Chapter 7
Regulatory Motif Analysis

Alan Moses and Saurabh Sinha

7.1 Introduction – Pattern Recognition and Discovery in *cis*-Regulatory Informatics

The first complete genome sequences of eukaryotes revealed that much of the genetic material did not code for protein sequences (Lander et al. 2001; Venter et al. 2001). Although this noncoding DNA was once thought to be "junk" DNA, it is now appreciated that large portions of it are actively conserved over evolution (Waterston et al. 2002; Johnston and Stormo 2003), suggesting that these regions contain important functional elements.

A first hypothesis about the function of this noncoding DNA is that it is involved in the regulation of gene activity. One of the best-understood mechanisms of gene regulation is the modulation of transcriptional initiation by sequence specific DNA binding proteins (or transcription factors). These proteins recognize short sequences in noncoding DNA that fall into families or contain consensus patterns or motifs.

In general, we have little understanding of how the information in noncoding regulatory sequence specifies complex patterns of gene expression. In analogy to the genetic code that translates DNA sequence to amino acids in a protein, researchers have suggested the existence of an unknown "*cis*-regulatory code" that translates DNA sequence to patterns of gene expression (Levine and Davidson 2005).

To specify complex patterns of regulation, genes are often regulated by multiple transcription factors, and the binding sites for these factors are organized into discrete regulatory regions, often called "enhancers" or "*cis*-regulatory modules." These regulatory regions are often found in the proximal 5′ promoter regions, but they may also occur much further upstream, downstream, or in intronic regions.

A. Moses (✉)
Department of Cell & Systems Biology, University of Toronto, 25 Willcocks Street, Toronto, ON, Canada, M5S 3B2
e-mail: alanmoses@utoronto.ca

S. Sinha
Dept. of Computer Sciences , University of Illinois, Urbana-Champaign, 201 N. Goodwin Ave, Urbana, IL 61801
e-mail: sinhas@uiuc.edu

D. Edwards et al. (eds.), *Bioinformatics*: *Tools and Applications,*
DOI 10.1007/978-0-387-92738-1_7,

It is these regulatory regions that execute the *cis*-regulatory code, and systematic identification of these noncoding DNA regulatory regions and the binding sites within them is of great interest in postgenome era molecular biology; the sheer vastness of the noncoding DNA sequence to be analyzed implies that computational methods will have an important role to play.

7.1.1 Two Major Challenges

The biological questions regarding *cis*-regulatory sequences can be broken into two major parts. The first can be thought of as identifying the patterns or motifs associated with each transcription factor. Given this set of patterns, the next challenge is to identify the specific positions in the noncoding DNA where the transcription factors actually bind in vivo. This is directly analogous to the two steps of a statistical clustering problem; first to identify the clusters and second to assign each datapoint to a cluster. As we shall see, sophisticated statistical methods aim to solve these simultaneously. This distinction is important because the experimental approaches to attack these problems can be quite different so that historically they were distinct problems. Here we will use the terminology that "motifs" or "consensus sequences" refer to the representations of specificity or patterns associated with transcription factors, whereas "instances," "matches," or "regulatory sequences" refer to the specific places in noncoding DNA where transcription factors are predicted or known to bind.

7.1.2 Overview of Regulatory Informatics

This chapter will cover three reasonably well defined types of bioinformatic applications. The first are databases and repositories for organizing, storing, and distributing experimentally identified regulatory sequences and motifs; next are pattern matching site prediction methods that begin with known motifs or patterns and attempt to predict the regulatory sequences in noncoding DNA; and finally are de novo or ab initio motif-finding methods that attempt to discover the motifs (and perhaps matches to them simultaneously). In each section, we provide a table with some examples of software implementations. However, these tables are not intended to be comprehensive, but are rather representative of the work in the area. As regulatory bioinformatics is still rapidly developing, readers should refer to recent reviews to find the latest implementations.

7.2 Databases and Repositories for Regulatory Sequences and Motifs

The simplest function of online databases is to store binding sites that have been characterized through biochemical and genetic experiments (Heinemeyer et al. 1998). The technically difficult aspect of these applications is to extract the

experimental data from the primary biological literature. Usually this is performed by experts who read large numbers of papers and enter the results into the databases. More recently, computational text mining approaches have also been applied to extract regulatory sequences and information from the literature (Aerts et al. 2008). Several databases of curated motifs are described in Table 7.1.

7.2.1 Mathematical/Computational Representations of Motifs

Given a set of experimentally characterized regulatory sequences that are known to be bound by a particular factor, a first task is to identify and summarize the specificity of the transcription factor in a motif or consensus. There are two popular strategies to do this.

Table 7.1 Databases for storing experimentally identified *cis*-regulatory sequences

Resource	Types of data	Tools	Notes
Transfac[a]	Classification of transcription factors, experimentally proven binding sites, counts matrices	Many	Available with subscription
Jaspar[b]	Matrices	Logos, reverse complements, and more	Freely available, plant and animal matrices only
SCPD[c]	Transcription factors, characterized binding sites, counts matrices, consensus sequences	Pattern matching	Freely available, *Saccharomyces cerevisiae* only
REDfly[d]/Drosophila DNase I Footprint Database[e]	Transcription factor binding sites and regulatory regions (CRMs) z	Links to genome-wide alignments	Freely available, *Drosophila melanogaster* only
ORegAnno[f]	Regulatory regions, Transcription factor binding sites, includes evidence for each record		Freely available, open source data and web application, integrates information from multiple databases
PRODORIC[g]	Transcription factor binding sites, operons, matrices, promoter architecture	Composite patterns, Genome Browser, and more	Freely available, prokaryotes only

[a] (Wingender et al. 1996), [b] (Sandelin et al. 2004a), [c] (Zhu and Zhang 1999), [d] (Gallo et al. 2006), [e] (Bergman et al. 2005), [f] (Montgomery et al. 2006), [g] (Münch et al. 2003)

Matrix representation: Here each position of the motif is treated as a multinomial distribution on the residues. This representation of motifs is used in probabilistic methods and implies an infinitely large, continuous space of motifs. Despite this, the matrix representation has several attractive features, discussed in more detail later on in the chapter:

(i) The parameters of the multinomial at each position can be readily estimated using statistical inference methods.
(ii) The multinomial distribution at each position can be used to obtain a measure of "information" contained in each position in the motif (Schneider et al. 1986).
(iii) These multinomials can be transformed into "log-odds" or weight matrices, which are a computationally convenient form to store classifiers (Stormo 2000).
(iv) Experimental and theoretical evidence suggests that this representation is related to the binding energy of the protein-DNA interactions (Berg and von Hippel 1987).

Consensus representation: Consensus representations of motifs are more familiar to most biologists and have also been important for computational approaches. A consensus representation of a motif may simply be the most frequent letter at each position in the motif. Alternatively, "degeneracy codes" or mismatches may be used to represent non-optimal matches. The main computational advantages of the consensus representation are:

(i) The space of motifs is discrete, so computational strategies for matching and de novo motif finding are highly efficient, and
(ii) The space of motifs is finite, so computational strategies for de novo motif-finding can aim to search exhaustively.

To illustrate the various representations of motifs, we consider a set of known binding sites (called GATA sites) from the SCPD database.

```
>YIR032C   GATAAG
>YIR032C   GGTAAG
>YIR032C   GATAAG
>YJL110C   GATAAT
>YKR034W   GATAGA
>YKR034W   GATAAC
>YKR039W   GATAAG
>YKR039W   GATAAC
```

The consensus representations for this motif might be GATAAG with one mismatch allowed or GRTARN where *R* represents *A* or *G* and *N* represents any base.

We next derive the maximum likelihood estimate (MLE) for the frequency matrix representation using this example. This example will introduce the notation and terminology that we will use later on in the chapter. We represent the sequence data at each position as a four-dimensional vector, where each dimension corresponds to one of the bases A, C, G, T.

	A	C	G	T
1	0	0	8	0
2	7	0	1	0
3	0	0	0	8
⋮	8	0	0	0
	7	0	1	0
w	1	2	4	1

This is often referred to as a "counts" matrix and such matrices are provided by many databases.

The likelihood of the data is defined as the probability of the data given the model, i.e.,

$$L(X) = p(X \mid \text{model})$$

where $p(A|B)$ represents that probability of the random variable A conditioned on the random variable B. Under the multinomial model for each position, the likelihood of the counts matrix X is the product of the probability of each base, in our case

$$\begin{aligned} L(X) &= p(X \mid \text{motif}) \\ &= f_{1G}{}^{8} \times f_{2A}{}^{7} \times f_{2G}{}^{1} \times f_{3T}{}^{8} \times f_{4A}{}^{8} \times f_{5A}{}^{7} \times f_{5G}{}^{1} \times f_{6A}{}^{1} \times f_{6C}{}^{2} \times f_{6G}{}^{4} \times f_{6T}{}^{1} \end{aligned}$$

where f are the parameters of the multinomial at each position in the motif. This can be written compactly as

$$p(X \mid \text{motif}) = \prod_{i=1}^{w} \prod_{b \in ACGT} f_{ib}^{X_{ib}}$$

where i is the position in the motif, and b indexes the bases. To find maximum likelihood estimators (MLEs) of these parameters, we simply need to find the values of the parameters that maximize this function. The simplest strategy to do this is take derivatives with respect to the parameters and set them to zero. However, in this case, as in many probabilistic models, taking derivatives of the products is difficult. To get around this issue, we instead optimize the logarithm of the likelihood, such that the products become sums. Because the logarithm is monotonic, any values of the parameters that maximize the logarithm of the likelihood will also maximize the likelihood. In addition, we note that we will not accept any values of the parameters as the MLEs: we want to enforce the constraint that the probabilities at each position must sum to one, $\sum_b f_{ib} = 1$. Such constraints can be included using Lagrange multipliers. Putting

all this together gives

$$\log[L(X)] = \sum_{i=1}^{w} \sum_{b \in ACGT} X_{ib} \log f_{ib} + \lambda \left(1 - \sum_b f_{ib} \right)$$

as the function we wish to maximize, where λ is the Lagrange multiplier. We now set the derivatives with respect to each of the frequency parameters to zero.

For example, using the linearity of the derivative and that $\frac{\mathrm{d}}{\mathrm{d}x}\log(x) = \frac{1}{x}$, for the parameter at position j, for base c, we have

$$\frac{\partial}{\partial f_{jc}}\log[L(X)] = \frac{X_{jc}}{f_{jc}} - \lambda = 0$$

Solving this and substituting into the constraint gives $f_{jc} = \frac{X_{jc}}{\lambda}$ and $\lambda = \sum_b X_{jb}$, the total number of observations at position j.

Thus, we have the intuitive result that the MLE of the frequency for each base is just the number of times we observed that base (X_{qc}) divided by the total number of observed bases at that position. It is important to note that in our example, our estimates of the frequency at position 1 are $f_1 = (0,0,1,0)$. This implies that based on our data we conclude that there is no probability of observing "*A*," "*C*," or "*T*" at this position. Given that we have only observed 8 examples of this motif, this seems a somewhat overconfident claim. Therefore, it is common practice to "soften" the MLEs by adding some "fake" or pseudo data to each position in the counts matrix. For example, if we use 1 as the psuedocount, our estimate of the frequencies at the first position becomes $f_1 = (1/12,1/12,9/12,1/12)$, and reflects our uncertainty about the estimates. These pseudocounts can be justified statistically using the concept of prior probabilities, which is discussed in detail elsewhere (Durbin et al. 1998).

7.3 Identifying Binding Sites Given a Known Motif

Given a matrix or consensus representation of a motif, we now consider the problem of identifying new examples of binding sites.

Given a consensus representation, it is possible to say for each possible sequence of length w, whether it is a match to the motif or not. For example, a consensus sequence with a mismatch allowed at any position will match $1+4w$ of the 4^w possible sequences of length w. For our example of GATAAG with one mismatch, we have $\frac{1+4\times 6}{4^6} = \frac{25}{4096} = 0.0061$. This means that 0.6% of 6-mers will match this motif. For the degeneracy code representation, the number of sequences that match is the product of the degeneracies at each position. For GRTARN, this is $\frac{1\times 2\times 1\times 1\times 2\times 4}{4^6} = \frac{16}{4096} = 0.0039$. Although this may seem to be a few (99.6% of sequences do not match), in a random genome of 100MB, we expect ~390,000 matches by chance! This is two orders of magnitude greater than the maximal reasonable expectation for the number of GATA sites in a genome. Although real genomes are not random, matches to motifs do occur frequently by chance, swamping the number of matches that are functionally bound in the cell. The so-called "Futility Theorem" (Wasserman and Sandelin 2004) conjectures that the large number of random matches relative to functional binding sites makes identification based on pattern matching futile.

Using the matrix representation of the motif, for any sequence of length w, we can follow a number of explicit statistical classification strategies to decide whether a sequence is an example of the binding site. Here we use X to represent a single sequence of length w.

One commonly used test statistic to compare two models is the likelihood ratio (not to be confused with a likelihood ratio test). In our case, we compare the likelihood that the sequence of interest, X, is drawn from our motif frequency matrix, to the likelihood that X was drawn from a background null distribution. There are many ways to construct such a background distribution; here we consider the simplest, namely, that the background is a single multinomial.

If the sequence we are considering is GATAAG, $X = \begin{matrix} A \\ C \\ G \\ T \end{matrix} \begin{matrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{matrix}$,

we can calculate the likelihood of X under the models as we did for the counts matrix above. In the case of the matrix model for the motif (f) and a background distribution (g), the likelihood ratio is simply

$$S(X) = \log \frac{p(X \mid \text{motif})}{p(X \mid bg)} = \log \frac{\prod_{i=1}^{w} \prod_{b \in ACGT} f_{ib}^{X_{ib}}}{\prod_{i=1}^{w} \prod_{b \in ACGT} g_{b}^{X_{ib}}} = \sum_{i=1}^{w} \sum_{b \in ACGT} X_{ib} \log\left(\frac{f_{ib}}{g_b}\right)$$

Thus, $S(X)$ provides a quantitative measure of how similar a sequence is to the frequency matrix. When $S(X)>0$, X is more similar to the motif model than the background model.

To identify new examples of the motif in a new sequence, a typical strategy is to compute the statistic, S, for each overlapping subsequence of length w in the sequence. For computational simplicity, this is often done using a "weight" matrix in which entries are given by $M_{ib} = \log\left(\frac{f_{ib}}{g_b}\right)$, where as above, i indexes the position in the motif, and b indexes the nucleotide bases. To calculate S, one simply adds up the entries in this matrix corresponding to the observed bases. In our notation, this can be written simply as the inner product

$$S(X) = M \cdot X$$

For the example above, using g = (0,3,0,2,0,2,0,3) this is

$$S(GATAAG) = \begin{bmatrix} -1.28 & -0.875 & 1.32 & -1.28 \\ 0.799 & -0.875 & -0.182 & -1.28 \\ -1.28 & -0.875 & -0.875 & 0.916 \\ 0.916 & -0.875 & -0.875 & -1.28 \\ 0.799 & -0.875 & -0.182 & -1.28 \\ -0.588 & 0.223 & 0.734 & 0.588 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} = 5.485$$

Table 7.2 Tools for matrix matching

Tool	Purpose	Notes
Patser[a]	Matching known matrices to sequences	Calculates *P*-values
Delila-genome[b]	Matching known matrices to sequences	Information theory-based scoring

[a] (Hertz and Stormo 1999), [b] (Gadiraju et al. 2003)

the maximum possible likelihood ratio score for this matrix. Some examples of implementations of matrix matching are described in Table 7.2.

7.3.1 Choosing a Cutoff for the Likelihood Ratio

An important question in using such a classification framework is how high a value of S is needed before we can be confident that we have identified a novel example of the motif. Several approaches to this problem have been proposed. There is a finite set of possible scores to a matrix model, and the maximum and minimum score for each matrix are different. In order to standardize the scores when comparing between multiple matrix models, the likelihood ratio for a particular sequence is often transformed into normalized score that reflects how close it is to the maximum score. For example, the transformation

$$S(X) = \frac{S(X) - S_{\text{MIN}}}{S_{\text{MAX}} - S_{\text{MIN}}}$$

standardizes the scores to fall between zero and one, which can be interpreted intuitively.

We next consider three statistically motivated approaches to standardizing the scores from matrices in order to choose a cutoff. The classical statistical treatment (Staden 1989) of this problem is to treat the background distribution as a null hypothesis and consider the *P*-value or probability of having observed the score S or more under the background distribution. In order to calculate *P*-values, we must add up the probabilities of all sequences X that have a score greater than S, which means enumerating ~4^w sequences. However, because the positions in the motif are treated independently, these calculations can be done recursively in computational time ~$4w$. This allows us to calculate, for each value S, the *P*-value under the null hypothesis (Fig. 7.1).

It is important to note that the validity of these *P*-values depends on the accuracy of the null model or background distribution. For this reason, it is often preferred to use an "empirical" null distribution in which the *P*-value is computed simply by counting the number of times a score of S or more is achieved in a large sample of "null" sequences comprising genomic sequence not thought to contain real examples of the binding site.

Regardless of the method for obtaining these *P*-values, in a sequence of length l, we expect to test $l - w$ subsequences, and therefore can apply a multiple testing

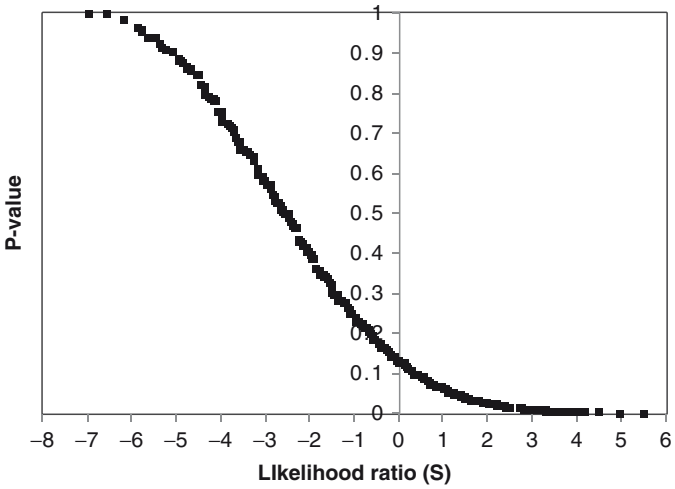

Fig. 7.1 Exact *P*-values for the likelihood ratio score

correction to these *P*-values to control the expected false positives. For example, if we are to search a promoter of one KB of sequence, and we expect one match, we might choose the cutoff $0.05/1000 = 5\times10^{-5}$, known as the Bonferoni correction. Alternatively, we can express the confidence as an *E*-value (or expect value, which is the *P*-value multiplied by the number of tests) or using a correction for false discovery rate (Benjamini and Hochberg 1995).

A second statistical approach to choosing a threshold for classification is to note that likelihood ratio statistics such as this have the attractive property that when $S>0$, the likelihood of the sequence under the motif model is higher than that under the background model, and under the "maximum likelihood" (ML) rule for classification we should assign the data to the model that has the higher likelihood. However, this rule carries the implicit assumption that our prior expectation is that each subsequence is equally likely to be an example of the motif or the background model. In a real regulatory region, the unknown locations of binding sites might represent a small number of positions amongst thousands of basepairs of background sequences. This "prior" expectation can be incorporated in a maximum a posteriori classification rule (MAP) by using $S > \log\left(\frac{1-\pi}{\pi}\right)$, where π is the prior probability of observing the motif.

Finally, using these priors, it is also possible to compute the posterior probability that a given position in the sequence is an example of the motif using Bayes' theorem

$$p(\text{motif} \mid X) = \frac{p(X \mid \text{motif})p(\text{motif})}{p(X \mid \text{motif})p(\text{motif}) + p(X \mid bg)p(bg)} = \frac{1}{1+\frac{1-\pi}{\pi e^{S(X)}}}$$

This yields a number between 0 and 1 that is intuitively interpretable and can be expressed as a function of the likelihood ratio statistic $S(X)$.

Classification based on the likelihood ratio affords greater control of the false positives, as it allows us to increase the cutoff as the search becomes large, thus reducing the number of spurious matches. However, even the best possible match to the matrix will still occur by chance in about 4^w base-pairs. Thus, while the likelihood ratio gives a quantitative measure of how close a putative sequence is to the matrix, it does not address the large number of expected matches in random sequence – matrix matching does not escape the Futility Theorem.

7.3.2 Relationship to Information Theory

Given this statistical model of motifs, it is possible to ask for each frequency matrix, how strong a classifier is it. In other words, given that a sequence is a true example of the motif, how easy is it to distinguish from the random background. To measure this quantitatively, we can calculate the average or expectation of S given than the sequences have come from the motif model. This average is over all possible sequences of length w. However, as with the P-value calculation above, we can use the independence of positions in the motif model to factor this sum

$$E[S(X) \mid motif] = \sum_{\text{all } X} S(X) p(X \mid \text{motif}) = \sum_{p=1}^{w} \sum_{b \in ACGT} f_{pb} \log\left(\frac{f_{pb}}{g_b}\right) = \mathrm{I}$$

where $E[\ ]$ represents the expectation. Interestingly, this formula can also be obtained using an information theoretic approach (Schneider et al. 1986). If base 2 is used for the logarithms, I is known as the "information content" in the motif and gives a value in bits (Schneider et al. 1986). Several interesting bioinformatic results can be obtained from this information theoretic perspective. For example, in the case of the uniform background distribution, the probability of observing a match to the matrix with score >0 in random sequence is given by 2^{-I}. Furthermore, the information theoretic perspective yields an intuitive relationship between the presence of binding sites in noncoding sequence and the entropic force of random mutation. Indeed, some early "de novo" motif finding approaches (Stormo and Hartzell 1989) were motivated by the assumption that the motif in a set of binding sites would be the maximally informative motif, and this could be quantified by searching for the patterns with the most information.

The information content of a motif as defined above is also the Kullback–Leibler divergence (Kullback and Leible 1951) between the motif model and the background distribution, and can be shown to be related to the average binding energy of the transcription factor for its target binding sites (Berg and von Hippel 1987). The convergence of the statistical, information theoretic and biophysical treatments of this problem on the formula above is a great achievement in computational biology, and suggests that there are deep connections between the models that have motivated

these analyses. As we shall see below, the likelihood ratio, $S(X)$ will have an important role to play in de novo motif finding as well.

7.4 Second Generation Regulatory Sequence Prediction Methods: Combinations and Conservation of Motifs to Improve Classification Power

A simple calculation of the *P*-values or information content for an example motif indicates that in a large genome, high-scoring matches to the motif matrix are very likely to appear even under the background null model. This is the motivation of the so-called Futility Theorem: if *bona fide* regulatory elements are rare, searching for them with motifs as described above will yield many false positives and have little power to identify functional examples of binding sites. Two major approaches have been developed to improve predictive power, and we discuss each of these in turn.

7.4.1 Exploiting Binding Site Clustering

The first method is to search for combinations or clusters of transcription factor binding sites (Wasserman and Fickett 1998; Markstein and Levine 2002). Some transcription factors tend to have multiple binding sites in short regions, so as to increase the probability of binding to the DNA in that region. This results in what is sometimes called "homotypic clustering" of binding sites (Lifanov et al. 2003), i.e., an above average density of binding sites of the same factor at a locus. Moreover, transcriptional regulation is known to be combinatorial, i.e., multiple transcription factors often act in concert to regulate the activity of a target gene. Therefore, regulatory sequences may have binding sites for multiple transcription factors, a phenomenon called "heterotypic clustering." From the perspective of pattern recognition, the presence of multiple binding sites improves the signal to noise ratio.

To take advantage of the additional signal, methods (Table 7.3) have been designed to search for regions of the genome that contain multiple closely related binding sites. A simple implementation of this idea is to begin with one or more motifs, predict sites matching each motif using the method described above, and count the number of sites in a sequence window of some fixed length (Berman et al. 2002; Halfon et al. 2002). One would then scan the entire genome for windows with the largest numbers of sites and the predicted binding sites in those windows would be reported.

This simple approach has been shown to empirically add statistical power to regulatory sequence prediction. However, one potential problem with this scheme is its use of ad hoc (and usually high) thresholds on matches to motifs when the matrix representation is used. There are biological examples of regulatory sequences that function by using several weak affinity binding sites rather than one or a few strong sites (Mannervik et al. 1999). Identifying weak sites would require very low thresholds

Table 7.3 Methods to search for clusters of binding sites

Tool	Purpose	Notes
MAST[a]	Identifies matches to motif matrix	Combines *P*-values for multiple motifs
cis-analyst[b]	Identifies clusters of matrix matches	User defined sliding window and matrix cutoffs
Stubb[c]	Identifies clusters of matrix matches	Uses HMM; User defined sliding window
Cluster buster[d]	Identifies clusters of matrix matches	Uses HMM; window length automatically learned

[a] (Bailey and Gribskov 1998), [b] (Berman et al. 2002), [c] (Sinha et al. 2006), [d] (Frith et al. 2003)

on $S(X)$ in our computational procedure (Sect. 3), leading to a large number of site predictions, including several false ones. What is needed here is a method that considers both the number and strengths of binding sites in a candidate regulatory sequence: it should accommodate the presence of weak binding sites, but more of these should be required to provide as much confidence as a smaller number of strong sites. Since the strength of binding sites cannot be captured by consensus string models, the following discussion will assume a matrix model of motifs.

One way to allow for the clustering of motifs of different strengths is to score every substring X in a sequence window using the score $S(X)$ described above, and determine the sum of these scores. That is, the sequence window Y is scored by $T(Y) = \sum_{i=1}^{|Y|} S(Y_i)$ where Y_i is the substring at offset i in Y. This allows us to assess the extent of homotypic clustering in Y, while allowing for strong as well as weak sites, and without imposing any thresholds. This scheme could be extended to work with more than one motif by simply summing over the motifs. Notice however, that adding the different $S(Y_i)$ terms amounts to multiplying probabilities of events, which is questionable since the events (different Y_i) are not independent. Another alternative is to use $T(Y) = \sum_{i=1}^{Y} e^{S(Y_i)}$. This in fact is more justified statistically, as we see next. Consider a probabilistic process (Segal et al. 2003) that generates sequences of length $L_Y = |Y|$ by:

(a) Choosing, uniformly at random a number i between 1 and $(L_Y - w + 1)$,
(b) Sampling a site (of length w) from the motif,
(c) Planting this site starting at position i (and ending at position $i + w - 1$), and
(d) Sampling every other position (outside of $i \ldots i + w - 1$) from the background frequency distribution.Denoting the random variable indicating the start position of the planted site (step c) by i, we have the joint probability

$$p(Y,i) = \frac{1}{L_Y - w + 1} p(Y_i \mid \text{motif}) \prod_{j \notin \{i \ldots i+w-1\}} g_{Y_j},$$

where g_x is the background probability of base x. Summing this over all i to obtain $p(Y)$, and contrasting it with the likelihood under the null model, we get the likelihood ratio as

$$\frac{p(Y \mid \text{motif})}{p(Y \mid bg)} = \frac{1}{L_Y - w + 1} \sum_i \frac{p(Y_i \mid \text{motif})}{p(Y_i \mid bg)},$$

which equals (up to a constant factor) the score $T(Y) = \sum_i e^{S(Y_i)}$ suggested above.

A more comprehensive materialization of this idea is in the form of "Hidden Markov Model" (HMM) based methods. Such methods assume a "generative model" for regulatory sequences, and compute the likelihood of the sequence under the model. The generative model is a stochastic process with states corresponding to motifs for the different transcription factors that are expected to be involved in combinatorial regulation of the genes of interest (Fig. 7.2). The process visits the states probabilistically, and emits a sample of a motif whenever it visits the state corresponding to that motif (Fig. 7.2, *red arrows*). The emitted binding site is appended to the right end of the sequence generated thus far. A "background" state (Fig. 7.2, BKG) allows for these emitted binding sites to be interspersed with randomly chosen non-binding nucleotides. At any point, the process may transition to any state with some fixed probability called the "transition probability" of that state, which is a parameter of the model. (Fig. 7.2, p_1, p_2, p_3, p_b). Different implementations take different strategies to choosing values for these parameters. The sequence of states that the process visits is called a "path" of the HMM.

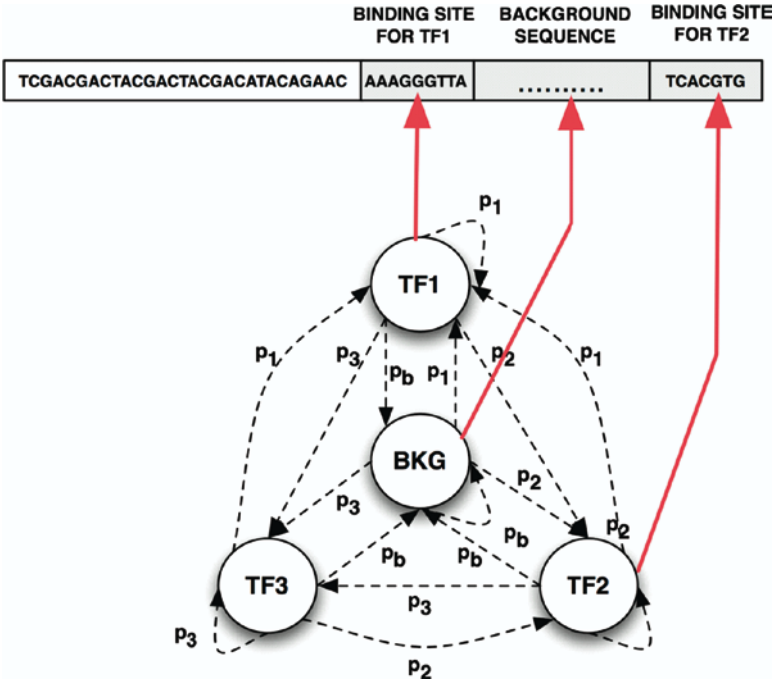

Fig. 7.2 Hidden Markov Model for CRM discovery

Any given sequence Y may be generated by many exponentially paths, and the joint probability $p(Y, \pi)$ of the sequence Y and a particular path π can be computed efficiently. The likelihood of the sequence Y is then computed by summing over all possible paths, i.e., $p(Y \mid \theta) = \sum_{\pi} p(Y, \pi \mid \theta)$, where θ denotes the parameters of the HMM. This summation can be performed efficiently using the algorithmic technique of "dynamic programming." The score of sequence Y is the log-ratio of this likelihood to the likelihood of Y being generated by a background model (that does not use the motifs), i.e., $T(Y) = \log \frac{p(Y \mid \theta)}{p(Y \mid \theta_b)}$ where θ_b denotes the parameters of the background model, and $p(Y|\theta_b)$ is the likelihood of this model.

7.4.2 *Evolutionary Comparisons for Regulatory Sequence Prediction*

A second major class of methods to improve the predictive power in the search for regulatory sequences is the incorporation of evolutionary information. The intuition here is that mutations in functional sequences will lead to a fitness defect, and individuals carrying them will be removed from the population by natural selection. Mutations in nonfunctional sequences will have no effect on fitness, and therefore may persist in the population and become fixed through genetic drift. Thus, over long evolutionary time, functional noncoding sequences will show few changes in their sequences, while nonfunctional sequences will evolve rapidly. This is the guiding principle of comparative genomics.

In order to apply comparative methods, the first step is to identify orthologous noncoding DNA sequences. There are many ways to accomplish this. In some cases simply searching closely related genomes for similar sequences can identify the orthologous noncoding regions. More sophisticated approaches include distance or tree-based methods to rule out paralogous regions, as well as considering the homology of nearby coding regions to ensure chromosomal synteny. Once orthologous noncoding sequences have been identified, these must be aligned, preferably using a DNA multiple aligner that performs a global alignment of the shorter sequence.

The technique of identifying evolutionarily conserved sequences in alignments has been called phylogenetic footprinting, to indicate the idea that functional constraint leaves a footprint of conservation in DNA sequences. Simple approaches to phylogenetic footprinting identify regions of alignments of noncoding regions above a certain percentage identity cutoff. Such comparative methods were first combined with matrix matching approaches by requiring that the matches to the matrix fall into "conserved" regions. These approaches have been demonstrated to greatly improve the power of motif matching methods by removing large numbers of false positives.

More elegant statistical approaches to phylogenetic footprinting employ explicit probabilistic models for the evolution of noncoding DNA. Based on the hypothesis

that functional sequences will evolve at a slower rate than surrounding regions, methods have been developed that explicitly compare the likelihood of each stretch of sequence under slow or fast evolutionary models. Because the sequences in the multiple alignments have evolved along an evolutionary tree, it is necessary to explicitly account for their phylogenetic relationships using an evolutionary model. This can be done by using a continuous-time Markov process to model the substitutions between the DNA bases, and a phylogenetic tree relating the sequences, in which the bases in the ancestral sequences are treated as unobserved or hidden variables. To compute the likelihood of such a model it is necessary to sum over all possible values of (or marginalize) the hidden variables (Felsenstein 1981).

Like the multinomial models for single sequences described above, probabilistic evolutionary models treat each position in the sequence independently, although rather than single bases at each position, the data are now columns in the multiple alignment. For example, for a pair wise alignment we have a tree with three nodes, the two sequences, X and Y and the unobserved ancestral sequence A. The likelihood of the pair wise alignment can be written as

$$L(X,Y) = \prod_{i=1}^{l} p(X_i, Y_i \mid R, T)$$

where the joint probability of the sequences can be written in terms of the unobserved ancestral residue as

$$p(X_i Y_i \mid R,T) = \sum_{A_i \in ACGT} p(X_i Y_i \mid A_i, R, T) p(A_i) = \sum_{A_i \in ACGT} p(X_i \mid A_i, R, T) p(Y_i \mid A_i, R, T) p(A_i)$$

where R represents the transition matrix of the continuous time Markov process, T represents the topology of the evolutionary tree (in this case the three nodes) and $p(A)$ are prior probabilities on the ancestral bases, and are usually assigned to be equal to the equilibrium distribution of the continuous time Markov process. An important feature of this model is that the evolution along each lineage (that leading to X and that leading Y) is independent, conditioned on the state of the ancestor A. To identify conserved sequences, one can form a likelihood ratio at each position between a "background" evolutionary model, say R_b, and a "conserved" evolutionary model where substitutions happen at a slower rate, R_c.

$$U(X_i, Y_i) = \frac{p(X_i, Y_i \mid R_c, T)}{p(X_i, Y_i \mid R_b, T)}$$

Extending this approach further, it is possible to posit a "hidden" state determining whether a piece of alignment is drawn from a conserved model or from the background model, and develop an HMM to identify conserved sequences. HMMs emitting columns of alignments rather than individual residues are often referred to as "phylo-HMMs" and are increasingly used in comparative genomics.

Finally, it is possible to combine probabilistic models for sequence evolution for the specificity and evolution of the transcription factor binding sites and the

Table 7.4 Comparative methods to identify regulatory sequences

Tool	Purpose	Notes
ConSite[a]	Identifies conserved matrix matches	Pairwise analysis only
VISTA[b]	Identifies conserved regions, matrix matches	Popular graphical display format
Footprinter Package[c]	Identifies conserved regions	Uses (i) binomial distribution and (ii) parsimony-based approaches to asses conservation in windows
eShadow[d]	Identifies conserved regions	Uses likelihood ratio tests
PhastCons[e]	Identifies conserved regions	Uses phyloHMM
MONKEY[f]	Identifies conserved matches to matrix models	Probabilistic model of binding site evolution, computes *P*-values

[a] (Sandelin et al. 2004b), [b] (Dubchak and Ryaboy, 2006), [c] (Blanchette and Tompa 2003), [d] (Ovcharenko et al. 2004), [e] (Siepel et al. 2005), [f] (Moses et al. 2004b)

background sequences. The critical step here is to assign the prior probabilities on the ancestral states to be the frequencies in the motif matrix. Classifiers based on these models can be constructed in much the same way as described above and have achieved much greater predictive power than approaches that match in single sequences.

Table 7.4 lists some of the implementations employing approaches utilizing comparative information.

7.5 De novo Motif-Finding

So far, we have assumed that the specificity of transcription factors was known, and the goal was to identify the regulatory regions or binding sites they controlled. However, in many cases, neither the sequence specificity, nor the binding sites of a transcription factor are known. Instead, the challenge is to infer the sequence specificity directly from a set of noncoding DNA sequences believed to contain binding sites. These methods rely on little biological information and are often referred to as "de novo" or "ab initio" because the computational method must identify the new motifs, starting from the beginning.

7.5.1 Statistical Overrepresentation of Consensus Sequence Motifs

The first approach to the ab initio discovery of transcription factor motifs assumes that the motifs are described by their consensus sequences (Sect. 2). There are a few commonly used variants of this motif model. In the simplest model, the motif is a string over the four letter alphabet {A, C, G, T}, and binding sites are required to be exact occurrences of the string (van Helden et al. 1998). In a second variant, the

binding sites are allowed to be at most one mismatch (Hamming distance 1) away from the motif sequence (Tompa 1999). A third commonly used model uses "degenerate symbols" such as "*R*" (which stands for "purine," i.e., "*A*" or "*G*") and "*Y*" (for pyrimidine, i.e., "*C*" or "*T*") in the motif alphabet, and binding sites have to be exact matches with any of the allowed nucleotides at the degenerate positions (Sinha and Tompa 2000). Typically, the motif is specified to be of a short (< 10 bp), fixed length k. Each of the above motif models clearly defines a "search space" of all possible motifs; e.g., in the first variant, the search space includes all 4^k strings of length k. It also lays out a prescription to count a motif's occurrences in any given sequence. Ab initio motif discovery in this framework then amounts to finding the one (or few) motif(s) in the search space that has the greatest statistical significance, as determined by their respective counts in the given set of sequences. We will next see a simple illustration of how such statistical significance may be determined.

Suppose we are given a DNA sequence S of length L_s, and a motif m. Let $N(S,m)$ denote the count of m in S. Next, consider a random sequence X, also of length L_s, that is generated by sampling one character at a time, as per the probability distribution $\{\pi_a, \pi_c, \pi_g, \pi_t\}$. The count $N(X,m)$ is therefore a random variable defined by the generative process (the "null model"), whose probability distribution will tell us about the statistical significance of m. Intuitively, if it is highly unlikely that a count of $N(S,m)$ or greater is observed in a random sequence, then we should interpret the motif m as being statistically overrepresented in S. Let us define p_m as the probability that motif m occurs at a specific position in X. In the simplest motif model of exact matches, this is given by

$$p_m = \prod_{i=1}^{k} \pi_{m_i}$$

Now, consider each of the positions $j = 1$ to $j = L_s - k + 1$, where the motif m may occur in the random sequence X. The probability of occurrence of m at position j is given by p_m, for all j. If we further assume that these events are independent, we have $L_s - k + 1$ independent and identically distributed ("i.i.d") *Bernoulli* trials with parameter p_m. Therefore, the number of occurrences of m in X follows a *Binomial* distribution with parameters $L_s - k + 1$ and p_m. That is, the P-value of the observed count $N(S,m)$ is given by

$$\sum_{n \geq N(S,m)} \binom{L_s - k + 1}{n} p_m^n (1 - p_m)^{L_s - k + 1 - n}$$

This is an estimate of the statistical significance of the motif m in sequence S (van Helden et al. 1998). The smaller the value, the greater the significance.

In the above calculation, we made a crucial assumption that the events of motif m occurring at different positions in a sequence are statistically independent. This is obviously a flawed assumption, since a motif's occurrence at a position j and the next position $j + 1$ (overlapping occurrences) are dependent variables: for a self-overlapping motif like "AAAAAA," occurrence at a position j implies a high probability of occurrence at the very next position $j + 1$, while for a motif such as

"ACGTTG," occurrence at j and $j+1$ are mutually exclusive events. We therefore turn our attention to a slightly different approach to evaluating statistical significance – through the use of "z-scores." We shall see how the flawed independence assumption is avoided in this new approach.

We assume the same null model as above, i.e., the random sequence X is generated by L_s i.i.d. samples from the probability distribution $\{\pi_a, \pi_c, \pi_g, \pi_t\}$. Let X_{mi} be an indicator random variable for the occurrence of motif m at position i. That is, this variable takes the value "1" if m occurs at position i in X, and "0" otherwise. Let X_m be the count of m in X. That is,

$$X_m = \sum_{i=1}^{L} X_{mi} \text{ where } L = L_s - k + 1 \tag{7.1}$$

Note that the observed count $N(S,m)$ is the value of this random variable X_m for the sequence S. Let $\mu_m = E(X_m)$ denote the expectation of this random variable, and σ_m denote its standard deviation, under the i.i.d null model. Then we define the z-score of the motif m as

$$z(S,m) = \frac{N(S,m) - \mu_m}{\sigma_m}$$

This is the number of standard deviations by which the observed count exceeds the expectation. A high value of this statistic indicates statistical significance. Our next task then is to compute the mean and standard deviation of X_m.

The expectation follows directly from (7.1). We note that the expectation of a sum is the sum of the expectations (the principle of "linearity of expectation"); hence we have

$$E(X_m) = E\left(\sum X_{mi}\right) = \sum E(X_{mi}) = \sum p(X_{mi} = 1) = Lp_m$$

Here, the third equality comes from the fact that the expectation of an indicator (0/1) variable is simply the probability of it being 1. Also, $p(X_{mi} = 1)$ is equal to p_m, as seen above. The standard deviation computation is slightly more complicated, but similar in spirit. Recalling that the variance is given by $\sigma_m^2 = E(X_m^2) - E(X_m)^2$, we need only to calculate $E(X_m^2)$, for which we have

$$E(X_m^2) = E\left(\left(\sum X_{mi}\right)^2\right) = E\left(\sum_{i,j} X_{mi}X_{mj}\right) = E\left(\sum_i X_{mi}^2 + 2\sum_i \sum_{j=i+1}^{i+k-1} X_{mi}X_{mj} + 2\sum_i \sum_{j=i+k}^{L-k+1} X_{mi}X_{mj}\right)$$

$$= E\left(\sum_i X_{mi}^2\right) + 2E\left(\sum_i \sum_{j=i+1}^{i+k-1} X_{mi}X_{mj}\right) + 2E\left(\sum_i \sum_{j=i+k}^{L-k+1} X_{mi}X_{mj}\right)$$

Note that the first term is simply $E\left(\sum_i X_{mi}\right) = E(X_m)$ since X_{mi} is an indicator variable.

The second term is (twice of) the expected number of occurrences, in a sequence of length L, of two overlapping sites matching the motif. This may be computed by

enumerating all strings of length $2k-1$ or less that have two overlapping occurrences of m, and adding their expectations, computed in the same way as $E(X_m)$. This term makes the variance depend on the self-overlapping structure of motif m. It is easy to see that among two motifs with the same p_m, and hence the same mean, if one has self-overlap and the other does not, the former will have the greater variance in its count. Finally, the third term amounts to (twice of)

$$E\left(\sum_i \sum_{j=i+k}^{L-k+1} X_{mi}X_{mj}\right) = \sum_i \sum_j E(X_{mi}X_{mj}) = \sum_i \sum_j p(X_{mi}=1, X_{mj}=1)$$
$$= \sum_i \sum_j p(X_{mi}=1)p(X_{mj}=1) = \sum_{i=1}^{L-2k+1} p(X_{mi}=1) \sum_{j=i+k}^{L-k+1} p(X_{mj}=1)$$
$$= \sum_{i=1}^{L-2k+1} p(X_{mi}=1)(L-2k-i+2)p_m = \frac{(L-2k+2)(L-2k+1)}{2} p_m^2$$

Here, the third equality follows from the fact that in an i.i.d. generated sequence, nonoverlapping occurrences are independent events. Note that if the null model is not i.i.d., and instead follows a higher order Markov chain (as is often the case), this independence assumption falls through, and other techniques are required to efficiently compute the third term.

The above calculations have been performed under several simplifying assumptions. In practice, the null model is often taken to be a second or third order Markov chain to capture adjacent nucleotide correlations that are present in real genomic sequences. The motif model typically allows for mismatches, so that the random variable X_m must represent counts under that model. Another complication arises from the fact that motif finding is often performed on both strands of the given sequence(s). Counting occurrences of the motif on both strands leads to additional statistical dependencies that must be handled. It is possible to extend the above calculations to account for all these complications in an efficient manner (Sinha and Tompa 2000).

7.5.2 De novo Motif Finding for the Matrix Representation

The classical probabilistic formulation of the motif finding problem posits that a biological sequence is made up of short subsequences, each of which may be an instance of the motif or drawn from a random background distribution (Table 7.5). The first models used in motif-finding were designed to solve the following problem. Given a set of sequences each containing one example of an unknown motif at an unknown location, find both the motif and the locations. From this perspective motif-finding was related to multiple alignments, such that the unknown position of the binding site was the point at which the sequences could be placed into ungapped multiple alignments.

Here we treat a slightly more general, but intuitively simpler version of this problem, where there is no constraint on the input sequences or the number of

Table 7.5 De novo motif finding methods

Tool	Purpose	Notes
RSAT[a]/YMF[b]	Consensus string based moti f-finder	Word statistics with enumeration of motif space
MobyDick[c]	Consensus string based motif-finder	Uses a segmentation algorithm to identify optimal "words"
MITRA[d]	Exhaustive consensus with mismatch search	Uses suffix tree
Weeder[e]	Exhaustive search with statistical ranking	Best performing algorithm in a systematic comparison (Tompa et al. 2005)
Gibbs Motif sampler[f]	Matrix-based de novo motif finder	Original Gibbs sampler
MEME[g]	Matrix-based de novo motif finder	Popular EM-based method, includes several models for the distribution of motifs in the input sequences
Consensus[h]	Matrix-based de novo motif finder	Information based method
NestedMICA[i]	Matrix-based de novo motif finder	Nested sampling method; no need for initial "seeding" step

[a](van Helden et al. 1998), [b](Sinha and Tompa 2000), [c](Bussemaker et al. 2000), [d](Eskin and Pevzner 2002), [e](Pavesi et al. 2004), [f](Lawrence et al. 1993), [g](Bailey and Elkan, 1994), [h](Stormo and Hartzell 1989), [i](Down and Hubbard 2005)

motifs in each input sequence. This implies a simple, two-component mixture model where each subsequence of length w is drawn from either the motif or background multinomials. In practice, modern de novo motif finders often provide several variations on the assumptions about the distribution of motifs in the input data.

If each subsequence is considered to be independent, the likelihood of the entire sequence under the mixture model can be written as the product of all the subsequences of length w,

$$L(X) = \prod_{i=1}^{l-w} p(X_i \mid \text{motif}) p(\text{motif}) + p(X_i \mid bg) p(bg)$$

where i indexes the position of the beginning of the subsequence relative to the input sequence X. Above, in the case of the multinomial model for a counts matrix, it is possible to maximize the likelihood directly and obtain the parameter estimates. However, it is not possible to obtain closed form solutions for the parameter estimates by directly differentiating the likelihoods of mixture models such as the one proposed above. Two major strategies have been employed for optimization, namely sampling approaches (here we consider Gibbs sampling) and Expectation-Maximization (EM), and we discuss each in turn.

7.5.2.1 Expectation-Maximization

The EM approach views the free parameters of the model as the unknown frequencies of each residue in the multinomial at each position in the motif. This means that for

a DNA motif of width w, there are $3w$ parameters, or a likelihood surface with $3w$ dimensions. The motif-finding problem is simply the problem of estimation of these parameters by maximizing the likelihood. However, because the positions of the motifs are unknown, the EM approach is to posit the existence of unobserved (or hidden) variables that specify at each position in the input sequence data, whether a particular position is an example of a binding site or not (Lawrence and Reilly 1990).

We represent these hidden variables as a vector at each position, $Z_i = (1,0)$ if the w-mer starting at position i is a binding site, and $Z_i = (0,1)$ if it is drawn from the background. To find parameter estimates, we assume that these hidden variables are observed, and then try to follow the maximization procedure above. Given the positions of the binding sites, we could write the "complete" likelihood:

$$L_c(X) = \prod_{i=1}^{l-w} \prod_{m \in \text{motif}, bg} [p(X_i \mid m)p(m)]^{z_{im}}$$

We therefore maximize this function as above. Taking logarithms yields,

$$\log[L_c(X)] = \sum_{i=1}^{l-w} \sum_{m \in \text{motif}} z_{im} \left[\left(\sum_{p=i}^{i+w} \sum_{b \in ACGT} X_{pb} \log[p(X_{pb} \mid m)] \right) + \log \pi_m \right]$$

We can now add Lagrange multipliers for the various constraints and differentiate with respect to the parameters as above. For example, for the frequency parameters we include the constraint $\sum_b f_{pb} = 1$, and obtain $\sum_{i=1}^{l-w} \frac{Z_{i0} X_{pb}}{f_{pb}} - \lambda = 0$, which after rearranging and substituting into the constraint, yields

$$f_{pb} = \frac{\sum_{i=1}^{l-w} Z_{i0} X_{pb}}{\sum_{i=1}^{l-w} Z_{i0}}$$

We now recall that this derivation was done under the assumption that we actually knew the positions of the binding sites in the input, i.e., that Z_i were observed. This is where the expectation step of the EM algorithm arises: we simply replace the Z_i with their expectations, based on our current estimates of the parameters. The expectation of a variable that takes on only 1 or 0 is simply the probability of non-zero outcome, so the expectations of these variables can be calculated using Bayes' theorem as above.

$$Z_{i0} \to E[Z_{i0}] = p(\text{motif} \mid X_i) = \frac{1}{1 + \frac{1-\pi}{\pi e^{S(X_i)}}}$$

Thus, the EM algorithm constitutes filling in these "posterior probabilities" of the unknown positions of the binding sites, recomputing the estimates of the parameters based on these, then recomputing the estimates of the hidden variables, etc., until convergence. This iterative strategy is guaranteed to increase the likelihood at

each step. Once the parameter estimates have stabilized, we can be confident that we have reached a local maximum in the likelihood. It is important to note, however, that this may not represent the global maximum in the likelihood. Furthermore we have not yet addressed the issue of where to obtain the initial estimates of the parameters to begin the EM procedure. In fact, these issues must be addressed in practice with heuristics based on intuition about the problem.

7.5.2.2 Gibbs Sampling

Sampling approaches posit a Markov chain whose equilibrium distribution is the posterior distribution of interest. This Markov chain starts with initial guess parameters, and then iteratively refines the guess. Once the chain has reached equilibrium, parameter estimates can be obtained by averaging over the states visited by the chain. Needless to say, the key to this procedure is how to define the transition probabilities in such chains; in other words, the rule for refining the guess. In general such approaches are called Markov-chain-Monte-Carlo or MCMC methods and have been considered elsewhere. Here we will focus on a particular type of MCMC algorithm that has been applied effectively to the de novo motif finding problem.

Gibbs Sampling is the procedure of sampling a new parameter estimate (or guess) according to the probability of the new guess conditioned on the current estimates of the remaining parameters. In our case the parameters are regarded as the unknown positions of the binding sites in the input data, and the estimates of the frequency matrix. For motif finding, therefore, at each iteration, one of the current binding site positions is selected at random to be replaced. The frequency matrix is recalculated leaving out the selected binding site and every available position in the input data is re-evaluated by computing the statistic $S(X_i)$ described above at each position.

While the derivation for the exact equations for the Gibbs Sampler is too complicated to reproduce here, it can be shown (Liu et al. 1995) that the probability required for the Gibbs Sampler is given approximately by

$$p(\text{new site at } i \mid X_i) = \frac{S(X_i)}{\sum_j S(X_j)}$$

Thus, a new binding site is then chosen by choosing randomly from the available positions with probability proportional to S. Interestingly, the new binding site does not necessarily improve the likelihood: the site at p might have a lower likelihood by chance than the one it was replacing. However, on average this procedure will tend to sample binding sites that are near the current motif. Critically, the "tighter" or more information contained in the motif, the more likely the sampling procedure is to sample "near" it in sequence space. Thus, although the Gibbs Sampler will explore the entire space, it will be strongly biased to sample near maxima in the likelihood; the higher these maxima, the stronger this bias.

7.6 Second Generation Motif-Finding Methods

In addition to the computational difficulty of finding the local maxima in a high-dimensional likelihood space, several lines of evidence suggested that there is not enough information in some motifs to identify them in large regions of noncoding DNA of the eukaryotic genomes that are becoming available. The motif-finding approaches described above are general; they do not take into account specific properties of the transcription factor binding site finding problem. Recently, new motif finding methodology has been developed that used similar computational techniques and models, but included additional data about sequence specific transcription factor binding sites in the motif finding.

7.6.1 Associations with Functional Genomics Data

The first methods explicitly designed to identify motifs in noncoding DNA used additional information about which genes were likely to be regulated by a transcription factor. The simplest of these cases is simply where motif finding is done on two sets of sequences, those likely to be regulated and those unlikely to be regulated. This information can be as simple as the functional classification of a gene. Searching for motifs that separate two sets of noncoding regions can be thought of as a statistical discrimination problem, and many statistical methods can be applied.

It is possible to increase the sophistication of such discriminative methods, such that the motifs are taken as explanatory variables for quantitative, possibly multivariate data. For example, genome-wide transcription factor binding data can provide a ranked list of genes that are most likely (and least likely) to be bound by a transcription factor. Motif-finding methods can exploit this information by searching for patterns that are statistically associated with these rankings. Similarly, genome-wide gene expression data can give information about which genes change expression in response to developmental changes or to the environment. If a transcription factor leads to a change in expression of transcripts in a particular condition, the expression of all (or many) of the genes containing the motif are expected to change. Therefore, motifs can be identified based on whether they can explain the variance in genome-wide gene expression data using regression and other statistical methods.

7.6.2 Incorporating Comparative Information into de novo Motif Finding

Another important group of next generation de novo motif-finders are comparative or phylogenetic methods. With the availability of complete genome sequences for closely related organisms, including comparative sequence information in motif

finding was a natural extension. The first methods to use comparative information did so using heuristics to encapsulate the notion that motifs should be conserved in alignments of homologous sequences. Soon after, motif-finders that incorporated explicit probabilistic models of evolution into motif finding were developed.

7.6.3 *Other Methods and Future Directions*

Another interesting strategy for motif finding is to use prior information about the pattern of information content in real transcription factor binding sites. This essentially attempts to reduce the number of nonbiological maxima in the likelihood function, by biasing the search away from regions of the motif space that are unlikely to represent real biological motifs (Table 7.6). In general, methods that identify additional biological features of motifs in eternal datasets, or in sequence data will be of continued interest in the short term.

Table 7.6 De novo motif-finders that incorporate additional information

Tool	Purpose	Notes
SeedSearch[a]	Consensus-based discriminative approach	Hypergeometric statistics
DME[b]	Identifies motifs overrepresented in one set of sequences relative to a background set	Matrix-based, enumerative discriminative approach
DRIM[c]	Identifies motifs in a ranked list of sequences	Consensus-based, enumerative hypergeometric statistics with corrections
REDUCE[d]	Identifies motifs correlated with gene expression	Uses multiple regression
GMEP[e]	Identifyies motifs associated with gene expression data	Uses z-scores
Footprinter[f]	Identifies conserved motifs in orthologous noncoding sequences	Parsimony based approach
Kellis et al.[g]	Identifies conserved motifs in genome-wide alignments	No popular implementation, computes conservation of "mini"-motifs, and combines these into larger motifs
EMnEM[h]/PhyME[i]	Identifies conserved motifs in orthologous noncoding sequences	Phylogenetic E–M based approach
PhyloGibbs[j]	Identifies conserved motifs in orthologous noncoding sequence	Phylogenetic sampling based approach. Relaxes assumption of complete conservation
TFEM[k]	Identifies motifs with particular information content profiles	Adds additional constraints to traditional E–M maximization

[a](Barash et al. 2001), [b](Smith et al. 2005), [c](Eden et al. 2007), [d](Bussemaker et al. 2001), [e](Chiang et al. 2001), [f](Blanchette and Tompa 2003), [g](Kellis et al. 2004), [h](Moses et al. 2004a), [i](Sinha et al. 2004), [j](Siddharthan et al. 2005), [k](Kechris et al. 2004)

As more diverse data become available, computation systems that combine diverse data types, as well as the pattern recognition methods described in this chapter will become increasingly powerful. Indeed, recent methods have attempted to construct regulatory networks using model-based approaches to synthesize motif finding and analysis of functional genomics data (e.g., Segal et al. 2003). Computational methods will undoubtedly have an exciting role to play as we advance toward the goal of predicting gene expression from sequence (Segal et al. 2008).

References

Aerts S, Haeussler M, van Vooren S, Griffith OL, Hulpiau P, Jones SJ et al (2008) Text-mining assisted regulatory annotation. Genome Biol 9(2):R31

Bailey TL, Elkan C (1994) Fitting a mixture model by expectation maximization to discover motifs in biopolymers. Proc Int Conf Intell Syst Mol Biol 2:28–36

Bailey TL, Gribskov M (1998) Methods and statistics for combining motif match scores. J Comput Biol 5(2):211–221

Barash Y, Bejerano G, Friedman N (2001) A simple hyper-geometric approach for discovering putative transcription factor binding sites. Proceedings of the first international workshop on algorithms in bioinformatics, Springer

Benjamini Y, Hochberg Y (1995) Controlling the false discovery rate: A practical and powerful approach to multiple testing. J Royal Stat Soc B 57(1):289–300

Berg OG, von Hippel PH (1987) Selection of DNA binding sites by regulatory proteins. Statistical-mechanical theory and application to operators and promoters. J Mol Biol 193(4):723–750

Bergman CM, Carlson JW, Celniker SE (2005) Drosophila DNase I footprint database: A systematic genome annotation of transcription factor binding sites in the fruitfly, Drosophila melanogaster. Bioinformatics 21(8):1747–1749

Berman BP, Nibu Y, Pfeiffer BD, Tomancak P, Celniker SE, Levine M et al (2002) Exploiting transcription factor binding site clustering to identify *cis*-regulatory modules involved in pattern formation in the Drosophila genome. Proc Natl Acad Sci USA 99(2):757–762

Blanchette M, Tompa M (2003) FootPrinter: A program designed for phylogenetic footprinting. Nucleic Acids Res 31(13):3840–3842

Bussemaker HJ, Li H, Siggia ED (2000) Building a dictionary for genomes: Identification of presumptive regulatory sites by statistical analysis. Proc Natl Acad Sci USA 97(18):10096–10100

Bussemaker HJ, Li H, Siggia ED (2001) Regulatory element detection using correlation with expression. Nat Genet 27(2):167–171

Chiang DY, Brown PO, Eisen MB (2001) Visualizing associations between genome sequences and gene expression data using genome-mean expression profiles. Bioinformatics 17(Suppl 1):S49–S55

Down TA, Hubbard TJ (2005) NestedMICA: Sensitive inference of over-represented motifs in nucleic acid sequence. Nucleic Acids Res 33(5):1445–1453

Dubchak I, Ryaboy DV (2006) VISTA family of computational tools for comparative analysis of DNA sequences and whole genomes. Methods Mol Biol 338:69–89

Durbin R, Eddy SR, Krogh A, Mitchison GJ (1998) Biological sequence analysis: Probalistic models of proteins and nucleic acids. Cambridge University Press, Cambridge, UK

Eden E, Lipson D, Yogev S, Yakhini Z (2007) Discovering motifs in ranked lists of DNA sequences. PLoS Comput Biol 3(3):e39

Eskin E, Pevzner PA (2002) Finding composite regulatory patterns in DNA sequences. Bioinformatics 18(Suppl 1):S354–S363

Felsenstein J (1981) Evolutionary trees from DNA sequences: A maximum likelihood approach. J Mol Evol 17(6):368–376

Frith MC, Li MC, Weng Z (2003) Cluster-Buster: Finding dense clusters of motifs in DNA sequences. Nucleic Acids Res 31(13):3666–3668

Gadiraju S, Vyhlidal CA, Leeder JS, Rogan PK (2003) Genome-wide prediction, display and refinement of binding sites with information theory-based models. BMC Bioinformatics 4:38

Gallo SM, Li L, Hu Z, Halfon MS (2006) REDfly: A regulatory element database for Drosophila. Bioinformatics 22(3):381–383

Halfon MS, Grad Y, Church GM, Michelson AM (2002) Computation-based discovery of related transcriptional regulatory modules and motifs using an experimentally validated combinatorial model. Genome Res 12(7):1019–1028

Heinemeyer T, Wingender E, Reuter I, Hermjakob H, Kel AE, Kel OV et al (1998) Databases on transcriptional regulation: TRANSFAC, TRRD and COMPEL. Nucleic Acids Res 26(1): 362–367

Hertz GZ, Stormo GD (1999) Identifying DNA and protein patterns with statistically significant alignments of multiple sequences. Bioinformatics 15(7–8):563–577

Johnston M, Stormo GD (2003) Evolution. Heirlooms in the attic. Science 302(5647):997–999

Kechris KJ, van Zwet E, Bickel PJ, Eisen MB (2004) Detecting DNA regulatory motifs by incorporating positional trends in information content. Genome Biol 5(7):R50

Kellis M, Patterson N, Birren B, Berger B, Lander ES (2004) Methods in comparative genomics: Genome correspondence, gene identification and regulatory motif discovery. J Comput Biol 11(2–3):319–355

Kullback S, Leible RA (1951) On information and sufficiency. Ann Math Stat 22(1):79–86

Lander ES, Linton LM, Birren B, Nusbaum C, Zody MC, Baldwin J et al (2001) Initial sequencing and analysis of the human genome. Nature 409(6822):860–921

Lawrence CE, Altschul SF, Boguski MS, Liu JS, Neuwald AF, Wootton JC (1993) Detecting subtle sequence signals: A Gibbs sampling strategy for multiple alignment. Science 262(5131): 208–214

Lawrence CE, Reilly AA (1990) An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences. Proteins 7(1):41–51

Levine M, Davidson EH (2005) Gene regulatory networks for development. Proc Natl Acad Sci USA 102(14):4936–4942

Lifanov AP, Makeev VJ, Nazina AG, Papatsenko DA (2003) Homotypic regulatory clusters in Drosophila. Genome Res 13(4):579–588

Liu JS, Neuwald AF, Lawrence CE (1995) Bayesian models for multiple local sequence alignment and Gibbs sampling strategies. J Am Stat Assoc 90(432):1156–1170

Mannervik M, Nibu Y, Zhang H, Levine M (1999) Transcriptional coregulators in development. Science 284(5414):606–609

Markstein M, Levine M (2002) Decoding *cis*-regulatory DNAs in the Drosophila genome. Curr Opin Genet Dev 12(5):601–606

Montgomery SB, Griffith OL, Sleumer MC, Bergman CM, Bilenky M, Pleasance ED et al (2006) ORegAnno: An open access database and curation system for literature-derived promoters, transcription factor binding sites and regulatory variation. Bioinformatics 22(5):637–640

Moses AM, Chiang DY, Eisen MB (2004a) Phylogenetic motif detection by expectation-maximization on evolutionary mixtures. Pac Symp Biocomput:324–335

Moses AM, Chiang DY, Pollard DA, Iyer VN, Eisen MB (2004b) MONKEY: Identifying conserved transcription-factor binding sites in multiple alignments using a binding site-specific evolutionary model. Genome Biol 5(12):R98

Münch R, Hiller K, Barg H, Heldt D, Linz S, Wingender E et al (2003) PRODORIC: Prokaryotic database of gene regulation. Nucleic Acids Res 31(1):266–269

Ovcharenko I, Boffelli D, Loots GG (2004) eShadow: A tool for comparing closely related sequences. Genome Res 14(6):1191–1198

Pavesi G, Mereghetti P, Mauri G, Pesole G (2004) Weeder Web: Discovery of transcription factor binding sites in a set of sequences from co-regulated genes. Nucleic Acids Res 32(Web Server issue):W199–W203

Sandelin A, Alkema W, Engstrom P, Wasserman WW, Lenhard B (2004a) JASPAR: An open-access database for eukaryotic transcription factor binding profiles. Nucleic Acids Res 32(Database issue):D91–D94

Sandelin A, Wasserman WW, Lenhard B (2004b) ConSite: Web-based prediction of regulatory elements using cross-species comparison. Nucleic Acids Res 32(Web Server issue): W249–W252

Schneider TD, Stormo GD, Gold L, Ehrenfeucht A (1986) Information content of binding sites on nucleotide sequences. J Mol Biol 188(3):415–431

Segal E, Raveh-Sadka T, Schroeder M, Unnerstall U, Gaul U (2008) Predicting expression patterns from regulatory sequence in Drosophila segmentation. Nature 451(7178):535–540

Segal E, Yelensky R, Koller D (2003) Genome-wide discovery of transcriptional modules from DNA sequence and gene expression. Bioinformatics 19(Suppl 1):i273–i282

Siddharthan R, Siggia ED, van Nimwegen E (2005) PhyloGibbs: A Gibbs sampling motif finder that incorporates phylogeny. PLoS Comput Biol 1(7):e67

Siepel A, Bejerano G, Pedersen JS, Hinrichs AS, Hou M, Rosenbloom K et al (2005) Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. Genome Res 15(8):1034–1050

Sinha S, Blanchette M, Tompa M (2004) PhyME: A probabilistic algorithm for finding motifs in sets of orthologous sequences. BMC Bioinformatics 5:170

Sinha S, Liang Y, Siggia E (2006) Stubb: A program for discovery and analysis of *cis*-regulatory modules. Nucleic Acids Res 34(Web Server issue):W555–W559

Sinha S, Tompa M (2000) A statistical method for finding transcription factor binding sites. Proc Int Conf Intell Syst Mol Biol 8:344–354

Smith AD, Sumazin P, Zhang MQ (2005) Identifying tissue-selective transcription factor binding sites in vertebrate promoters. Proc Natl Acad Sci USA 102(5):1560–1565

Staden R (1989) Methods for calculating the probabilities of finding patterns in sequences. Comput Appl Biosci 5(2):89–96

Stormo GD (2000) DNA binding sites: Representation and discovery. Bioinformatics 16(1):16–23

Stormo GD, Hartzell GW III (1989) Identifying protein-binding sites from unaligned DNA fragments. Proc Natl Acad Sci USA 86(4):1183–1187

Tompa M (1999) An exact method for finding short motifs in sequences, with application to the ribosome binding site problem. Proc Int Conf Intell Syst Mol Biol:262–271

Tompa M, Li N, Bailey TL, Church GM, De Moor B, Eskin E et al (2005) Assessing computational tools for the discovery of transcription factor binding sites. Nat Biotechnol 23(1):137–144

van Helden J, Andre B, Collado-Vides J (1998) Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. J Mol Biol 281(5):827–842

Venter JC, Adams MD, Myers EW, Li PW, Mural RJ, Sutton GG et al (2001) The sequence of the human genome. Science 291(5507):1304–1351

Wasserman WW, Fickett JW (1998) Identification of regulatory regions which confer muscle-specific gene expression. J Mol Biol 278(1):167–181

Wasserman WW, Sandelin A (2004) Applied bioinformatics for the identification of regulatory elements. Nat Rev Genet 5(4):276–287

Waterston RH, Lindblad-Toh K, Birney E, Rogers J, Abril JF, Agarwal P et al (2002) Initial sequencing and comparative analysis of the mouse genome. Nature 420(6915):520–562

Wingender E, Dietze P, Karas H, Knuppel R (1996) TRANSFAC: A database on transcription factors and their DNA binding sites. Nucleic Acids Res 24(1):238–241

Zhu J, Zhang MQ (1999) SCPD: A promoter database of the yeast Saccharomyces cerevisiae. Bioinformatics 15(7–8):607–611