Chapter 10: Computational Methods for Promoter Recognition

Michael Q. Zhang
Associate Professor
Watson School of Biological Sciences
Cold Spring Harbor Laboratory
1 Bungtown Road, NY 11724
mzhang@cshl.org

10.1 Introduction

In this chapter, we shall describe the problem of promoter recognition. We begin with a brief introduction of the biology of promoter structure and function. We then review some of the current computational approaches to the problem, with emphasis on basic concepts and methodologies in real applications. Interested readers should consult the references for more technical details or program specifications.

There are two main classes of functional information encoded in the genomic DNA of every living organism. One class is the coding regions, which specify the structure and function of each gene product; another class is the regulatory regions (occasionally, but very rarely, overlapping with a
coding region), which control and regulate when, where and how the genes are expressed. Promoter is the most important regulatory region that controls and regulates the very first step of gene expression: mRNA transcription. For a comprehensive review on the related biology; one is referred to the excellent book “Transcriptional Regulation in Eukaryotes” by Carey and Smale (1999).

Promoter is commonly referred to as the DNA region that is required to control and regulate the transcriptional initiation of the immediately downstream gene. For a typical eukaryotic (polII or protein-coding) gene, it contains a core promoter about 100 bp centered around the transcriptional start site (TSS) and a proximal promoter about 500 bp immediately upstream of the core promoter. Often complex regulation in vivo can involve many more features, such as enhancers, locus control regions (LCRs), and/or scaffold/matrix attachment regions (S/MARs). Some people refer enhancers as the distal promoter elements, which can be either upstream or
downstream of the gene or within an intron and can be in any orientations. For our purpose, we use the region (-500,+100) with respect to a TSS as a specific definition.

The main characteristic of a promoter is it contains aggregates of transcription factor (TF) binding sites. During the process of development, genes are turned on and off in a pre-programmed fashion, a process that eventually generates cell specificity. This developmental program is orchestrated by TFs, which bind to specific DNA sites in the promoters near genes they control. A single TF is not dedicated to each regulatory event. Instead, different combinations of ubiquitous and cell-specific regulatory factors are used to achieve a combinatorial control.

(Figure 10.1)

Core promoter, approximately in (-50,+50), (a) binds to and controls assembly of the preinitiation complex (PIC) containing Pol II, the general transcription factor (GTF), and coactivators; (b)
positions the TSS and controls the direction of transcription; and (c) responds to nearby or distal activators (we use the same terms “activators” or “enhancer” to also imply “repressors” and “silencers”, depending on the context, for simplicity) binding proximal promoter and enhancers. The PIC comprises the GFTs (Pol II, TFIIA, TFIIB, TFIID, TFIIE, TFIIF and TFIIF) and coactivators that mediate response to regulatory signals. A typical core promoter contains four DNA elements (Fig.10.1): TATA-box (binding site for the TBP subunit of TFIID. Although recently some alternative TATA binding proteins have been discovered in a few specific types of cells, see Holmes and Tjian 2000, this review deals with only the major type of TBP); Inr (overlapping with TSS); DPE (downstream core promoter element) and BRE (TFIIB recognition element). Not every element occurs in a core promoter, people have classified core promoters according to the presence or the absence of TATA and/or Inr elements (reviewed in Novina and Roy 1996). Many “housekeeping gene” core promoters appear to lack both TATA and Inr
elements but instead contain several TSSs, a high G+C content, and multiple binding sites for the ubiquitous TF Sp1 (Smale 1994). Sp1 directs the formation of PIC to a region 40 to 100 bp downstream of its binding sites. Purified GTFs and Pol II mediate basal (low level) transcription on a core promoter in vitro but cannot support activated transcription in the absence of coactivators. More recent studies indicate that the functional form of PIC in vivo must also include coactivators/mediators, the interaction of activators with any surface of this large GTF-containing complex (also called holoenzyme, reviewed in Parvin and Young 1998) allows recruitment of the complex to the core promoter and response of the polymerase to the regulatory signals.

((Figure 10.2))

Transcriptional regulation is controlled by the binding of sequence-specific DNA-binding TFs to proximal promoters, approximately in (-500,-50) (also called regulatory promoters), and enhancers (reviewed in Blackwood and Kadonaga 1998). It should be noticed
that there is no real distinction between proximal and distal (enhancer) regulatory elements, they often involve the same set of TF binding sites. Some cooperative binding of activators to enhancers and proximal promoters can lead to assembly of nucleoprotein structures termed “enhanceosomes” (Fig.10.2, see Thanos and Maniatis 1995).

In a living eukaryotic cell, DNA is not naked, instead it is wrapped into nucleosomes by histones. With the help of many other non-histone proteins (NHP), nucleosomes are further condensed into chromatin filament. These higher order structures are believed to be necessary to keep most of genes in a (default) repressed state. To activate a gene, the chromatin encompassing that gene and its control regions must be altered or “remodeled” to permit TFs to access their specific binding sites. Because of the complexity of such long-range interaction among many global regulators, chromatin remodeling is beyond the reach of current promoter recognition algorithms. Therefore, all existing computational methods
implicitly assume all TF sites are accessible which is the intrinsic source of a large number of false positives.

In summary, promoter is the key DNA region that control and regulate transcription. Delineation of the promoter architecture is fundamental for understanding gene expression pattern, regulation networks, cell specificity and development. It is also important for designing efficient expression vector or target specific delivery system in gene therapy. In the large-scale genomic sequencing era, promoter prediction is also crucial for gene discovery and annotation.

There have been many computational approaches to this extremely difficult problem. For a starter, I recommend some recent reviews (e.g. Pedersen et al. 1999; Werner 1999; Fickett and Wasserman 2000; Stormo 2000) where one could find further references. Depending on the goals, computational approaches can be divided into two classes: general promoter recognition methods and specific promoter recognition
methods. The primary goal for the general methods is to identify TSS and/or core promoter elements for all genes in a genome; while the specific methods focus on identifying specific regulatory elements (TF sites) that are shared by a particular set of transcriptionally related genes. Specific methods can have very high specificity when searching against the whole genome and can provide immediate functional clues to the downstream gene. But because of the broad coverage, the general methods are extremely useful for large-scale genome annotation. I shall first describe the specific promoter recognition problem, which is essentially how to find functional TF sites. I shall then take on the general problem, which is essentially how to discriminate a promoter region from other genomic regions.

10.2 Finding transcription factor (TF) binding sites

10.2.1 Site, consensus and weight matrix

As a specific promoter class is characterized by a specific set of TFs, finding TF binding sites is the most important step in promoter recognition. There are
at least two classes of TFs (From now on, we use TF to refer DNA binding transcription factors): one class is the general or ubiquitous TFs, such as TATA-box binding protein (TBP) or Sp1, their binding sites can be identified by simply collecting a large number of promoter (-500,+50) sequences; another class is the specific TFs, they can only be identified by getting a specific set of promoters that share the same site (i.e. their target genes are co-regulated by the same TF). Experimentally, biologists are able to identify a TF site de novo with a single promoter sequence. They can characterize such a site by mutagensis and obtain a consensus description (such as the E. coli TATA-box TATAAT or allowing degeneracy TATRNT). As more sites are known, one can get the same information by aligning the sites. While it is easy to write a consensus pattern to represent aligned sites, it is difficult to find one that is optimal for predicting the occurrence of new sites (generalizability) or for discriminantly ranking the binding activities of the
sites (differentiability). In most applications, a position **weight matrix** (PWM) is often more superior.

### 10.2.2 Constructing a matrix given the alignment

To explain high degeneracy found in many sites, Berg and von Hippel (1987) proposed a theory on selection of DNA binding sites by regulatory proteins. They assumed that specific sites have been selected according to some functional constraint (e.g. the binding affinity or activity must be in some tolerable range), and all sequences that can fulfill this requirement are equally likely to occur. This theory, which has been checked by experiments, provided a link between a natural scoring function (minus binding energy) and observed base frequency at each position:

\[
S_{bx} = \log \left( \frac{p_{bx}}{p_{0x}} \right),
\]

where \( S_{bx} \) is the score (PWM element) for a base “b” at position “x”, \( p_{bx} \) is the frequency the base “b” is found at “x” in the site and “0” indicates the base (consensus base) which has the lowest energy. Often it is desirable to use a scoring function that can best discriminate the set of target sites from a set of
control sites, in this situation, we replace $p_{0x}$ by $p_{0b}^*$, the base frequency evaluated in the control set (here, we assume they do not depend on the position). Choosing $f_{0b}^*$ appropriately (representing the correct background contrast) can be very important for searching other such sites in a genome. The total score of the site is the sum of individual position scores. Using single base frequency implies the assumption of independence between any pair of bases, although this can be easily generalized to high order Markov models (e.g. Zhang and Marr 1993).

It should be mentioned that, to construct a statistical measure, the quality of sequences is obviously extremely crucial. In addition to the integrity of the data, statistical independence of the sequences is also essential so that the result will not be biased by the sample. There are many ways to reduce the redundancy; a simple criterion may be that no pair of sequences should have more than 90% identity within the 100 bp surrounding region.
There are many ways to introduce pseudo-counts in order to avoid a null frequency. This is equivalent to introduce a prior probability, which is necessary when an observed count is rare. The Laplace plus-one method (i.e. add one to every base count at every position, and hence correspond to a uniform prior) is a simple and popular choice.

How to determine the length of a motif can be a very subtle problem. Conventionally, one uses the relative information to measure the significance of each position (Schneider et al. 1986):

\[
I_x = \Sigma_b p_{bx} \log \left( \frac{p_{bx}}{p_{0b}} \right) = \Sigma_b p_{bx} s_{bx}, \quad (10.2)
\]

which obviously has the meaning of the average binding energy of all the known sites at position “x”. The optimal length of the motif can be obtained by an optimization procedure. One procedure, consistent with Berg & von Hippel theory, would be to find an optimal window site so that the total information within the window (the area under the curve of \(I_x\)) minus the average of the total information in the two flanking
window become maximum. It can be further refined by a discriminant procedure described below.

10.2.3 Searching for a known binding site

Given a motif, either in the form of a consensus or a matrix, one must first assess the quality of the motif and determine a threshold value before one can use it for searching new members of the site. The way to do this is to perform a standard classification test (e.g. Fukunaga 1990) in which both the threshold score and the motif length may be optimized by minimizing the classification (Bayesian) error. Because a single TF site does not have enough specific information due to its short length (about 5 ∼ 25 bp) and high degeneracy, any unconstraint genome search will almost certainly result in a lot of false positives. The specificity can only be achieved by combining interactive (correlated) sites into a promoter module (also called a composite site) and by higher order structure constraints (long range control elements).

10.3 Identifying motifs with unaligned sequences
In order to discover novel motif sites, one has to use more sophisticated approaches. Given a set of related sequences, these methods must be able to find motif(s), which are shared by majority of the sequences and are statistically significant, in a reasonable time. Although we are focusing on TF sites, all methods should be applicable to more general sequence motif discovery problems. There are existing numerous algorithms (see reviews, e.g. Vanet et al 1999; Brazma et al. 1998; Pesole et al. 1996; French et al. 1997), below is necessarily a personal selection which represents a short list of generic methodologies. HMM (see e.g. Durbin et al) and neural network (see e.g. Baldi and Brunak 1998) are not included because there are special books describing these machine learning approaches.

10.3.1 K-tuple or exhaustive pattern search methods

For finding short and highly conserved motifs (such as many typical TF sites in yeast), k-tuple based methods can be very effective. The basic idea is
to detect over-represented (with respect to a control set or to a background set) \(k\)-tuples systematically.

10.3.1.1 Relative Information (RI)

The simplest is to calculate the frequencies for all \(k\)-tuples in both the target set and the control set. \(k\) is usually limited by the size of the sample data \(4^k \leq (L-k+1)N\), where \(L\) is the length of each sequence and \(N\) is the number of sequences). One can define a RI for every word \(w\) of length \(k\) as (also called LOG-ODD Ratio, or negative relative entropy):

\[
RI(w) = \log \left( \frac{f(w)}{f^0(w)} \right).
\]

Although, the exact statistics is not known, one can use \(z = (RI - \text{mean}(RI))/\text{std}(RI)\) to estimate the significance. If \(f^0\) represent a random background (usually a Markov approximation of order less than \(k\)), one could use chi-square test on \(\chi^2 = \sum_w \left[ O(w) - E(w) \right]^2/E(w)\) where \(O(w)\) is the observed number of \(w\) and \(E(w)\) is the expected number (calculated using \(f^0\)). This has been applied in yeast promoter analysis (Zhang 1999, and see 10.3.5 below). This method can be easily generalized to allow limited degeneracy and/or
iterative extension of the k-tuple motif (see Zhu and Zhang 2000, where a motif pattern was defined as a 6-tuple allowing up to one mismatch and an iterative procedure for extending such a motif using the $\chi^2$-test).

10.3.1.2 WORDUP

This is a similar method but requiring the motif to be shared by majority of the sequences (Pesole et al 1992). The statistical significance of each k-tuple word $w$ is determined by comparing, through a $\chi^2$-test, the actual number of different sequences in which $w$ is present with the expected occurrences. Expectations are calculated on the basis of two assumptions: (a) oligonucleotides are Poisson distributed, and (b) nucleotide sequences can be generated according to a first-order Markov chain. Since the probability $p_i(w)$ that $w$ is found at least once in the $i$th sequence is $p_i(w) = 1 - \exp[-\lambda_i(w)]$ with $\lambda_i(w) = p^0_i(w) (L_i-k+1)$ and $p^0_i(w)$ is the Markov approximation of $f_i(w)$, namely $p^0_i(w) = f(w_{1,2}) f(w_{2,3}) \ldots f(w_{k-1,k})/f(w_{2}) f(w_{3}) \ldots f(w_{k-1})$. The expected number of sequences containing $w$ is given by $E(w) = \Sigma_i p_i(w)$. If $O(w)$ is the observed number of sequences containing $w$, the standard $\chi^2$-value given above can be used to rank significant k-tuples (the default threshold is 20) which form a vocabulary. An iterative procedure was also used to construct a new vocabulary containing all significant words of length greater or equal to $k$. 
A similar method was developed by van Helden et al (1998) using slightly different statistical criterion. More sophisticated algorithms for detecting more complex patterns (with multiple sites) are also developed recently (Marsan and Sagot 2000).

10.3.2 Multiple sequence alignment methods

For longer and more degenerate motifs, one has to use multiple sequence (local) alignment algorithms. Given N (number of sequences), L (length of each sequence) and k (length of the motif with indels), there are \((L-k+1)^N\) possible alignments. Finding an optimal alignment which maximizing an objective function (say, \(I = \Sigma_x I_x\)) is a hard problem. Various heuristic approaches are available to attack this multi-dimensional optimization problem. I describe here three generic methods: CONSENSUS (a greedy algorithm), EM/MEME (EM algorithms) and Gibbs sampler (a stochastic sampling algorithm)

10.3.2.1 CONSENSUS
A greedy algorithm originally developed by Stormo & Hartzell III (1988) and implemented in CONSENSUS (Hertz et al. 1990) is a heuristic method, which is quite efficient and has been widely used in DNA motif discoveries.

(((Figure 10.3)))

The basic idea is illustrated in a toy example shown in Fig.10.3. Given the three sequences (A) to be aligned, the algorithm starts by forming a frequency matrix for each of the $k$-tuple in the first sequence (B). Each of these matrices is then combined with each $k$-tuple in the second sequence to form new matrices containing two $k$-tuples (C). However, for each $k$-tuple from the first sequence, the program only saves the “best” progeny matrix (measured by the information content). In the next cycle, each saved matrix is combined with each $k$-tuple in the third sequence to form new matrices each containing three $k$-tuples (D). Again, the program only saves the best progeny of each matrix from the previous cycle. This cycle is repeated until the last sequence in the set has contributed a
k-tuple to the saved matrices. Of the matrices saved after the last cycle, the one with the lowest probability of occurring by chance is considered to describe the consensus motif (the first in D). In practice, ties occur during the cycles so that the number of matrices at the end is greater than the number of k-tuples in the first sequence.

The CONSENSUS program was first used to accurately identify the known consensus pattern for the E.coli CRP protein binding sites (Stormo & Hartzell 1989). It was then further improved and tested for robustness on the E.coli LexA protein binding sites (Hertz et al. 1990). In both cases, the order in which the sequences are presented to the program is not critical (the latest version of the program allows the user to set a parameter so that the result will not depend on the input order at all). The program is also robust enough to tolerate some sequences that do not contain binding sites. Thanks for the effort of Hertz, CONSENSUS has been constantly improved upon. Some of the important additions to the original algorithm are (a)
independence of the input sequence order, (b) auto-detecting motif length, (c) allowing limited insertions/deletions, (d) more rigorous statistical evaluation of the p-value (Hertz & Stormo 1999).

\section*{10.3.2.2 EM and MEME}

EM (Expectation Maximization) is a standard technique widely used in maximum-likelihood estimations (Dempster et al 1977). EM algorithms are named for their two iterative steps, the expectation (E) step and the maximization (M) step, which are alternately repeated until a convergence criterion is satisfied. Lawrence and Reilly (1990) first developed an EM algorithm and tested on cyclic adenosine monophosphate receptor protein (CRP) binding sites. One starts with an initial guess on the base probability $p_{bx}$ within the sites and the background $p^0_b$ for the non-sites, then the probability of the event $B_{jy}$ that the site begins at position $y$ in sequence $j$ can be calculated by Bayes formula:

$$P(B_{jy}|p, S) = P(S|B_{jy}, p) / \sum_x P(S|B_{jx}, p) \quad (10.4)$$
where $S$ is the sequence data and the prior probability $P^0(B_{jy})$ is uniform, i.e. $1/(L-k+1)$ and $P(S|B_{jy}, p) = \text{product of probabilities for all bases}$. Using (10.4), one can complete the E-step by calculating the expected number $n_{bx}$ of the base $b$ at position $x$ in the site and the expected number $n_{0b}$ of the base $b$ in the non-sites. The M-step is simply to replace $p = \{p_{bx}, P^0_b\}$ by $\{n_{bx}/N, n_{0b}/N\}$ (the maximum likelihood estimators). One then iterates this to convergence when the parameter estimates no longer change.

MEME (Bailey and Elkan 1994) added several extensions to EM to overcome some limitations. MEME chooses starting points systematically, based on all subsequences of the data. It eliminates the assumption of one motif per sequence and allows each sequence to contain zero, one or several appearances of the shared motif. Furthermore, MEME probabilistically “erases” the appearances of a site after it is found, and continues searching for other shared motifs in the dataset. The newer version has made MEME smarter and more robust as an unsupervised motif-discovering tool,
as it will automatically determine the motif length and/or choose whether or not to enforce the palindrome constraint (Bailey and Elkan 1995). Once a MEME motif is found, MAST (Bailey and Gribskov 1998) can be used to search other sequences for new members. Both MEME and MAST is available at http://www.sdsc.edu/MEME.

10.3.2.3 Gibbs sampler

As greedy or EM based algorithms cannot guarantee to find the global maximum and may be prone to local optima, stochastic algorithms have been developed to overcome this problem. Gibbs sampler, which consists of a site sampler (Lawrence et al 1993) and a motif sampler (Neuwald et al 1995), has been a very successful one.

The site sampler is assuming every sequence contains at least one site. The algorithm is initialized by choosing random starting positions within all the sequences. It then proceeds through the following two steps of Gibbs sampler iteratively: (a) Building model step. Construct a model $p = \{p_{b,x}, p_0^b\}$ like in the EM case using all the sequences with the
selected sites except the first sequence; (b) Sampling step. Sample a new site for the first sequence from every possible positions according to a relative weight $p_{b,x}/p_0^b$. Then repeat (a),(b) for the second sequence, etc. and one cycle is complete when the site for the last sequence is re-sampled. Theoretically, after infinite number of cycles, the relative information $I = \Sigma_x I_x$ (10.2) will reach its maximum. In practice, the alignment often converges fairly fast. Sometimes the sampler could get stuck at sub-optima, which may require simultaneous shift of all the aligned sites to the left or to the right by few bases. To speed up the convergence, the Gibbs sampler automatically samples a shift according again to the relative weight of the likelihood ratio after certain specified number of cycles. This basic algorithm was also generalized to allow more than one type of motif per sequence.

In order to find sites that may have multiple copies in some sequences and zero copy in others, the motif (or Bernoulli) sampler was designed to
concatenate all the sequences into a single one. It is initialized to an alignment of sites randomly spread throughout (no site can overlap another or across a sequence boundary) and the rest are non-sites. The algorithm starts with picking the first possible site position out of either the aligned set or the no-site set, update the model \( p = \{ p_{bx}, p_0 \} \) (i.e. recalculate the base counts), then sample this position into the aligned set or the no-site set according to the odd ratio \( \frac{P(site) \cdot p_{bx}}{P(non-site) \cdot p_0} \), where the posterior (prior + pseudo-counts) \( P(site) \) and \( P(non-site) \) may be specified by a user. Then the algorithm continues with picking the second and etc. One cycle is complete when the last possible site position is sampled. Iterate enough cycles till convergence of the motif alignment (or the maximum number of cycles specified). This has been generalized to handle more than one type of motifs. Two other useful technical features are the column sampling (to allow automatically increasing the site length by sampling in more conserved flanking columns) and the near
optimal sampling (to allow estimating relative probability each site is sampled in to the alignment). A Gibbs sampler server is currently maintained at http://bayesweb.wadsworth.org/gibbs/. Two other modified versions of Gibbs sampler for DNA sequence analysis have been reported in microarray data analysis applications (AlignACE: Roth et al 1998 and GibbsDNA: Zhang 1999a).

10.3.3 Statistical significance

It is also important to know how significant a particular alignment is with respect to a random model. Since all the alignments are ranked by the relative information \( I \), it would be desirable to calculate the p-value, namely, the probability of finding an alignment with relative information greater or equal to \( I \). Assuming the null model for each alignment column is an independent multinomial model:

\[
P_{\text{matrix}} = \prod_{x=1}^{k} \left[ \frac{N!}{\prod_{b=A}^{T} \eta_{hx}} \sum_{b=A}^{T} p_{b}^{hx} \right]. \quad (10.5)
\]
If $I$ is small and $N$ is large, $2NI$ tends to a $\chi^2$ distribution (df = 3k). Unfortunately, promoter analyses generally involve very large scores and frequently few sequences, the limiting distribution tend to give poor probability estimates. Using large-deviation technique, Hertz and Stormo (1999) obtained the approximate mathematical formula for the p-value and the E-value (expected number of alignments with $I$ or greater). They also implemented an efficient algorithm for calculating these values in CONSENSUS. Other methods have also been reported for estimating statistical significance of a matrix search result (e.g. Staden 1989; Claverie and Audic 1996).

10.3.4 Constructing regulatory modules

Since promoter is regulated by TF modules made of composite sites, simultaneous detecting correlated sites is much more significant and hence provides better specificity. Claverie and Sauvaget (1985) published one of the earliest methods for detecting two sites in a fixed distance and orientation in the heat-shock promoters.
Another interesting example was given by the identification of regulatory modules that confer muscle-specific gene expression (Wasserman and Fickett 1998), where a logistic regression analysis (LRA: Hosmer and Lemeshow 1989) was used to combining matrix scores for multiple TF sites in each module. This directly generalized the study of the two site module (MEF2/MyoD model: Fickett 1996).

More recently, experimental analysis and computer prediction of CTF/NFI TF sites were reported (Roulet et al. 2000) where a generalized profile model for CTF-1 DNA binding specificity was proposed. This model consists of a conserved half-site (5bp) + a spacer (5,6,7 or ∞) + a less conserved (palindromic) repeat. Detailed experimental analysis reveals flexible and correlated nature of this protein binding site.

With detailed modeling of TF modules, one will be able to recognize promoters of a specific class with extremely high specificity (Frech et al. 1998). Unfortunately, generation of these models require high quality as well as systematic experimental data which
are still very rare nowadays. The development of composite site database, such as COMPEL and TRRD (Heinemayer et al 1998) will greatly facilitate the advances in this field.

10.3.5 **Large-scale gene expression**

Recent advent of large-scale gene expression technologies is having a great impact on the understanding of gene regulation (e.g. Schena et al. 1995; Lockhart et al. 1996). By clustering of gene expression profiles, different groups of co-regulated genes can be identified and their promoter elements may be detected by either k-tuple (Zhu and Zhang 2000; van Helden et al 1998) or multiple alignment (Zhang 1999a, Hughes et al 2000) methods once the upstream sequences become available for the transcriptionally co-regulated genes (e.g. DeRisi et al 1997; Spellman et al. 1998; Cho et al 1998; Roth et al 1998). Using large-scale expression data to detect novel promoters and to infer regulation networks will become the cutting-edge bioinformatics in the functional genomics era (Zhang 1999b, Bucher 1999, McGuire et al 2000).
Species specific promoter databases, such as SCPD (Zhu and Zhang 1999), shall become extremely useful resources for studying large-scale transcription data.

10.3.6 Phylogenetic footprinting

As more genomes become available, comparative analysis of noncoding regions have also become an important approach for detecting promoters or regulatory regions in general (Aparicio et al 1995, Gumucio et al 1996, Jareborg et al 1999). Phylogenetic footprinting is referred (Fickett and Wasserman 2000) to as the identification of any functional regions by comparison of orthologous genomic sequences between species. Although the orthologous coding regions (ORFs) are highly conserved, the conservation of regulatory regions varies widely with particular genes. To detect short TF sites, one would want to compare orthologous regulatory regions between species that are not too close (so that the sequences have enough time to diverge) and not too distant (so that some related regulatory regions are still recognizable, see Duret and Bucher 1997). But we are
currently limited to the few sequenced model systems. Several methods for detecting conserved blocks from a multiple alignment have been evaluated by Stojanovich et al (1999). Programs designed for very long alignments of syntenic regions have also become available (e.g. PIPmaker: Schwartz et al 2000; MUMmer: Delcher et al 1999). Among many applications, PIPmaker was very successfully used for the identification of a coordinate regulator of interleukins 4, 13 and 5 (Loots et al 2000).

10.4 Predicting transcriptional start site (TSS)

Since TF sites can occur anywhere, even with the location of a regulatory module in proximal promoter, identification of TSS is still not easy. General promoter prediction methods mainly focus on TSS site prediction in order to locate the beginning of a gene instead of seeking for specific regulatory elements.

10.4.1 CpG-islands

Vertebrate genomic DNA is known to be generally depleted in the dinucleotide CpG. In the human genome, for example, the occurrence of CpG dinucleotides is
five times less than statistically predicted from the nucleotide composition (Bird 1980). CpG depletion is believed to result from methylation of Cs at 80% CpG dinucleotides which leads to mutation of the methylated C to T, and thus conversion of the CpG dinucleotides to TpG (Bird 1999). There are, however, genomic regions of high G+C content, termed CpG islands, where the level of methylation is significantly lower than the overall genome. In these regions, the occurrence of CpGs is significantly higher, close to the expected frequency. As defined by Gardiner-Garden & Frommer (1987), CpG islands are greater than 200 bp in length, have more than 50% of G+C content, and have the ratio of the CpG frequency to the product of the C and G frequencies above 0.6. CpG island is an important signature of 5' region of many mammalian genes, often overlapping with, or within 1000 bases downstream of, the promoter (Cross and Bird 1995), where a promoter associate nucleosome is found (Ioshikhes et al 1999). The identification of promoters by CpG islands with a resolution of 2 Kb
will be most useful for large-scale sequence annotation. Although visual inspection of CpG islands is often used for gene identification by many molecular biologists, Ioshikhes and Zhang (2000) have recently optimized the features that can best discriminate the promoter-associate CpG islands from the non-associated ones. This led to an effective algorithm (CpG_Promoter) for large-scale promoter mapping with 2 kb resolution. Statistical tests showed that ~85% of CpG islands within an interval from -500 to +1500 around the TSS (transcriptional start site) were correctly identified and that ~93% of the CpG-island containing promoters were correctly mapped. The basic procedure is to use CpGPlot program of R.S. Lopez (available at http://www.sanger.ac.uk/Software/EMBOSS/, see Larsen et al. 1992) for mapping of potential CpG islands and then to use a Quadratic Discriminant classifier for prediction of promoter associated islands. The EMBL CPGISLE database of human CpG islands was used for training the classifier on three discriminant features: length, G+C content and the CpG
ratio (obversed / expected). The information about CpG_Promoter is available at ftp://cshl.org/pub/science/mzhanglab/ioshikhes/. It should be mentioned that, like CpGPlot, PIPmaker can also display CpG islands in a large genome.

10.4.2 TSS prediction based on TF site scan

As TF sites are over-represented in the promoter region, it is natural to seek a prediction program based on putative TF site density. PROMOTERSCAN is one of such programs developed by Prestridge (1995). It was based on the study of three datasets: a TF database (TFD: Ghosh 1993), a promoter database (EPD: Bucher and Trifonov, 1986), and a non-promoter set constructed from protein and RNA gene sequences. Density of all putative TF sites is calculated separately for promoter and non-promoter sequences (within 250 bp window upstream of TSS for promoters) and use the ratio \( R = \frac{D_p}{D_n} \) of the two densities as the scoring function supplemented with a TATA-matrix score (Bucher 1990) when scanning a test sequence using the same window. A web server for the program is

A very similar, albeit statistically more sophisticated, approach was taken by Kondrakhin et al (1995) and implemented in AUTOGENE.

10.4.3 TSS prediction based on K-tuples

Methods based on putative TF sites do have severe limitations: important context effect may be overlooked, majority of putative sites are false positives, site matrices are scoring function/cut-off dependent, database is bias by limited number of known samples, etc. A statistical learning approach without using any putative TF site information has become the attractive alternative. PromFind (Hutchinson 1996) is based on the difference in 6-mer frequencies between promoters, coding regions and noncoding regions downstream of the first coding exon. Among all sites in an input sequence where the promoter versus coding region discriminant exceeds a certain threshold, the site where the promoter versus noncoding region discriminant reaches its maximum (over the input
sequence) is taken as a promoter. But this “content” approach would loss all the positional information. TSSG and TSSW (Solovyev and Salamov 1997) both use LDA (linear discriminant analysis) to combine (a) a TATA score, (b) triplet preferences around TSS, (c) 6-tuple score in three non-overlapping windows of 100 bp upstream TSS, and (d) putative binding site scores. TSSG is based on TFD (Ghosh 1993) and TSSW is based on TRANSFAC (Wingender et al 1996). They are available at http://dot.imgen.bcm.tmc.edu:9331/gene-finder/gf.html.

Fickett and Hatzigeorgious (1997) had evaluated several earlier promoter prediction algorithms, including a Markov model based algorithm (Audic and Claverie 1997). TSSW appeared to be ranked as one of the bests with sensitivity of 42% and specificity of 1 false positive per 800 bp.

Using positional dependent 5-tuple measures, a QDA method for core promoter prediction was implemented in CorePromoter (Zhang 1998). Statistical tests indicated that when given a 2kb upstream region, CorePromoter was able to localize the TSS to a 100 bp
interval approximately 60% of the time. The utility of CorePromoter and CpG_Promoter was recently demonstrated (Zhang 2000) in the re-analysis of human chromosome 22 genes in conjunction with our internal exon finder MZEF (Zhang 1997) and the terminal exon finder JTEF (Davuluri et al 2000a).

A recent algorithm PromoterInspector (Scherf et al 2000) is based on libraries of degenerate words extracted from training sequences by an unsupervised learning approach. It consists of three classifiers which discriminate promoter from intron, exon and 3’-UTR separately and predicts a promoter when all three classifiers agree. Their test showed that 43% of the predictions can be expected to be true positives, while 43% of the annotated TSS were predicted correctly. PromoterInspector is available at http://genomatix.gsf.de/cgi-bi/promoterinspector/promoterinspector.pl.

In principle, TSS may also be predicted by discriminant analysis of 5’UTRs and with more full length cDNAs becoming available, 5’UTR features will
become as important as promoter features in TSS prediction (Davuluri et al 2000b).

10.5 References


17 Claverie J-M and Sauvaget I. Assessing the biological significance of primary structure


28 French K, Quandt K and Werner T. Muscle actin genes: a first step towards computational


33 Heinemayer T, Wingender E, Reuter I, Hermjakob H, Kel AE, Kel OV, Ignatieva EV, Ananko EA, Podkolodnaya OA, Kolpakov FA, Podkolodny NL,


44 Lawrence CE, Altschul SF, Bogouski MS, Liu JS, Neuwald AF and Wooten JC. Detecting subtle


59 Scherf M, Klingenhoff A and Werner T. Highly specific localization of promoter regions in large genomic sequences by Promoter Inspector: A


73 Wingender EP, Dietze P, Karas H and Knueppel R. TRANSFAC: A database on transcription factors and


79 Zhang MQ. Discriminant analysis and its application in DNA sequence motif recognition. Presented at the Gene-finding Workshop, June


Figure 10.1. Sequence elements and GTF footprints in a typical core promoter.
Figure 10.2. Recruitment and activation of the PIC.
Figure 10.3 A diagram of the algorithm in CONSENSUS (Hertz et al. 1990).