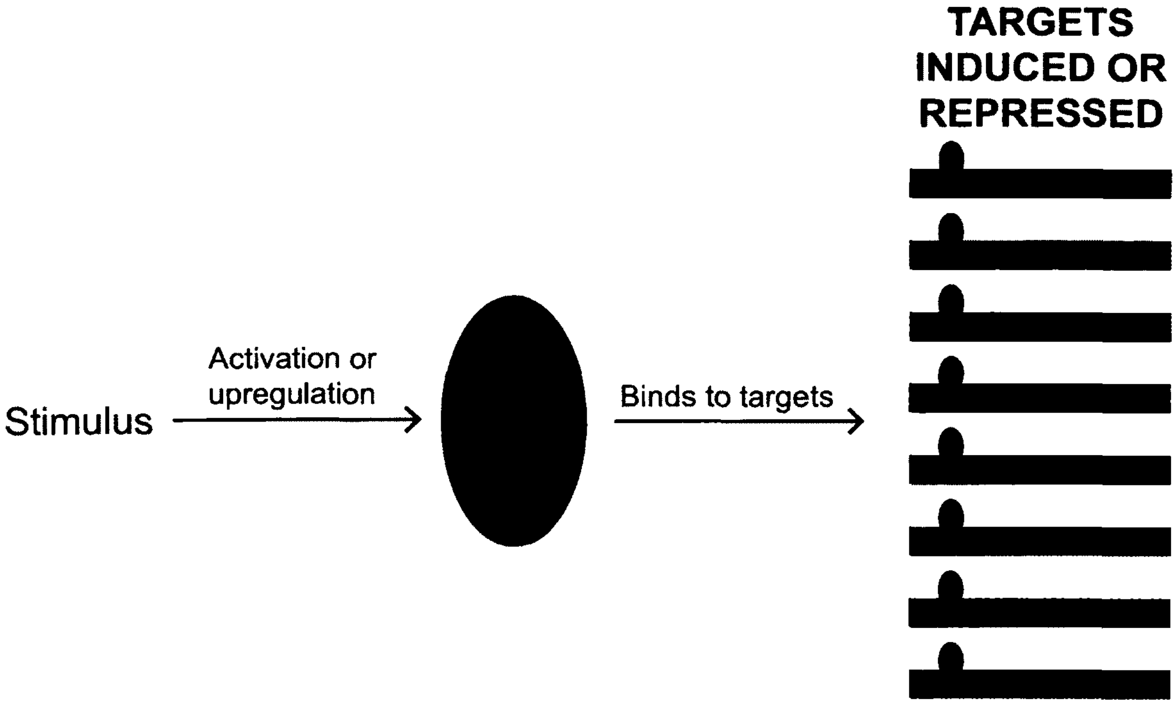


Figure 6.1: **A regulon consists of the gene targets of a single TF or set of TFs.** These gene targets are all induced or repressed by the same TF, and therefore should share a common expression profile.

rial rules that assigned genes to expression clusters using their upstream sequence. The framework of rules assigned 73% of yeast genes to their correct expression clusters [14]. This method was less successful in metazoans: it correctly predicted the expression about 50% of the genes in *C. elegans*, and the fraction was lower in Drosophila and humans.

The greater genomic complexity of higher organisms is responsible for at least part of the loss of predictive power. However, it is also possible that the clustering of genes is confounded by noise in the expression data. For example, if two regulons have overlapping expression profiles, clustering might incorrectly assign some genes in each regulon to the wrong cluster (Fig. 6.3A). Another problematic situation arises when there is a

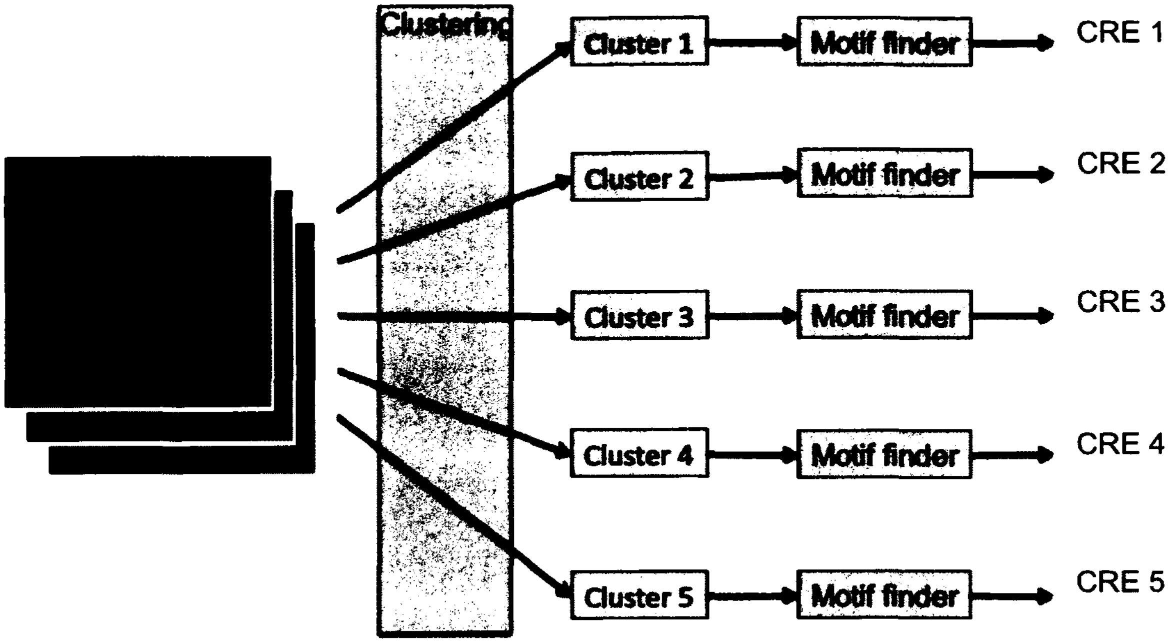


Figure 6.2: **Traditional motif-finding using expression data.** Expression data was clustered to find sets of co-expressed genes. Motif-finding algorithms were then run on the upstream sequences of the genes in each cluster to find cis-regulatory elements (CREs).

large regulon whose broad expression pattern encompasses a smaller tightly regulated set of genes (Fig 6.3B). In both these cases, the motif-finding algorithm will have reduced sensitivity due to incorrectly precomputed clusters. We propose that INSPECTOR's dynamic expression clustering mode, which uses an integrated model for sequence and expression and does not use precomputed clusters, will have better sensitivity by alternating between expression clustering and motif optimization.

Three previously published methods have utilized a combined model for sequence and expression. The first method employed the concept of *module networks* to discover regulatory programs in yeast [38]. It had two drawbacks: it used precomputed clusters

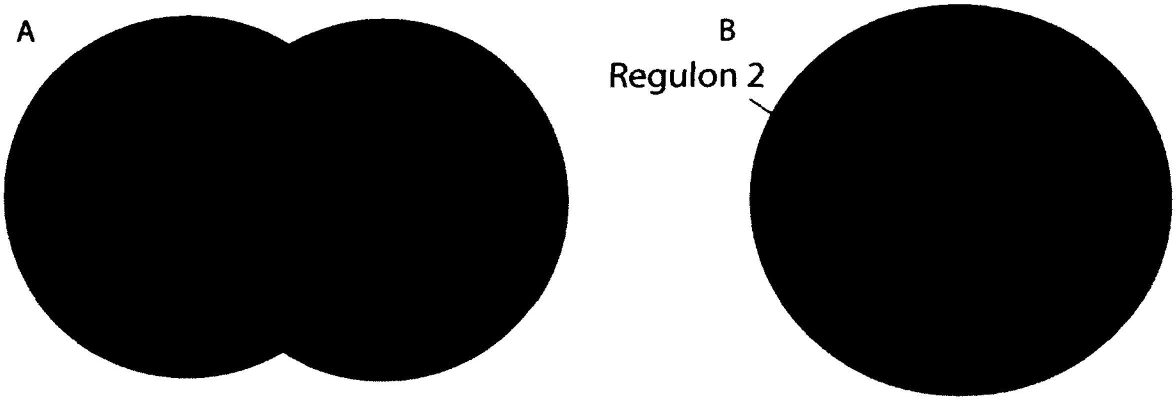


Figure 6.3: **Incorrect clustering of genes in regulons** A: Two regulons have expression profiles that overlap (shaded region). Genes that lie in this expression space could be assigned to the wrong regulon. B: A small tightly regulated regulon has an expression profile that overlaps with a large loosely regulated regulon. Clustering is likely to find only one of these regulons, depending on the parameters used.

of genes and a known set of motifs and it required prior knowledge of the genes that coded for TFs. It also assumed that the expression of a TF would be correlated with that of its gene targets. There are cases where a TF is expressed constitutively, but does not bind to DNA or induce transcription until a further post-translational activation step to the TF [?]. Nuclear hormone receptors such as the estrogen receptor (ER) are another example: these TFs are present in the nucleus or cytoplasm until bound by their ligands, after which they bind to DNA and regulate expression of target genes [32]. In both these situations, correlation between the expression of the TF and its targets will be minimal.

The second method is the MEDUSA algorithm designed to discover sequence-expression programs [39]. MEDUSA has certain unique features: it builds a global regulatory program to explain the expression of all genes and can also identify condition-specific regulation. It successfully found regulatory motifs for the oxygen and heme regulatory networks in yeast [40]. However, it also requires prior knowledge of TFs and co-expression of TFs with their targets. It also needs ChIP-chip data to fine-tune the discovered motifs, which may not be available.

The last method is ALLEGRO, which uses AMADEUS as a motif finder in conjunction with an integrated sequence-expression model [41]. Unfortunately, it is limited to discovering only 20 motifs at a time, which is far fewer than would be expected genome-wide in even a simple eukaryote, like yeast.

We believe that INSPECTOR represents an innovative approach to this problem space, and it does not suffer from the limitations of the existing algorithms.

6.2 Methods

6.2.1 Synthetic sequence-expression data

The sequence dataset consisted of 5000 sequences with lengths sampled from a Gaussian distribution with mean 800 and standard deviation. The sequences were generated using a single nucleotide frequency distribution with the same GC content as yeast intergenic sequences. Each sequence was assigned to one of 80 functional motifs, with varying numbers of sequences assigned to each functional motif to approximate small and large regulatory modules. An instance of the functional motif to which a sequence was assigned was seeded into the sequence. Finally, there were 20 non-functional motifs, and 4 of these were randomly seeded into each sequence.

Yeast has two large groups of genes that are anti-correlated in expression [42]. In conditions of environmental stress, a large set of genes is repressed, including proteins that are apart of the ribosome or involved in RNA processing, nucleotide biosynthesis and secretion. A second set of genes is induced in stress conditions and forms the *environmental stress response*. The expression diversity of yeast genes is therefore less than that of a perfectly random dataset (Fig. 6.4).

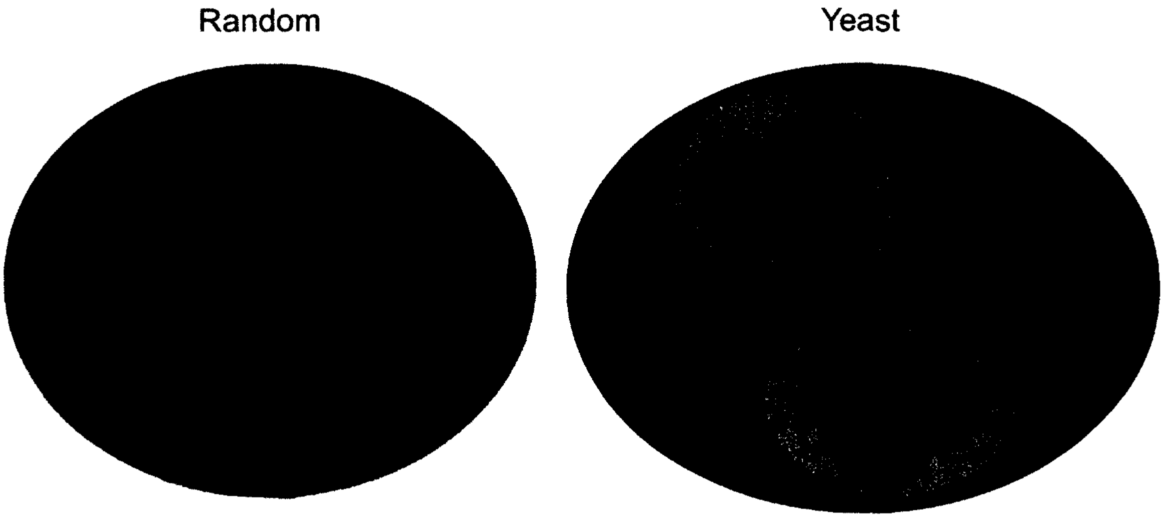


Figure 6.4: **Yeast genes show less expression diversity than a random dataset.** The blue ellipse is the entire expression landscape, and the red circles depict clusters of correlated genes. In a random dataset, the gene clusters are spread throughout the expression landscape. In yeast, there are two main groups of gene clusters that are co-regulated.

In order to mimic the nature of yeast expression data, 40 of the functional motifs were designated as "growth" motifs and 40 were designated as "stress" motifs. Anti-correlated expression patterns were assigned to the "growth" and "stress" sets. Individual motifs were then assigned expression patterns sampled from a Gaussian distribution whose mean was either the "growth" or the "stress" expression pattern, according to whether the motif belonged to the "growth" or the "stress" set respectively. The expression pattern for individual sequences was sampled from a Gaussian distribution whose mean was the expression pattern of the functional motif contained in that sequence. Fig. 6.5 shows the co-expression characteristics of all the genes in the genome and cluster means for real yeast expression data and the synthetic dataset we used.

6.2.2 Yeast sequence-expression dataset

We extracted upstream sequences for all yeast (*Saccharomyces cereviseae*) ORFs as previously described [14]. We also created a combined gene expression dataset from three different yeast studies, which included cell cycle timepoints and various metabolic stimuli, for a total of 5228 genes across 292 conditions [42–44].

6.2.3 Worm sequence expression dataset

We extracted upstream sequences for all genes in *C. elegans* as previously described [14]. We also combined expression data from three different studies in *C. elegans* [45–47] for a total of 82 conditions and 5691 genes.

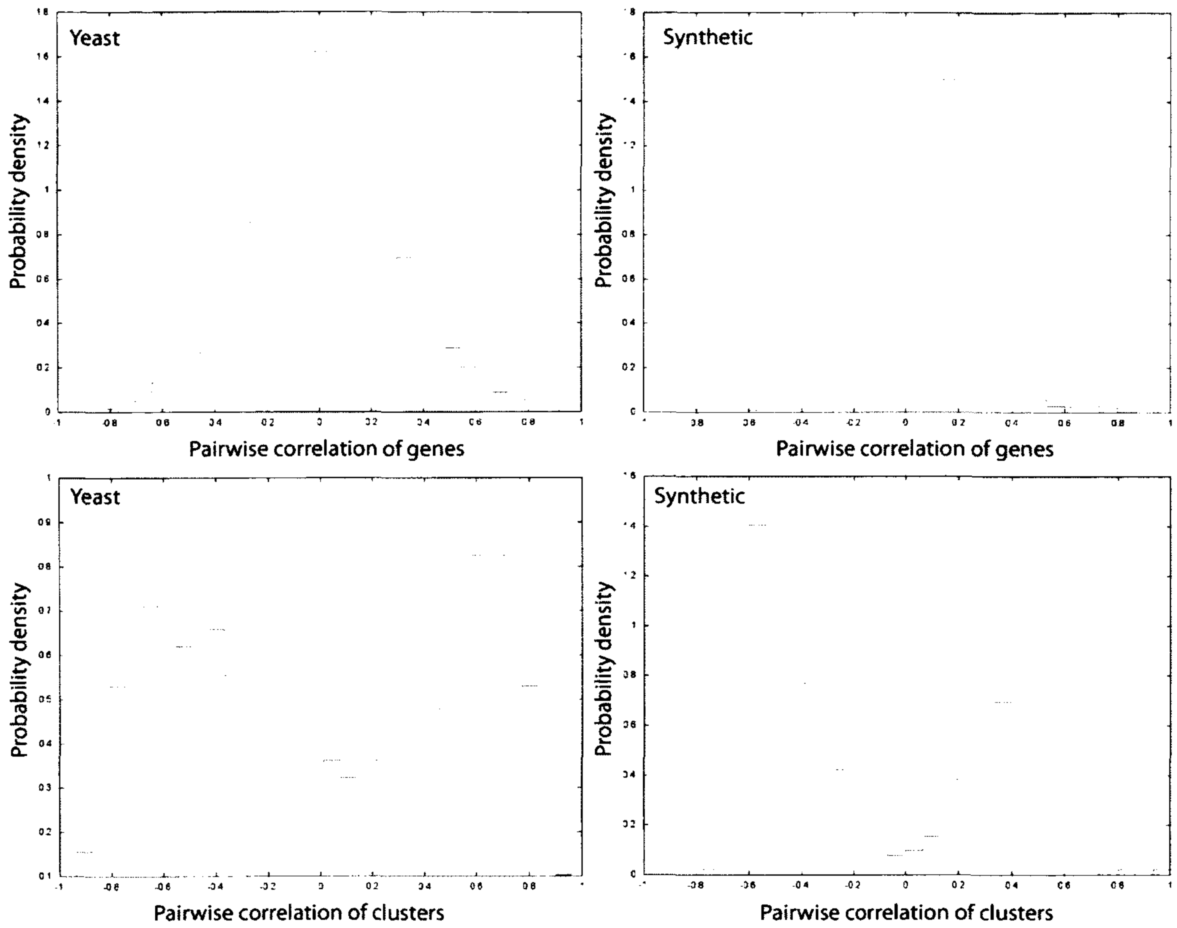


Figure 6.5: **Correlation characteristics of yeast and synthetic expression data:** The left panels show the distribution of pairwise correlation coefficients between the expression profiles of individual genes for the yeast and synthetic datasets. The right panels show the distribution of pairwise correlation coefficients between the mean expression profiles of co-expressed clusters for each dataset.

6.3 Results

6.3.1 Synthetic sequence-expression dataset

To evaluate the performance of this approach on a simulated dataset, we ran INSPECTOR against the synthetic data described in Methods. To compare Inspector to the two-step

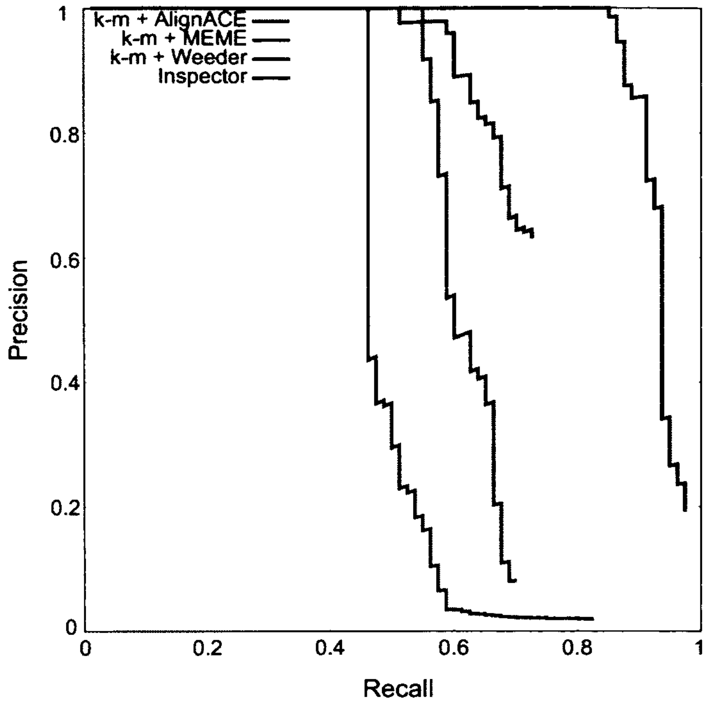


Figure 6.6: **INSPECTOR outperforms k-means clustering and motif-finding using ALIGNACE, MEME and WEEDER.** INSPECTOR recovers seeded motifs from a synthetic sequence-expression dataset better than two-step procedures of k-means clustering and motif-finding using ALIGNACE, MEME and WEEDER as shown by a precision-recall curve.

approach, we also clustered the genes by using k-means clustering on the expression data and ran the motif finding algorithms ALIGNACE, MEME and WEEDER on the upstream sequences of the genes in each cluster. We evaluated the ability of these individual methods to recover seeded motifs from the upstream sequences. INSPECTOR performed much better than the two-step approach, as shown by the precision-recall curve (Fig. 6.6). At a 10% false positive rate (90% precision), INSPECTOR recovered 85% of the seeded motifs, while the two-step algorithms recovered between 46% and 60%.

6.3.2 Yeast sequence-expression data

We created a combined gene expression dataset from three different yeast studies, which included cell cycle timepoints and various metabolic stimuli, for a total of 5228 genes across 292 conditions [42–44]. We compared the performance of INSPECTOR to that of a two-step process of clustering using k-means and motif-finding within clusters using AlignACE (km-aa).

The list of motifs generated by each algorithm was compared to a compendium of 97 known yeast motifs using the COMPAREACE program [4]. INSPECTOR found more known motifs (65/97 or 67%) than km-aa (39/97 or 40%) at the same COMPAREACE threshold (Fig. 6.7).

We created lists of gene target for the predicted motifs for each method. We compared these gene lists against the lists of target genes found by ChIP-chip [13], using Fisher's exact test. At a p-value threshold of 10^{-7}, 40 gene lists produced by INSPECTOR overlapped significantly with gene lists from the ChIP-chip data, while only 30 target lists produced by km-aa showed significant overlap.

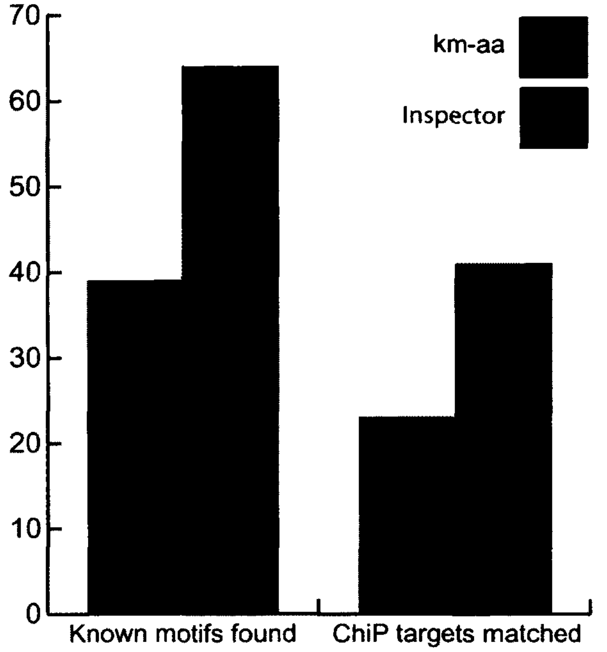


Figure 6.7: **INSPECTOR detects more known yeast motifs than the combination of k-means clustering and ALIGNACE (km-aa).** There were 97 known motifs in total. A COMPAREACE score of 0.75 or greater was considered a match. ChIP target sets [13] were considered a match if the hypergeometric p-value for overlap was less than 10^{-7}.

6.3.3 Worm sequence-expression data

We next used dynamic expression clustering to find motifs using Inspector on *C. elegans* expression data. To determine a significance threshold for reporting motifs, we repeated the search on randomized sequences as a negative control and observed the distribution of specificity scores for motifs found in the randomized dataset (Fig. 6.8). At a specificity score of 26 or higher, we found 135 motifs found in the real dataset and only 10 motifs in the randomized dataset, which translates to a false discovery rate of 7.4%.

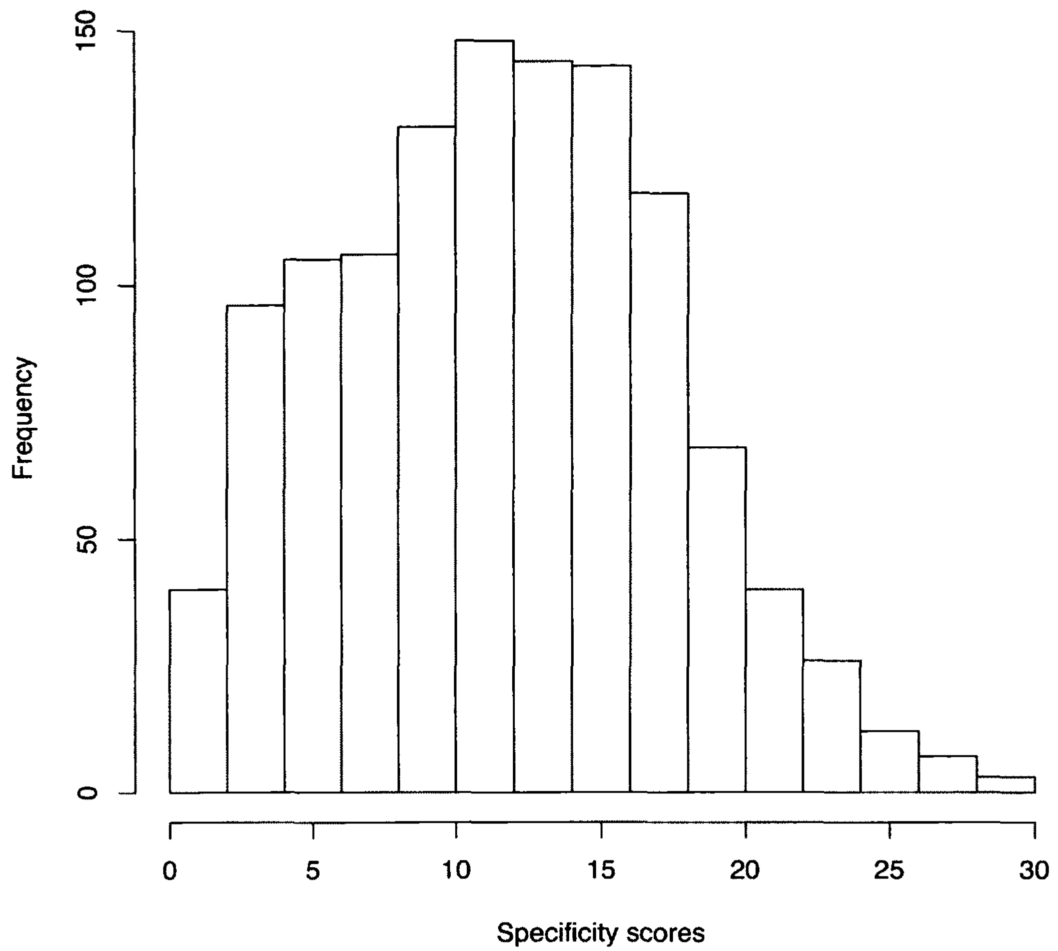


Figure 6.8: **Null distribution of specificity scores for** ***C. elegans*** **data.** INSPECTOR was run on a randomized *C. elegans* sequence-expression dataset. A histogram of the specificity scores of the top 10000 motifs found is shown.

We took the top 100 motifs identified by INSPECTOR and removed motifs that had significant overlap in the list of their target genes with a higher scoring motif, leaving 38 motifs in total. For each of the motifs, we used the list of target genes to perform enrichment analysis for Gene Ontology (GO) and Anatomy Ontology (AO) terms from Wormbase [48]. The top 20 motifs from this analysis are shown in Fig. 6.9 and Fig. 6.10, along with any enriched GO or AO terms.

Motif M1 is the known GATA factor binding site. As expected, GO terms such as

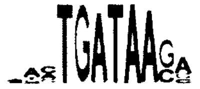
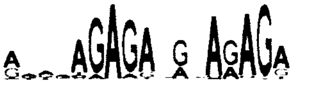
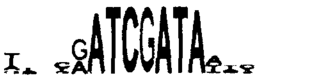
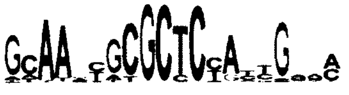
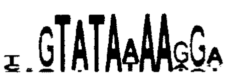
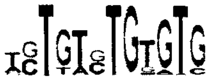
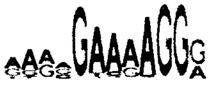
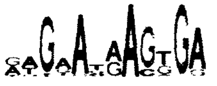
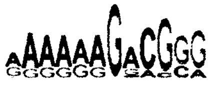
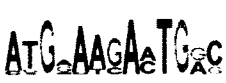
Motif	Logo	Specificity score	Enriched GO term	Enriched AO term
M1		83	hydrolase activity	digestive tract
M2		81	locomotion	muscle cell
M3		62	ubiquitin-dependent protein catabolic process	
M4		59	sex differentiation	germline
M5		53	structural constituent of cuticle	cuticle
M6		50	locomotion	
M7		50	locomotion	somatic nervous system
M8		40	molting cycle	
M9		39	locomotion	neuron
M10		39	protein catabolic process	intestine

Figure 6.9: **The top 10 motifs found by INSPECTOR in a genome-wide search of a *C. elegans* sequence and expression dataset.** Alongside each motif is its specificity score and any Gene Ontology (GO) and Anatomy Ontology (AO) terms that were enriched in the list of target genes.

“small molecule metabolic process” and “hydrolase activity” are highly enriched in the set of target genes, while the AO terms “digestive tract” and “intestine” are also enriched.

Motif M2 was previously found to be associated with the expression of muscle genes [49], and "locomotion" and "muscle cell" are the ontology terms most enriched in target genes. We note that several similar GA-rich motifs also have very high specificity scores and highly overlapping target gene sets, suggesting that the motif may be more degenerate than the highest scoring motif would suggest.

Motif M3 is novel and GO terms associated with ubiquitin-mediated proteolysis are significantly enriched among the target genes. The target genes include several F-box family proteins and non-coding RNAs.

Motif M4 matches a motif identified as the binding site for CEH-30 from modENCODE ChIP-seq data, while our own analysis of the LIN-15B ChIP-seq dataset from the modENCODE project identifies it as well (Fig. 4.7). CEH-30 ensures survival of male-specific neurons during development [50]. The GO terms "anatomical structure development" and "sex differentiation" are enriched in the target genes identified by Inspector, supporting the hypothesis that CEH-30 binds to this motif. LIN-15B is implicated in the development of vulval cells [51]. In either case, though expression data analysis may not identify the factor that actually binds this motif, it is likely to be functional.

Motif M5 is another novel motif. The GO term "structural constituent of cuticle" and the AO term "cuticle" are enriched in the target genes. It is therefore likely that motif 5 regulates genes involved in cuticle production, and the target list includes several genes involved in collagen (*col*) and lipid production (*vit*).

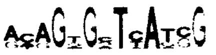
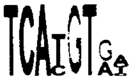
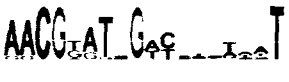
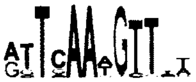
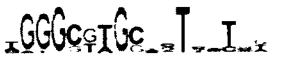
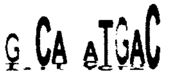
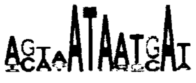
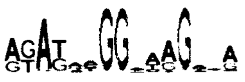
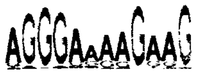
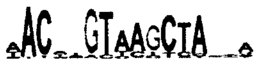
Motif	Logo	Specificity score	Enriched GO term	Enriched AO term
M11		37		
M12		36	cation binding	somatic nervous system
M13		35		
M14		34	locomotion	rectum
M15		34	plasma membrane	ventral nerve cord
M16		33	signaling	tail neuron
M17		33	peptidase activity	intestine
M18		33	structural constituent of cuticle	
M19		32	anatomical structure development	
M20		32		lumbar neurons (PHC)

Figure 6.10: **Motifs 11 through 20 found by INSPECTOR in a genome-wide search of a *C. elegans* sequence and expression dataset.** Alongside each motif is its specificity score and any Gene Ontology (GO) and Anatomy Ontology (AO) terms that were enriched in the list of target genes.

6.4 Summary and Discussion

We used INSPECTOR to search for cis-regulatory elements in a genome-wide fashion using expression data and proximal promoter sequence. We first validated the applicability of the INSPECTOR algorithm to this problem using a synthetic sequence-expression dataset, and showed that INSPECTOR recovered more predictive motifs than using clustering in combination with existing motif-finding algorithms. We then did a similar analysis with real yeast data and found that INSPECTOR identified known cis-regulatory elements in yeast with greater sensitivity than clustering followed by AlignACE. Finally, we used INSPECTOR to analyze sequence-expression data from *C. elegans* and found both known and novel cis-regulatory elements. These novel elements have very significant specificity scores and are candidates for experimental validation.

Chapter 7

Conservation of cis-regulatory elements in the nematode lineage

7.1 Introduction and background

Like coding regions, regulatory sequences in the genome show significant conservation. However, the level of conservation seen is generally lower than that in coding regions. Still, cross-species comparisons have been used successfully to identify cis-regulatory modules in a wide variety of eukaryotes [52, 53].

While there have been several approaches that search for regulatory motifs using sequence conservation, these methods rely on an over-representation metric to find motifs. As a result, while they are generally safe from detecting unconserved non-specific motifs, they often find conserved non-specific motifs that reflect either signals to general transcriptional machinery or broad regulatory programs. Therefore, we wanted to see if optimizing discriminative power while using sequence conservation would yield any novel regulatory motifs.

For our analysis, we chose the four nematode species, *Caenorhabditis elegans, Caenorhabditis brenneri, Caenorhabditis briggsae*, and *Caenorhabditis remanei*. The genomic sequences for all these species are available. While *C. elegans* was the first metazoan genome published and is the best annotated of the four, basic analysis has mapped orthologs of *C. elegans* genes in each of the other three species. This knowledge of homology allows us to design datasets that we can use with INSPECTOR in two ways.

First, while there is limited gene expression profile data for *C. briggsae*, *C. brenneri* and *C. remanei*, we have extensive expression datasets available for *C. elegans*. We can therefore look for sets of genes that are co-regulated in the *C. elegans* context, and search for discriminative motifs in sets of their orthologs in the other species.

Second, we can utilize functional annotation available for *C. elegans* to build groups of genes that are involved in a specific biological process or cellular function. Groups of orthologs of these genes can then be used to look for regulatory motifs that control the same process or function in sister species. We have done this analysis for genes that are associated with the ribosome and proteasome, two groups of genes that are highly conserved not only within the nematodes, but also across a wide diversity of species.

Our hypothesis is that—given the relatively recent evolutionary divergence of these species from each other—we would expect to find similar regulatory motifs in each species.

Any deviations from this hypothesis, especially where a single species seems to have lost or gained a regulatory motif, would imply that a change in the regulatory network has occurred within the nematode lineage.

7.2 Methods

7.2.1 Nematode sequence expression datasets

We created expression datasets for *C. briggsae*, *C. remanei* and *C. brenneri* by assigning expression profiles of *C. elegans* genes (see Chap. 6) to their orthologous genes (if present and known) in each species. We combined this expression dataset with the original *C. elegans* expression dataset. Up to 2000 base pairs of upstream sequence was extracted for genes from *C. briggsae*, *C. remanei* and *C. brenneri*. Along with the original *C. elegans* dataset, we created a combined sequence expression dataset for the four nematode species.

7.2.2 Ribosome genes

We downloaded a list of ribosomal genes in C. elegans from RPG [54]. We queried Wormbase [48] to find the orthologs for these genes in *C. briggsae*, *C. remanei*, and *C. brenneri*. We ran INSPECTOR in subset mode (fixed search space mode) with the set of ribosomal genes as the search space, and the set of all other genes as background. We did this both individually in each species as well as with a combined dataset for all four nematode species.

7.2.3 Proteasome genes

We used Wormbase to query for all *C. elegans* genes that were annotated with the Gene Ontology (GO) term “proteasome” or any of its children. We also queried Wormbase to find the *C. briggsae*, *C. remanei*, and *C. brenneri* orthologs for these genes. We ran INSPECTOR in fixed search space mode with the set of proteasomal genes as the search space, and the set of all other genes as background. We did this both individually in each species as well as with a combined dataset for all four nematode species.

7.3 Results

7.4 Sequence expression datasets

We ran INSPECTOR on the combined dataset for all four species, with orthologous genes sharing the expression profile of the *C. elegans* ortholog. In this analysis, sequence patterns that score well would be discriminative as well as conserved.

Our results show that several of the motifs recaptiulate those found when searching in *C. elegans* alone (Figs. 6.9, 6.10). However, we do find three new high-scoring motifs (Fig. 7.1). We analyzed the list of gene targets of these motifs as predicted by INSPECTOR for over-representation of GO terms to attempt functional annotation. We also compared these motifs with known motifs in JASPAR with the Tomtom motif comparison program, which includes JASPAR as a query target [55].

The first motif is GGGTCTCGCCACGA/TCGTGGCGAGACCC. The GO term "ribosome" is over-represented in the list of gene targets. This motif is also found when searching for motifs in the promoters of ribosomal genes (see below). The second motif, TTAAAGGNRCAT/ATGYNCCTTTAA, appears to be associated with meiosis and gamete generation, though the p-value is much higher than for the first motif. Both these motifs do not match any known motifs in JASPAR. The third motif, CGTAAATCBACA/TGTVGATTTACG, weakly matches the motif for the yeast TF YPR015C CGTAAATCCT. No sufficiently specific GO term was significantly over-represented in its target genes.

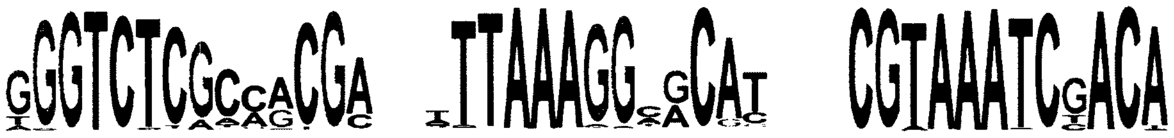


Figure 7.1: **New nematode motifs from phylogenetic search with INSPECTOR** These motifs were found when searching a combined sequence expression dataset from 4 nematode species, but not when searching in *C. elegans* alone.

7.5 Proteasome promoters

We first ran INSPECTOR on the promoters of the proteasomal genes in each individual species. The top 5 motifs found by INSPECTOR in each species are shown in Fig. 7.2. The motif GATGACAAT was found in all four species, ranking first in *C. elegans* and

C. remanei and second in *C. briggsae* and *C. brenneri*. Even in the two species where the GATGACAAT motif had the second highest specificity score, the scores of the first two motifs were essentially identical (21.10 and 20.71 in *C. briggsae*, 18.58 and 18.52 in *C. brenneri*). Given its high specificity to proteasomal promoters and high degree of conservation, we propose that this motif is a novel regulatory element in nematode proteasomal promoters.

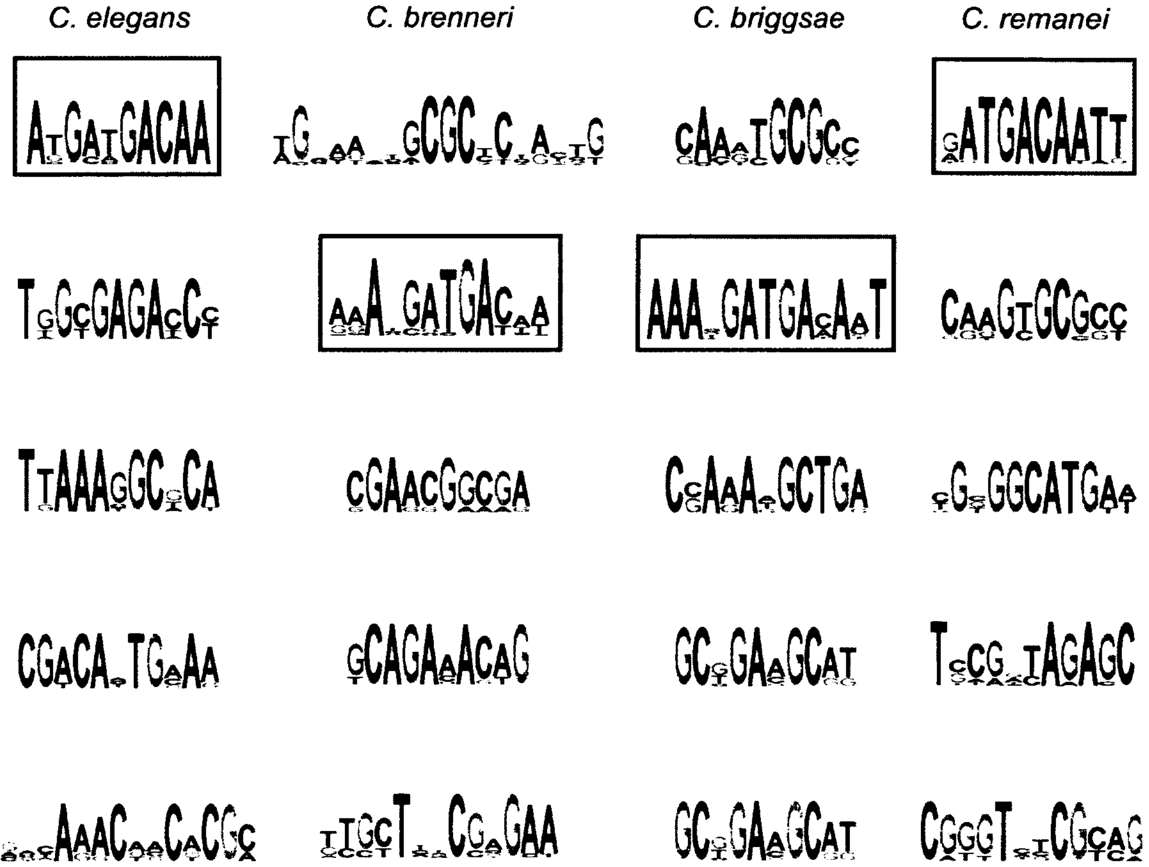


Figure 7.2: **Proteasomal motifs in individual nematode species.** The top 5 motifs reported by INSPECTOR in the ribosomal promoters of each species are shown. The GATGACAAT motif is boxed.

INSPECTOR was also run on the combined proteasomal dataset. In this case, GATGACAAT is the top-ranked motif, with a specificity score of 84.2 (Fig 7.3). The next highest ranked motif scores 30.2. Our results with the combined dataset reinforce the idea that this element is likely to be functional.

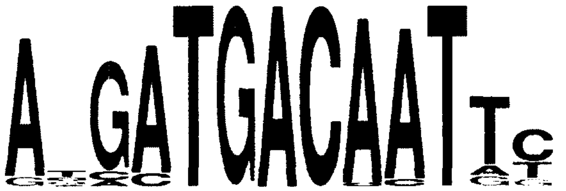


Figure 7.3: **Proteasomal motif from combined nematode dataset**

7.6 Ribosome promoters

We searched for motifs in the set of upstream sequences of ribosomal genes in each nematode species. The top 5 motifs found by INSPECTOR in each species are shown in Fig. 7.4.

INSPECTOR consistently reports the GGGTCTCG/CGAGACCC motif in *C. briggsae*, *C. brenneri*, and *C. remanei* as most specific to ribosomal promoters. However, this motif is less specific to the ribosomal promoters in *C. elegans*, ranking third. This element was previously found to be over-represented in these promoters; it was also shown to be functional *in vivo* in a reporter construct. One ongoing hypothesis in our lab is that in *C. elegans*, this element has been incorporated into a transposable element and is therefore present elsewhere in the genome [56]. Supporting this theory, the GGGTCTCG/CGAGACCC octamer occurs 4583 times in the *C. elegans* genome, deviating significantly (p-value less than 10^{-300} from the expected frequency according to a GC-content model [57].

Another motif GGCCTAA/TTTGGCC is found in all four species. It is the second most discriminative motif found in *C. briggsae* and *C. remanei*, the third most discriminative in *C. brenneri* and the fifth most discriminative in *C. elegans*. A third motif TTTCAGGTAA/TTACCTGAAA is found in all four species, though it is only the 14th ranked motif in *C. brenneri*.

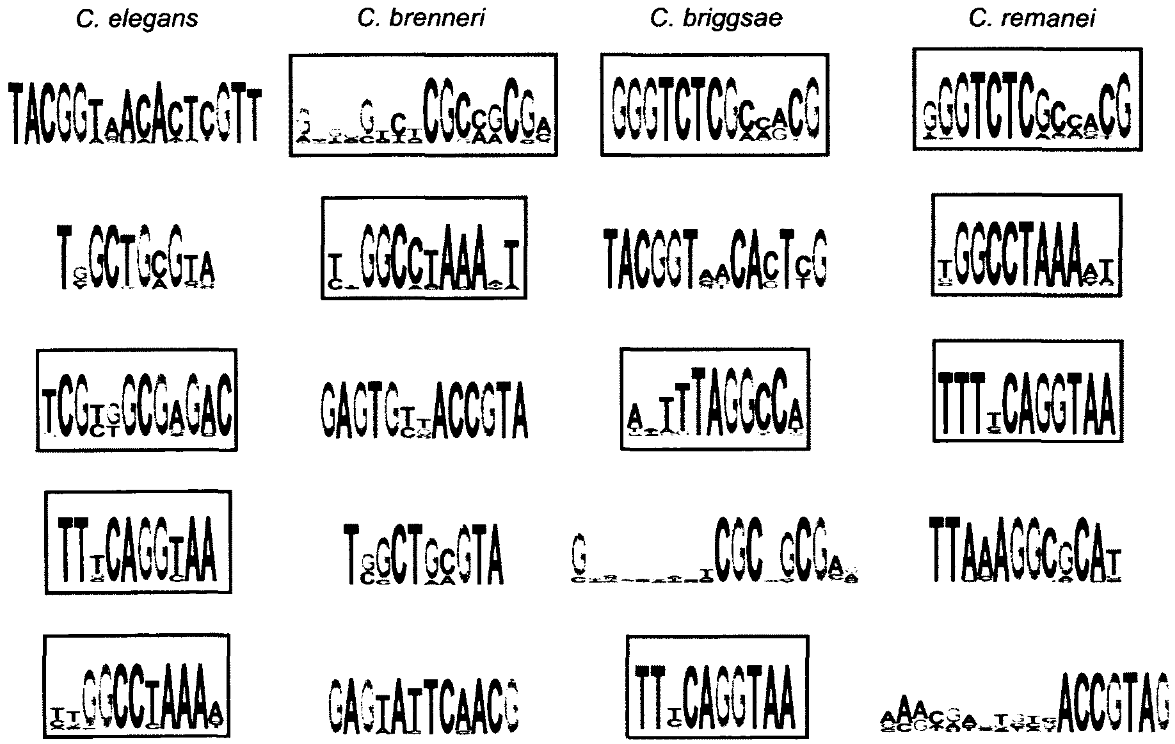


Figure 7.4: **Ribosomal motifs in individual nematode species.** The top 5 motifs reported by INSPECTOR in the ribosomal promoters of each species are shown. At least three motifs, GGGTCTCG/CGAGACCC (brown boxes), GGCCTAA/TTTGGCC (green boxes), and TTTCAGGTAA/TTACCTGAAA (blue boxes), are found in all four species.

We also ran INSPECTOR on the combined ribosomal dataset for the four species. The top 4 motifs found are shown in Fig. 7.5. In this case, the top-ranked motif is once again GGGTCTCG/CGAGACC. The motif ranked second is ACGGTnnCACTCG, which was ranked first in *C. elegans* and second in *C. briggsae* in the individual species search.

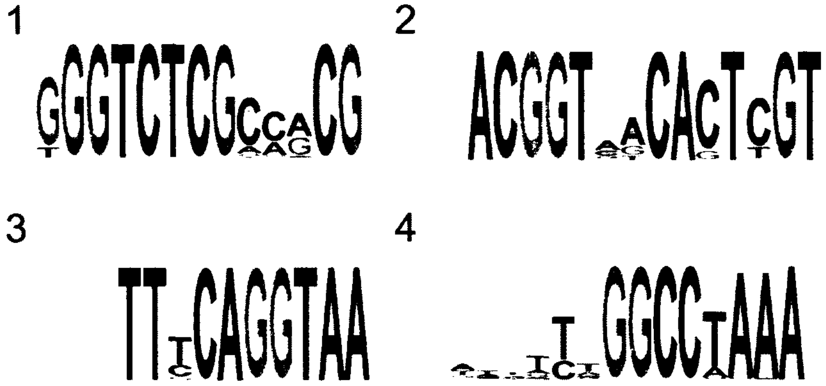


Figure 7.5: **Ribosomal motifs in a combined nematode dataset** The top 4 motifs reported by INSPECTOR in the ribosomal promoters of the four nematode species.

7.7 Summary and Discussion

Our hypothesis was that optimizing for specificity in combination with sequence conservation would detect novel motifs. We used INSPECTOR to search a combined sequence-expression dataset for four nematode species, *C. elegans*, *C. briggsae*, *C. brenneri*, and *C. remanei*. This analysis yielded several high-scoring motifs that were not reported in the individual species search in *C. elegans*, showing the power of using evolutionary conservation to find regulatory elements.

We also investigated two specific groups of promoters in the four nematode species. In proteasomal promoters, we found a novel sequence element that is highly conserved across all four species. This element is a promising candidate for experimental validation.

In ribosomal promoters, the most discriminative elements differed between species. Given the high degree of conservation of the proteins themselves, the differences seen once again demonstrate that regulatory programs evolve at a faster rate than coding sequence. Furthermore, it is possible that chance genomic events such as transposition can lead to rapid changes in the regulatory programs of even closely related species.

We note here that none of the motifs found either for the proteasomal promoters or in the ribosomal promoters matched existing elements in JASPAR. One obvious possibility is that we do not have information about similar elements in other metazoans in JASPAR. Given the lack of conservation of regulatory elements between yeast and nematodes, it is also probable that despite the high level of conservation seen in the gene products, the cis-regulatory elements controlling their expression are much less conserved.

Chapter 8

Discussion

In this thesis, we presented INSPECTOR, a novel motif-finding algorithm. We applied INSPECTOR to different kinds of datasets, and our results show how each of the innovative aspects of INSPECTOR results in the detection of more predictive or discriminative motifs than existing algorithms.

First, we analyzed ChIP-seq data from multiple species with INSPECTOR in fixed search space mode and compared its performance to the discriminative algorithms DREME and AMADEUS. All of these algorithms use similar objective functions, but have different models for the sequence motif being optimized. INSPECTOR uses a full PWM model, while DREME uses regular expressions and AMADEUS uses k-mers that are later combined into PWMs for final optimization. INSPECTOR finds more discriminative motifs than the other two algorithms—especially for TFs with long binding sites—and we believe we can attribute this improved performance to the use of the full PWM model.

We next tested another innovative aspect of INSPECTOR, the dynamic search space. We applied INSPECTOR to extract motifs from PBM data and compared its performance to that of Seed-and-Wobble, an algorithm designed specifically to analyze PBM datasets. We showed that INSPECTOR found motifs that predict the PBM binding data better than those found by Seed-and-Wobble.

Our third application of INSPECTOR was to conduct a genome-wide search for regulatory elements using expression data. We showed that INSPECTOR outperformed a two-step process of clustering genes by their expression profiles and subsequently searching their promoters for over-represented sequence elements, both with a synthetic dataset and with real yeast data. We also analyzed a sequence-expression dataset from *C. elegans* and found several novel putative regulatory elements.

Finally, we used INSPECTOR to investigate the conservation of regulatory elements in nematodes. INSPECTOR analysis of a sequence dataset of 4 nematode species combined with expression data from *C. elegans* uncovered regulatory elements that were not found in the analysis of the *C. elegans* dataset alone. We also searched for regulatory elements in proteasomal and ribsomal genes, and found highly significant putative elements.

Generally speaking, any single motif-finding algorithm is easily outperformed by a meta-algorithm that combines the results from multiple motif finding algorithms [?]. We believe that INSPECTOR should be part of any such ensemble, given its success as a discriminative motif finder in the context of a variety of different experimental datasets.

8.1 Future directions

We plan two main directions in which the research from this thesis will be advanced:

1. **Experimental validation of putative regulatory elements**

   Our analysis of nematode datasets with INSPECTOR detected several putative regulatory elements. We would like to validate these computational predictions *in vivo* using a reporter construct. If these elements are found to be functional, we would also like to characterize the functional significance of these elements by comparing the expression patterns seen with the reporter construct against a database of known expression patterns in *C. elegans*.

2. **Ensemble learning of sequence motifs**

   Several datasets are being generated where we have a set of genomic regions whose defining characteristic is the downstream result of multiple protein-DNA binding events. For example, ChIP-seq datasets exist for the transcriptional co-activator p300, which is associated with enhancers [24]. p300 can be recruited to enhancers by more than one TF, which means that searching for a single discriminative sequence motif is unsuccessful and an ensemble of sequence features does much better [36]. There is therefore a need for an algorithm that looks for a set of sequence patterns that can *together* discriminate between a positive and a negative set. Such an algorithm might attempt to simultaneously learn a group of motifs while optimizing some Bayesian objective function.

Bibliography

[1] Y. Blat and N. Kleckner, "Cohesins bind to preferential sites along yeast chromosome iii, with differential regulation along arms versus the centric region," *Cell*, vol. 98, no. 2, pp. 249–59, Jul 1999.

[2] D. S. Johnson, A. Mortazavi, R. M. Myers, and B. Wold, "Genome-wide mapping of in vivo protein-DNA interactions," *Science*, vol. 316, no. 5830, pp. 1497–502, Jun 2007.

[3] M. F. Berger, A. A. Philippakis, A. M. Qureshi, F. S. He, P. W. Estep, and M. L. Bulyk, "Compact, universal DNA microarrays to comprehensively determine transcription-factor binding site specificities," *Nature Biotechnology*, vol. 24, no. 11, pp. 1429–35, 2006.

[4] J. D. Hughes, P. W. Estep, S. Tavazoie, and G. M. Church, "Computational identification of cis-regulatory elements associated with groups of functionally related genes in *Saccharomyces cerevisiae*," *Journal of Molecular Biology*, vol. 296, no. 5, pp. 1205–1214, 2000.

[5] C. Zhu, K. J. Byers, R. P. McCord, Z. Shi, M. F. Berger, D. E. Newburger, K. Saulrieta, Z. Smith, M. V. Shah, M. Radhakrishnan, A. A. Philippakis, Y. Hu, F. De Masi, M. Pacek, A. Rolfs, T. Murthy, J. Labaer, and M. L. Bulyk, "High-resolution DNA-binding specificity analysis of yeast transcription factors," *Genome Res*, vol. 19, no. 4, pp. 556–66, 2009.

[6] M. B. Gerstein, Z. J. Lu, E. L. Van Nostrand, C. Cheng, B. I. Arshinoff, T. Liu, K. Y. Yip, R. Robilotto, A. Rechtsteiner, K. Ikegami, P. Alves, A. Chateigner, M. Perry, M. Morris, R. K. Auerbach, X. Feng, J. Leng, A. Vielle, W. Niu, K. Rhrissorrakrai, A. Agarwal, R. P. Alexander, G. Barber, C. M. Brdlik, J. Brennan, J. J. Brouillet, A. Carr, M. S. Cheung, H. Clawson, S. Contrino, L. O. Dannenberg, A. F. Dernburg, A. Desai, L. Dick, A. C. Dose, J. Du, T. Egelhofer, S. Ercan, G. Euskirchen, B. Ewing, E. A. Feingold, R. Gassmann, P. J. Good, P. Green, F. Gullier, M. Gutwein, M. S. Guyer, L. Habegger, T. Han, J. G. Henikoff, S. R. Henz, A. Hinrichs, H. Holster, T. Hyman, A. L. Iniguez, J. Janette, M. Jensen, M. Kato, W. J. Kent, E. Kephart, V. Khivansara, E. Khurana, J. K. Kim, P. Kolasinska-Zwierz, E. C. Lai, I. Latorre, A. Leahey, S. Lewis, P. Lloyd, L. Lochovsky, R. F. Lowdon, Y. Lubling, R. Lyne, M. MacCoss, S. D. Mackowiak, M. Mangone, S. McKay, D. Mecenas, G. Merrihew, r. Miller, D. M., A. Muroyama, J. I. Murray, S. L. Ooi, H. Pham, T. Phippen, E. A. Preston, N. Rajewsky, G. Ratsch, H. Rosenbaum, J. Rozowsky, K. Rutherford, P. Ruzanov, M. Sarov, R. Sasidharan, A. Sboner, P. Scheid, E. Segal, H. Shin, C. Shou, F. J. Slack *et al.*, "Integrative analysis of the *Caenorhabditis elegans* genome by the

modENCODE project," *Science*, vol. 330, no. 6012, pp. 1775–87, 2010, gerstein, Mark B Lu, Zhi John Van Nostrand, Eric L Cheng, Chao Arshinoff, Bradley I Liu, Tao Yip, Kevin Y Robilotto, Rebecca Rechtsteiner, Andreas Ikegami, Kohta Alves, Pedro Chateigner, Aurelien Perry, Marc Morris, Mitzi Auerbach, Raymond K Feng, Xin Leng, Jing Vielle, Anne Niu, Wei Rhrissorrakrai, Kahn Agarwal, Ashish Alexander, Roger P Barber, Galt Brdlik, Cathleen M Brennan, Jennifer Brouillet, Jeremy Jean Carr, Adrian Cheung, Ming-Sin Clawson, Hiram Contrino, Sergio Dannenberg, Luke O Dernburg, Abby F Desai, Arshad Dick, Lindsay Dose, Andrea C Du, Jiang Egelhofer, Thea Ercan, Sevinc Euskirchen, Ghia Ewing, Brent Feingold, Elise A Gassmann, Reto Good, Peter J Green, Phil Gullier, Francois Gutwein, Michelle Guyer, Mark S Habegger, Lukas Han, Ting Henikoff, Jorja G Henz, Stefan R Hinrichs, Angie Holster, Heather Hyman, Tony Iniguez, A Leo Janette, Judith Jensen, Morten Kato, Masaomi Kent, W James Kephart, Ellen Khivansara, Vishal Khurana, Ekta Kim, John K Kolasinska-Zwierz, Paulina Lai, Eric C Latorre, Isabel Leahey, Amber Lewis, Suzanna Lloyd, Paul Lochovsky, Lucas Lowdon, Rebecca F Lubling, Yaniv Lyne, Rachel MacCoss, Michael Mackowiak, Sebastian D Mangone, Marco McKay, Sheldon Mecenas, Desirea Merrihew, Gennifer Miller, David M 3rd Muroyama, Andrew Murray, John I Ooi, Siew-Loon Pham, Hoang Phippen, Taryn Preston, Elicia A Rajewsky, Nikolaus Ratsch, Gunnar Rosenbaum, Heidi Rozowsky, Joel Rutherford, Kim Ruzanov, Peter Sarov, Mihail Sasidharan, Rajkumar Sboner, Andrea Scheid, Paul Segal, Eran Shin, Hyunjin Shou, Chong Slack, Frank

J Slightam, Cindie Smith, Richard Spencer, William C Stinson, E O Taing, Scott Takasaki, Teruaki Vafeados, Dionne Voronina, Ksenia Wang, Guilin Washington, Nicole L Whittle, Christina M Wu, Beijing Yan, Koon-Kiu Zeller, Georg Zha, Zheng Zhong, Mei Zhou, Xingliang modENCODE Consortium Ahringer, Julie Strome, Susan Gunsalus, Kristin C Micklem, Gos Liu, X Shirley Reinke, Valerie Kim, Stuart K Hillier, LaDeana W Henikoff, Steven Piano, Fabio Snyder, Michael Stein, Lincoln Lieb, Jason D Waterston, Robert H R01GM088565/GM/NIGMS NIH HHS/ Howard Hughes Medical Institute/ Wellcome Trust/United Kingdom New York, N.Y. Science. 2010 Dec 24;330(6012):1775-87. Epub 2010 Dec 22.

[7] ENCODE Project Consortium, "A user's guide to the encyclopedia of dna elements (encode)," *PLoS Biol*, vol. 9, no. 4, p. e1001046, Apr 2011.

[8] N. Nègre, C. D. Brown, L. Ma, C. A. Bristow, S. W. Miller, U. Wagner, P. Kheradpour, M. L. Eaton, P. Loriaux, R. Sealfon, Z. Li, H. Ishii, R. F. Spokony, J. Chen, L. Hwang, C. Cheng, R. P. Auburn, M. B. Davis, M. Domanus, P. K. Shah, C. A. Morrison, J. Zieba, S. Suchy, L. Senderowicz, A. Victorsen, N. A. Bild, A. J. Grundstad, D. Hanley, D. M. MacAlpine, M. Mannervik, K. Venken, H. Bellen, R. White, M. Gerstein, S. Russell, R. L. Grossman, B. Ren, J. W. Posakony, M. Kellis, and K. P. White, "A cis-regulatory map of the drosophila genome," *Nature*, vol. 471, no. 7339, pp. 527–31, Mar 2011.

[9] D. E. Newburger and M. L. Bulyk, "Uniprobe: an online database of protein binding

microarray data on protein-dna interactions," *Nucleic Acids Res*, vol. 37, no. Database issue, pp. D77–82, Jan 2009.

[10] T. L. Bailey and C. Elkan, "Fitting a mixture model by expectation maximization to discover motifs in biopolymers," *Proc Int Conf Intell Syst Mol Biol*, vol. 2, pp. 28–36, 1994.

[11] C. Linhart, Y. Halperin, and R. Shamir, "Transcription factor and microRNA motif discovery: the Amadeus platform and a compendium of metazoan target sets," *Genome Res*, vol. 18, no. 7, pp. 1180–9, Jul 2008.

[12] T. L. Bailey, "DREME: motif discovery in transcription factor chip-seq data," *Bioinformatics*, vol. 27, no. 12, pp. 1653–9, Jun 2011.

[13] C. T. Harbison, D. B. Gordon, T. I. Lee, N. J. Rinaldi, K. D. Macisaac, T. W. Danford, N. M. Hannett, J.-B. Tagne, D. B. Reynolds, J. Yoo, E. G. Jennings, J. Zeitlinger, D. K. Pokholok, M. Kellis, P. A. Rolfe, K. T. Takusagawa, E. S. Lander, D. K. Gifford, E. Fraenkel, and R. A. Young, "Transcriptional regulatory code of a eukaryotic genome," *Nature*, vol. 431, no. 7004, pp. 99–104, Sep 2004.

[14] M. A. Beer and S. Tavazoie, "Predicting gene expression from sequence," *Cell*, vol. 117, no. 2, pp. 185–198, 2004.

[15] A. C. Aisenberg, "New genetics of burkitt's lymphoma and other non-hodgkin's lymphomas," *The American journal of medicine*, vol. 77, no. 6, pp. 1083–90, 1984,

aisenberg, A C CA 30020-03/CA/NCI NIH HHS/ Am J Med. 1984 Dec;77(6):1083-90.

[16] J. Wittwer, J. Marti-Jaun, and M. Hersberger, "Functional polymorphism in alox15 results in increased allele-specific transcription in macrophages through binding of the transcription factor spi1," *Human mutation*, vol. 27, no. 1, pp. 78–87, 2006, wittwer, Jonas Marti-Jaun, Jacqueline Hersberger, Martin Hum Mutat. 2006 Jan;27(1):78-87.

[17] J. Theuns, N. Brouwers, S. Engelborghs, K. Sleegers, V. Bogaerts, E. Corsmit, T. De Pooter, C. M. van Duijn, P. P. De Deyn, and C. Van Broeckhoven, "Promoter mutations that increase amyloid precursor-protein expression are associated with alzheimer disease," *American journal of human genetics*, vol. 78, no. 6, pp. 936–46, 2006, theuns, Jessie Brouwers, Nathalie Engelborghs, Sebastiaan Sleegers, Kristel Bogaerts, Veerle Corsmit, Ellen De Pooter, Tim van Duijn, Cornelia M De Deyn, Peter P Van Broeckhoven, Christine Am J Hum Genet. 2006 Jun;78(6):936-46. Epub 2006 Apr 10.

[18] C. E. Lawrence, S. F. Altschul, M. S. Boguski, J. S. Liu, A. F. Neuwald, and J. C. Wootton, "Detecting subtle sequence signals: a gibbs sampling strategy for multiple alignment," *Science*, vol. 262, no. 5131, pp. 208–14, Oct 1993.

[19] G. Pavesi, G. Mauri, and G. Pesole, "An algorithm for finding signals of unknown length in dna sequences," *Bioinformatics*, vol. 17 Suppl 1, pp. S207–14, 2001.

[20] M. Tompa, N. Li, T. L. Bailey, G. M. Church, B. De Moor, E. Eskin, A. V. Favorov,

M. C. Frith, Y. Fu, W. J. Kent, V. J. Makeev, A. A. Mironov, W. S. Noble, G. Pavesi, G. Pesole, M. Régnier, N. Simonis, S. Sinha, G. Thijs, J. van Helden, M. Vandenbogaert, Z. Weng, C. Workman, C. Ye, and Z. Zhu, "Assessing computational tools for the discovery of transcription factor binding sites," *Nat Biotechnol*, vol. 23, no. 1, pp. 137–44, Jan 2005.

[21] F. P. Roth, J. D. Hughes, P. W. Estep, and G. M. Church, "Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mrna quantitation," *Nat Biotechnol*, vol. 16, no. 10, pp. 939–45, Oct 1998.

[22] G. Thijs, M. Lescot, K. Marchal, S. Rombauts, B. De Moor, P. Rouze, and Y. Moreau, "A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling," *Bioinformatics*, vol. 17, no. 12, pp. 1113–22, 2001, thijs, G Lescot, M Marchal, K Rombauts, S De Moor, B Rouze, P Moreau, Y England Oxford, England Bioinformatics. 2001 Dec;17(12):1113-22.

[23] J. S. Liu, A. F. Neuwald, and C. E. Lawrence, "Bayesian models for multiple local sequence alignment and gibbs sampling strategies," *Journal of the American Statistical Association*, vol. 90, no. 432, pp. 1156–1170, Dec 1995.

[24] A. Visel, M. J. Blow, Z. Li, T. Zhang, J. A. Akiyama, A. Holt, I. Plajzer-Frick, M. Shoukry, C. Wright, F. Chen, V. Afzal, B. Ren, E. M. Rubin, and L. A. Pennacchio, "Chip-seq accurately predicts tissue-specific activity of enhancers," *Nature*, vol. 457, no. 7231, pp. 854–8, Feb 2009.

[25] W.-J. Welboren, M. A. van Driel, E. M. Janssen-Megens, S. J. van Heeringen, F. C. Sweep, P. N. Span, and H. G. Stunnenberg, "ChIP-Seq of ERalpha and RNA polymerase II defines genes differentially responding to ligands," *EMBO J*, vol. 28, no. 10, pp. 1418–28, May 2009.

[26] A. Barski, S. Cuddapah, K. Cui, T.-Y. Roh, D. E. Schones, Z. Wang, G. Wei, I. Chepelev, and K. Zhao, "High-resolution profiling of histone methylations in the human genome," *Cell*, vol. 129, no. 4, pp. 823–37, May 2007.

[27] J. Rozowsky, G. Euskirchen, R. K. Auerbach, Z. D. Zhang, T. Gibson, R. Bjornson, N. Carriero, M. Snyder, and M. B. Gerstein, "Peakseq enables systematic scoring of chip-seq experiments relative to controls," *Nat Biotechnol*, vol. 27, no. 1, pp. 66–75, Jan 2009.

[28] Y. Zhang, T. Liu, C. A. Meyer, J. Eeckhoute, D. S. Johnson, B. E. Bernstein, C. Nusbaum, R. M. Myers, M. Brown, W. Li, and X. S. Liu, "Model-based analysis of ChIP-Seq (MACS)," *Genome Biol*, vol. 9, no. 9, p. R137, 2008.

[29] J. E. Phillips and V. G. Corces, "Ctcf: master weaver of the genome," *Cell*, vol. 137, no. 7, pp. 1194–211, Jun 2009.

[30] Z. F. Chen, A. J. Paquette, and D. J. Anderson, "Nrsf/rest is required in vivo for repression of multiple neuronal target genes during embryogenesis," *Nat Genet*, vol. 20, no. 2, pp. 136–42, Oct 1998.

[31] N. Mori, C. Schoenherr, D. J. Vandenbergh, and D. J. Anderson, "A common silencer element in the scg10 and type ii na+ channel genes binds a factor present in nonneuronal cells but not in neuronal cells," *Neuron*, vol. 9, no. 1, pp. 45–54, Jul 1992.

[32] E. R. Levin, "Cell localization, physiology, and nongenomic actions of estrogen receptors," *J Appl Physiol*, vol. 91, no. 4, pp. 1860–7, Oct 2001.

[33] L. Song, Z. Zhang, L. L. Grasfeder, A. P. Boyle, P. G. Giresi, B.-K. Lee, N. C. Sheffield, S. Gräf, M. Huss, D. Keefe, Z. Liu, D. London, R. M. McDaniell, Y. Shibata, K. A. Showers, J. M. Simon, T. Vales, T. Wang, D. Winter, Z. Zhang, N. D. Clarke, E. Birney, V. R. Iyer, G. E. Crawford, J. D. Lieb, and T. S. Furey, "Open chromatin defined by DNaseI and FAIRE identifies regulatory elements that shape cell-type identity," *Genome Res*, vol. 21, no. 10, pp. 1757–67, Oct 2011.

[34] X. Chen, H. Xu, P. Yuan, F. Fang, M. Huss, V. B. Vega, E. Wong, Y. L. Orlov, W. Zhang, J. Jiang, Y.-H. Loh, H. C. Yeo, Z. X. Yeo, V. Narang, K. R. Govindarajan, B. Leong, A. Shahab, Y. Ruan, G. Bourque, W.-K. Sung, N. D. Clarke, C.-L. Wei, and H.-H. Ng, "Integration of external signaling pathways with the core transcriptional network in embryonic stem cells," *Cell*, vol. 133, no. 6, pp. 1106–17, Jun 2008.

[35] W. Niu, Z. J. Lu, M. Zhong, M. Sarov, J. I. Murray, C. M. Brdlik, J. Janette, C. Chen, P. Alves, E. Preston, C. Slightham, L. Jiang, A. A. Hyman, S. K. Kim, R. H. Waterston, M. Gerstein, M. Snyder, and V. Reinke, "Diverse transcription factor binding features

revealed by genome-wide ChIP-seq in *C. elegans*," *Genome Res*, vol. 21, no. 2, pp. 245–54, Feb 2011.

[36] D. Lee, R. Karchin, and M. A. Beer, "Discriminative prediction of mammalian enhancers from dna sequence," *Genome Res*, vol. 21, no. 12, pp. 2167–80, Dec 2011.

[37] S. Tavazoie, J. D. Hughes, M. J. Campbell, R. J. Cho, and G. M. Church, "Systematic determination of genetic network architecture," *Nature genetics*, vol. 22, no. 3, pp. 281–5, 1999, tavazoie, S Hughes, J D Campbell, M J Cho, R J Church, G M Nat Genet. 1999 Jul;22(3):281-5.

[38] E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller, and N. Friedman, "Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data," *Nat Genet*, vol. 34, no. 2, pp. 166–76, Jun 2003.

[39] A. Kundaje, S. Lianoglou, X. Li, D. Quigley, M. Arias, C. H. Wiggins, L. Zhang, and C. Leslie, "Learning regulatory programs that accurately predict differential expression with medusa," *Ann N Y Acad Sci*, vol. 1115, pp. 178–202, Dec 2007.

[40] A. Kundaje, X. Xin, C. Lan, S. Lianoglou, M. Zhou, L. Zhang, and C. Leslie, "A predictive model of the oxygen and heme regulatory network in yeast," *PLoS Comput Biol*, vol. 4, no. 11, p. e1000224, Nov 2008.

[41] Y. Halperin, C. Linhart, I. Ulitsky, and R. Shamir, "Allegro: analyzing expression

and sequence in concert to discover regulatory programs," *Nucleic Acids Res*, vol. 37, no. 5, pp. 1566–79, Apr 2009.

[42] A. P. Gasch, P. T. Spellman, C. M. Kao, O. Carmel-Harel, M. B. Eisen, G. Storz, D. Botstein, and P. O. Brown, "Genomic expression programs in the response of yeast cells to environmental changes," *Molecular biology of the cell*, vol. 11, no. 12, pp. 4241–4257, 2000.

[43] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, and B. Futcher, "Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization," *Molecular biology of the cell*, vol. 9, no. 12, pp. 3273–3297, 1998.

[44] M. J. Brauer, C. Huttenhower, E. M. Airoldi, R. Rosenstein, J. C. Matese, D. Gresham, V. M. Boer, O. G. Troyanskaya, and D. Botstein, "Coordination of growth rate, cell cycle, stress response, and metabolic activity in yeast," *Molecular biology of the cell*, vol. 19, no. 1, pp. 352–367, 2008.

[45] L. R. Baugh, A. A. Hill, D. K. Slonim, E. L. Brown, and C. P. Hunter, "Composition and dynamics of the *Caenorhabditis elegans* early embryonic transcriptome," *Development*, vol. 130, no. 5, pp. 889–900, 2003, baugh, L Ryan Hill, Andrew A Slonim, Donna K Brown, Eugene L Hunter, Craig P England Cambridge, England Development. 2003 Mar;130(5):889-900.

[46] A. A. Hill, C. P. Hunter, B. T. Tsung, G. Tucker-Kellogg, and E. L. Brown, "Genomic

analysis of gene expression in *C. elegans*," *Science*, vol. 290, no. 5492, pp. 809–12, 2000, hill, A A Hunter, C P Tsung, B T Tucker-Kellogg, G Brown, E L New York, N.Y. Science. 2000 Oct 27;290(5492):809-12.

[47] S. J. McKay, R. Johnsen, J. Khattra, J. Asano, D. L. Baillie, S. Chan, N. Dube, L. Fang, B. Goszczynski, E. Ha, E. Halfnight, R. Hollebakken, P. Huang, K. Hung, V. Jensen, S. J. Jones, H. Kai, D. Li, A. Mah, M. Marra, J. McGhee, R. Newbury, A. Pouzyrev, D. L. Riddle, E. Sonnhammer, H. Tian, D. Tu, J. R. Tyson, G. Vatcher, A. Warner, K. Wong, Z. Zhao, and D. G. Moerman, "Gene expression profiling of cells, tissues, and developmental stages of the nematode *C. elegans*," *Cold Spring Harbor symposia on quantitative biology*, vol. 68, pp. 159–69, 2003, mcKay, S J Johnsen, R Khattra, J Asano, J Baillie, D L Chan, S Dube, N Fang, L Goszczynski, B Ha, E Halfnight, E Hollebakken, R Huang, P Hung, K Jensen, V Jones, S J M Kai, H Li, D Mah, A Marra, M McGhee, J Newbury, R Pouzyrev, A Riddle, D L Sonnhammer, E Tian, H Tu, D Tyson, J R Vatcher, G Warner, A Wong, K Zhao, Z Moerman, D G AG-12689/AG/NIA NIH HHS/ GM-60151/GM/NIGMS NIH HHS/ Cold Spring Harb Symp Quant Biol. 2003;68:159-69.

[48] T. W. Harris, I. Antoshechkin, T. Bieri, D. Blasiar, J. Chan, W. J. Chen, N. De La Cruz, P. Davis, M. Duesbury, R. Fang, J. Fernandes, M. Han, R. Kishore, R. Lee, H.-M. Müller, C. Nakamura, P. Ozersky, A. Petcherski, A. Rangarajan, A. Rogers, G. Schindelman, E. M. Schwarz, M. A. Tuli, K. Van Auken, D. Wang, X. Wang, G. Williams,

K. Yook, R. Durbin, L. D. Stein, J. Spieth, and P. W. Sternberg, "WormBase: a comprehensive resource for nematode research," *Nucleic Acids Res*, vol. 38, no. Database issue, pp. D463–7, Jan 2010.

[49] D. GuhaThakurta, L. A. Schriefer, R. H. Waterston, and G. D. Stormo, "Novel transcription regulatory elements in *Caenorhabditis elegans* muscle genes," *Genome Res*, vol. 14, no. 12, pp. 2457–68, Dec 2004.

[50] H. T. Schwartz and H. R. Horvitz, "The *C. elegans* protein CEH-30 protects male-specific neurons from apoptosis independently of the Bcl-2 homolog CED-9," *Genes Dev*, vol. 21, no. 23, pp. 3181–94, Dec 2007.

[51] M. Boxem and S. van den Heuvel, "*C. elegans* class B synthetic multivulva genes act in G(1) regulation," *Curr Biol*, vol. 12, no. 11, pp. 906–11, Jun 2002.

[52] M. Kellis, N. Patterson, M. Endrizzi, B. Birren, and E. S. Lander, "Sequencing and comparison of yeast species to identify genes and regulatory elements," *Nature*, vol. 423, no. 6937, pp. 241–54, May 2003.

[53] X. Xie, T. S. Mikkelsen, A. Gnirke, K. Lindblad-Toh, M. Kellis, and E. S. Lander, "Systematic discovery of regulatory motifs in conserved regions of the human genome, including thousands of ctcf insulator sites," *Proc Natl Acad Sci U S A*, vol. 104, no. 17, pp. 7145–50, Apr 2007.

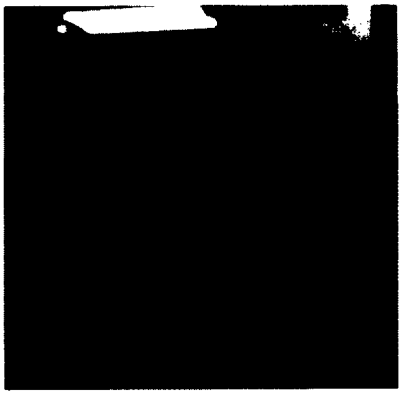
[54] A. Nakao, M. Yoshihama, and N. Kenmochi, "Rpg: the ribosomal protein gene database," *Nucleic Acids Res*, vol. 32, no. Database issue, pp. D168–70, Jan 2004.

[55] S. Gupta, J. A. Stamatoyannopoulos, T. L. Bailey, and W. S. Noble, "Quantifying similarity between motifs," *Genome Biol*, vol. 8, no. 2, p. R24, 2007.

[56] M. A. Beer, "CGAGACCC repeat hypothesis," personal communication, 2009.

[57] K. Kryukov, K. Sumiyama, K. Ikeo, T. Gojobori, and N. Saitou, "A new database (gcd) on genome composition for eukaryote and prokaryote genome sequences and their initial analyses," *Genome Biol Evol*, vol. 4, no. 4, pp. 501–12, 2012.

Vita



Rahul Karnik received the B.S. degree in Biology and Computer Science from from Davidson College in 2000 and a Masters in Computer Science from the University of Illinois (Urbana-Champaign) in 2003. He enrolled in the Biomedical Engineering Ph.D. program at Johns Hopkins University in 2006. He was inducted into the Phi Beta Kappa and Beta Beta Beta honor societies in 2000 and won the Direaux Award for Excellence in Undergraduate Research in 2000.