Analysis of Biological Networks:
Transcriptional Networks - Promoter Sequence Analysis

Lecturer: Roded Sharan Scribe: Shaul Karni and Yifat Felder *

Lecture 11, January 4, 2007

1 Introduction

Each cell of an organism contains an identical copy of the whole genome. However, different tissues have different functions, which means that each cell utilizes only a subset of its genes. The process in which the cell controls the activation of a subset of genes is called *gene regulation*. Gene regulation can occur at different time points along the gene expression process, including chromatin organization, transcription, pre-mRNA processing, translation and post-translational modifications. Primarily it occurs at the transcriptional level. This lecture focuses on computational approaches to decipher transcriptional regulation.

1.1 Transcriptional Regulation

The proteins that mediate transcriptional regulation are called *transcription factors (TFs)*. TFs bind to specific short DNA sequences called *binding sites (BSs)* or *transcription factor binding sites (TFBSs)*, which are 5-20 bp in length. They are mainly located in a region upstream to the regulated gene, called *promoter*. Although, some BSs have been found from 200 bp downstream to the ORI (Origin of replication), up to tens of thousands bp upstream. Promoter length can vary from 100 to 1000 base pairs. The RNA polymerase binds to the promoter in a region called the *core-promoter* (Figure 1).

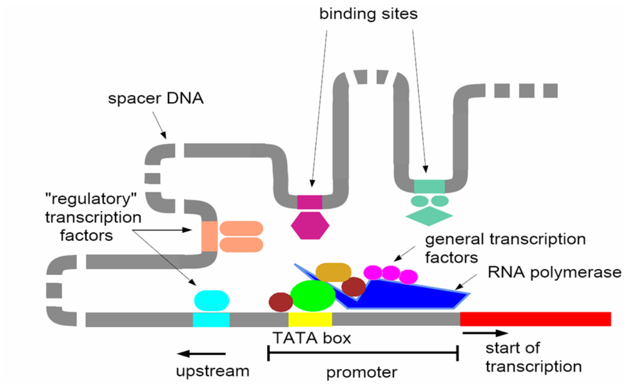


Figure 1: A cartoon of gene regulation

*Based on a scribe by Ofir Davidovich and Yonit Halperin

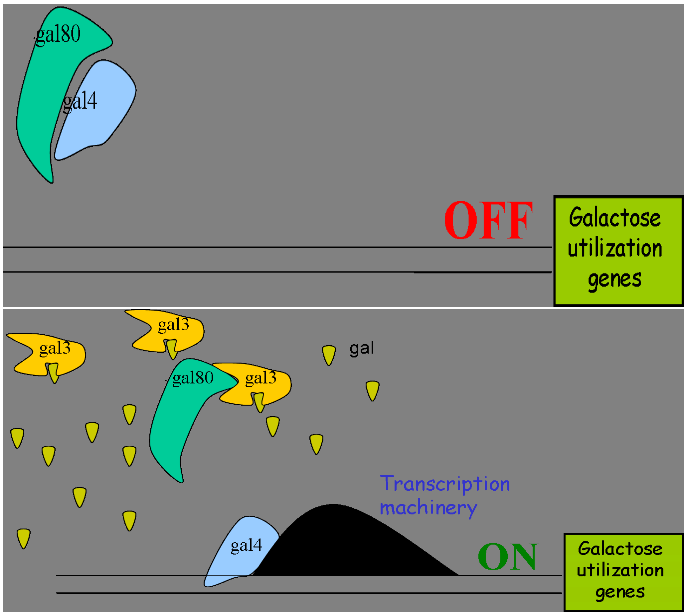


Figure 2: Yeast Galactose utilization gene regulation. The upper part demonstrate gal80 interacts with gal4 (TF) and prevents transcription. In the lower part gal3 interacts with gal80 and allows gal4 to bind to the DNA and activates transcription.

Each TF can regulate multiple genes. A TF can serve as an *activator*, i.e, increase the transcription level of the regulated gene, or as a *repressor*, i.e, decrease its transcription level. There are two main modes of activation:

1. After binding to the promoter, the TF interacts with components of the RNA polymerase. This interaction attracts the RNA polymerase in the vicinity of the gene promoter, thereby facilitating its binding to the core promoter.

2. When the TF binds the DNA, the chromatin structure in the promoter region changes and the binding area of the RNA polymerase becomes more accessible.

There are also two main modes of repression:

1. The repressor competes with an activator TF on its BS. Therefore, it decreases the effects of the activator, leading to less efficient binding of RNA polymerase to the promoter. This obviously results in lower expression levels of the gene.

2. The repressor interacts with the same components of RNA polymerase as does an activator. By doing this, the repressor prevents the activator from interacting with the RNA polymerase and promoting transcription.

A known example for transcriptional regulation is the galactose utilization genes in yeast (Figure 2). In glucose rich environment, gal4 (the TF that activates the transcription of galactose utilization genes) is inhibited by gal80. However, in galactose rich environment, the galactose activates the gal3 protein. The activated gal3 competes with gal4 over interacting with gal80. Thus, removing the inhibition of gal4 and leading to up-regulation of the galactose utilization genes.

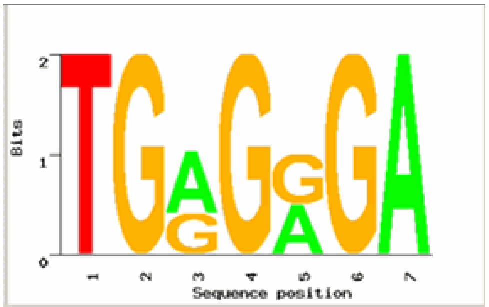


Figure 3: A motif logo representation for the following BSs: TGGGGGA, TGAGAGA, TGGGGGA, TGAGAGA, TGAGGGA. X-axis represents the position in the BS sequence and the Y-axis is the energy function.

1.2 BS Motifs

TFs binds to short sequences in the promoter. The protein structure of the binding region of a TF dictates the length and the base composition of the sequences it can bind. Though the BSs of a TF have a core pattern that is essential for the TF binding, their sequences may vary. This is because some parts of the sequence are more important to binding and thus less subject to viable mutations. The BS sequence pattern is called a *motif.* A common way of representing a motif is by using a *motif logo* (Figure 3). The motif logo illustrates the conserved and variable regions of the motif by displaying the information of every position in the motif. The information at each position is $2 + \sum_i p_i log p_i$ where i ∈{A,C,G,T} and p_i is the frequency of nucleotide i.

1.3 Modeling Motifs

To model a BS motif, we begin by lining up the sequences of the BSs and constructing a *profile matrix*. The profile matrix holds the frequencies of each nucleotide at each position (Figure 4).

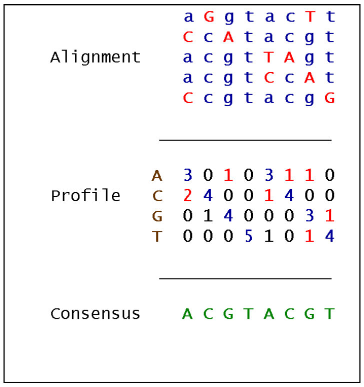
The following models are used to represent BS motifs:

Consensus string: Each position in the *consensus string* holds the most frequent nucleotide in that position (Figure 4).

Degenerate string: Each position in the *degenerate string* holds all the nucleotides that appear in that position in at least one of the BSs. For example, the first position in the degenerate string that models the BSs in figure 4 will hold A, C.

Frequency matrix: The *frequency matrix* is simply the profile matrix where each cell is divided by the number of BSs. With a very large number of BS sequences an observed nucleotide frequency is expected to be approximately equal to the actual probability of finding that nucleotide in a BS. However, in most cases the number of BSs is limited so that for some nucleotides the observed frequency is equal to 0 whereas the actual probability should be > 0. For this reason ”fake counts”, *pseudocounts*, are added to avoid zero probability.

Position Weight Matrix (PWM): The *Position Weight Matrix (PWM)* is calculated from the frequency matrix in the following manner: for position i and nucleotide b, the value in the matrix is $log\frac{P_i(b)}{f(b)}$ where



```
              a G g t a c T t
              C c A t a c g t
Alignment     a c g t T A g t
              a c g t C c A t
              C c g t a c g G

            _________________

           A  3 0 1 0 3 1 1 0
Profile    C  2 4 0 0 1 4 0 0
           G  0 1 4 0 0 0 3 1
           T  0 0 0 5 1 0 1 4

            _________________

Consensus     A C G T A C G T
```

Figure 4: Example of creating a consensus string of a profile matrix created from an alignment of a set of BSs.

$P_i(b)$ is the corresponding value in the frequency matrix and $f(b)$ is the background frequency collected from promoter sequences. Taking a subsequence and summing up the scores of each position according to the PWM gives the log-likelihood of this subsequence to be an instance of the corresponding motif. The PWM is the most commonly used model for representing BS motifs.

1.4 Computational Challenges In Promoter Analysis

Known TFBS models (discussed in Sections 2 and 4)

- Find all the BSs of a given TF in a given set of promoters, given its TFBS model. Since the BS motif is not a unique string, it is not obvious how to determine if a sequence is indeed a BS. The most common method to find the BSs is first to compute for every subsequence the log-likelihood score according to the PWM. Second, to determine a score threshold and declare the subsequences that pass this threshold as BSs.

- Find enriched known TFBS models in a given set of co-regulated genes. The biological assumption is that co-regulated genes (or at least a significant portion of them) are regulated by a common TF. So, if we can find a motif model that is significantly enriched in those genes' promoters we can deduce that the corresponding TF regulates them.

- Identify enriched combination of TFBS models. The biological motivation is that transcriptional regulation might be combinatorial, meaning that the gene expression level is affected by an interplay among several TFs, whose BSs are organized along the promoter.

Unknown TFBS models (discussed in Section 3)

- Find enriched de novo motifs in the promoters and a given a set of co-regulated genes. This challenge arises since not all the TFs are known or characterized in terms of their BSs.

1.5 Data sources

Data sources that are used in promoter analysis:

- Promoters sequences data - can be derived from several databases (e.g. RefSeq). Besides of the obvious use of this data as input to all promoter analysis algorithms, promoters of homologue genes can be used for finding conserved regions in the analyzed promoters. This can reduce the search area for binding sites to conserved regions only.

- ChIP-chip data - Chromatin Immunoprecipitation (ChIP) is a procedure that isolates DNA sequences that are bound to a studied TF in-vivo. These sequences are then amplified by PCR and analyzed using a DNA chip. This method helps us to identify BS sequences of a studied TF and to build a TFBS model. It also enables us to identify potentially co-regulated genes that then can be searched for TFBSs.

- Expression data - can be derived from several databases like GEO (Gene Expression Omnibus [12]). Expression data can be used to identify putative co-regulated genes from similar expression patterns. The basic biological assumption is that genes that are co-expressed over multiple biological conditions are regulated by common TFs, and therefore are expected to share common regulatory elements in their promoters. Also, knockout experiments, where a TF is knocked out, can be used to identify the genes that had a significant change in their expression pattern, and thus are possibly regulated by the knocked out TF.

- Biological annotations - can be derived from the GO database. They can be used to deduce groups of co-regulated genes and use them as input for algorithms. They also can be used to validate results of an algorithm by taking a group of genes that were predicted to have BSs of the same TF and test this group for functional enrichment.

- Databases of known TFs and BSs - the most extensive database is TRANSFAC [10], which is a commercial database. A free known database is JASPAR [9].

2 Analyzing known motifs

2.1 The PRIMA algorithm

PRIMA (Promoter Integration in Microarray Analysis [3]) is an algorithm for identifying TFs whose binding sites are enriched in a given set of promoters. By utilizing genomic sequences and PWM models for binding sites of known TFs, PRIMA identifies TFs whose BSs are significantly over-represented in a given set of co-expressed genes' promoters. PRIMA also allows identifying pairs of TFs whose BSs tend to co-occur in a given promotor set. The algorithm is integrated into the Expander software [11].

The input of the algorithm is a target set of co-regulated genes, a background set of genes, and PWMs of known TFs. The output is a set of enriched TFs and their p-values. For each TF, the algorithm starts with an annotation phase, in which it computes a threshold score for determining hits (putative BS) of the PWM and scans the background and target set promoters for hits. Each subsequence whose log likelihood score according to the PWM passes the threshold is marked as a hit. The threshold is computed as follows: First, we generate random sequences (e.g. 1,000 sequences of length 1,000 bp) by using a 2nd-order Markov-Model which is based on the background sequences. Then, we set a threshold for each PWM so that it has f% hits in the random sequences (PRIMA uses f=5). This method of determining the PWM's parameters ensures a pre-defined false-positives rate, but has no guarantee on false-negatives rate. Next, an enrichment score, based on the hits we annotated, is computed for each PWM. The enrichment score evaluates whether the number of hits in the target set is significantly higher than expected by chance, given the distribution of hits in the background.

A simple version of the algorithm assumes that each promoter has 0 or 1 hits. When B is the number of background promoters and b the number of background promoters that have hits (includes target promoters). T the number of target promoters, and t the number of target promoters that have hits. Then, based on a hyper-geometric distribution, the probability for t hits in the target-set in random equals to

$$P(t) = \binom{b}{t}\binom{B-b}{T-t}\Big/\binom{B}{T}$$

The enrichment score for a single PWM is the probability for at least t hits in random:

$$\sum_{i=t}^{\min\{b,T\}} P(i)$$

A more general enrichment score takes into account more than 1 hit per promoter. The motivation for this score is that there are TFs whose activity is affected by the number of BSs in the promoter. Because of complexity considerations PRIMA takes into account up to 3 hits per promoter. As above, B is the number of background promoters, T the number of target promoters and t total number of hits in the target set. Also$b_1,\ b_2,\ b_3 =$ are the number of background promoters with 1, 2 and at least 3 hits, respectively. The probability for at least t hits in a random (hyper- geometric distribution) is:

$$\frac{\sum_{i+2j+3k\geq t}\binom{b_1}{i}\binom{b_2}{j}\binom{b_3}{k}\binom{B-b_1-b_2-b_3}{T-i-j-k}}{\binom{B}{T}}$$

PRIMA can also test for TF synergism. The objective of this test is to find pairs of TFs that tend to co-occur in the same promoters. In order to find enriched pairs, the algorithm must overcome the possible artifact of getting an enriched pair only because each TF is enriched by itself. PRIMA does it by using a score for each pair of PWMs that ignores the enrichment of each TF on its own. As before, T is number of promoters in the target set. Also, t_1 is number of promoters in the target set with at least one hit of TF 1, t_2 number of promoters in the target set with at least one hit of TF 2 and t_{12} is the number of promoters in the target set with at least one hit of both TFs (without overlaps between hits). The score is based on a hyper-geometric distribution where we assume that the promoters with hits of TF 1 were already determined and we calculate the probability that at least t_{12} out of these promoters have also hits of TF 2 in random. Thus the probability is:

$$\frac{\sum_{i\geq t_{12}}\binom{t_1}{i}\binom{T-t_1}{t_2-i}}{\binom{T}{t_2}}$$

2.2 PRIMA results on Human Cell Cycle:

Whitfield et al. [4] partitioned a set of 568 *human cell cycle* (HCC) regulated genes according to their expression periodicity patterns into five clusters corresponding to different phases of the cell cycle. PRIMA was used to search for significantly enriched PWMs in the entire set of the 568 cell cycle-regulated promoters. A 13K set of known human genes promoters, each 1200 bp in length, was used as the background set. Six out of the 107 PWMs tested, corresponding to E2F, NF-Y, NRF-1, Sp1, ATF, and CREB TFs, were significantly enriched. Then, enriched PWMs only in specific phase clusters were searched. Arnt and YY1 PWMs were specifically enriched in the G1/S and the M/G1 clusters, respectively (Figure 5). The location distribution of the hits for the 8 enriched PWMs found was analyzed. The hits for E2F, NF-Y, NRF-1, Sp1, ATF, and CREB tend to concentrate in the proximity of the transcription start site (TSS) (Figure 6). This observation is in agreement with experimental data on the locations of in vivo binding sites of TFs. The set was also tested by PRIMA to contain enriched pairs of PWMs (synergism test). All possible pairs formed by the 8 PWMs mentioned before and ETF (found enriched in an analysis on a different HCC data set) were examined. Eight pairs showed a significant tendency to co-occur in the target set (Figure 7).

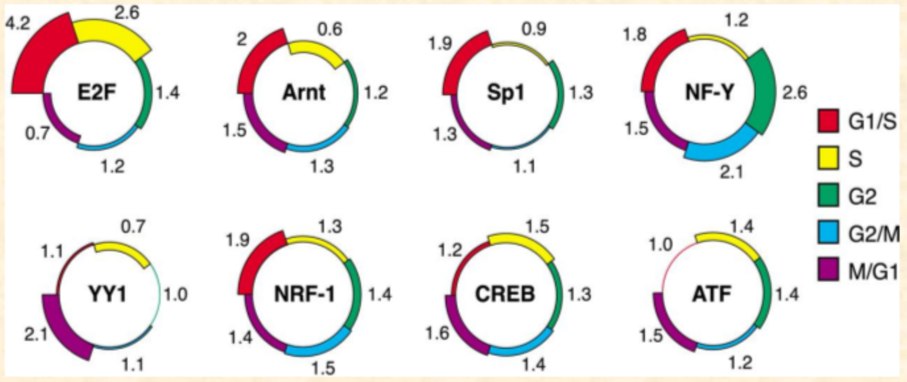


Figure 5: Source: [3]. Representation of TF PWMs in the cell cycle phase clusters. The eight circles correspond to the PWMs that were highly enriched in promoters of cell cycle-regulated genes. Each circle is divided into 5 zones, corresponding to the phase clusters. The number adjacent to the zone represents the ratio of its prevalence in promoters contained in each of the cell cycle phase clusters to its prevalence in the set of 13K background promoters. Note that several TFs show a tendency towards specific cell cycle phases: e.g., over-representation of the E2F PWM in promoters of the G1/S and S clusters, and its under-representation in promoters of the M/G1 cluster.

3 De-novo Motif Finding

An important problem in motif analysis is: given a set of co-regulated genes, finding motifs that are enriched in their promoters. There are two types of algorithms that try to solve this problem:

1. Combinatorial algorithms - assume a discrete model for a motif (e.g degenerate string, consensus string, a string with mismatches). These algorithms search for motifs with high rate of occurrences in the promoters. Such an algorithm is random projection [2].

2. Probabilistic algorithms - assume that BSs of a TF are generated by probabilistic model which is different from the background model. These algorithms try to find the model by which BSs of the same TF are generated. Examples: MEME (Baily & Elkan) - EM based [1], Lawrence et al. - Gibbs sampling based [8] and Segal et al. - discriminative approach [6].

3.1 MEME

The input for MEME is a set of sequences and a width of the searched motifs (denoted l). The output consists of PWMs of the motifs found. MEME uses the *Expectation Maximization* (EM) approach.

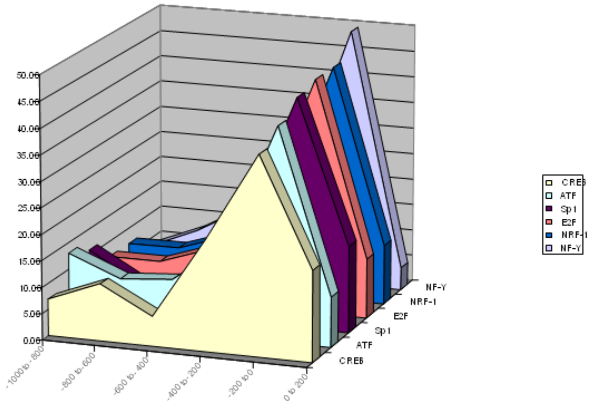


Figure 6: Source: [3]. Distribution of locations of TFs putative binding sites found in 568 cell cycle regulated promoters. Promoters were divided into six intervals, 200 bp each. For each of the PWMs, the number of times its computationally identified binding sites appeared in each interval was counted (after accounting for the actual number of bps scanned in each interval. This number changes as the masked sequences are not uniformly distributed among the six intervals). Locations of NRF-1, CREB, NF-Y, Sp1, ATF and E2F binding sites tend to concentrate in the vicinity of the TSSs (χ^2 test, where p-value is 0.01).

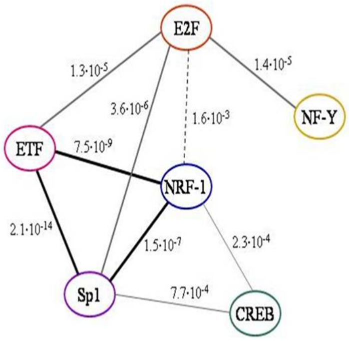


Figure 7: Source: [3]. Synergism test - Pairs of PWMs that co-occur significantly in promoters of genes regulated in a cell cycle manner. It was examined whether the PWMs can be organized into regulatory modules. For each possible pair formed by these PWMs, we tested whether the prevalence of cell cycle-regulated promoters that contain hits for both PWMs is significantly higher than would be expected if the PWMs occurred independently. Eight significant pairs were identified, each connected by an edge. The corresponding p-value is indicated next to the edge. The edge connecting the E2F-NRF1 pair is dashed to indicate that its significance is borderline.

3.1.1 The Model

To calculate the likelihood of a motif to occur, two models are calculated. The *motif model* which for each location, calculates the probability of that base being part of a motif at that location. And the *backround model*, which calculates the probability of a base to be part of a motif at any location (single nucleotide probabilities). Then, by using both models, calculate the likelihood of a motif to occur a a specific location, thus removing results that could be achieved in a random data set.

The promoter sequences are broken into all their n overlapping l-mers: X = (X_1,...,X_n). MEME assumes that the X_is were generated by a two-component mixture model θ = (θ_1, θ_2), where θ_1 is the parameter vector of the motif model and θ_2 is the parameter vector of the background model. θ_1 holds $f_{i,b}$ which is the probability of the occurrence of base b at position i in a motif, $i = 1..l$, $b \in \{A, C, G, T\}$. θ_2 holds $f_{0,b}$ which is the probability of the occurrence of base b at any position. The background model is determined by a zero order Markov model. λ = (λ_1 , λ_2) is the mixing parameter, i.e. λ_j is the probability that a l-mer was generated by model j ($\lambda_1 + \lambda_2 = 1$). The hidden data of the model is for each X_i, which model generated it, the background model or the motif model.

Let $Z = (Z_1, ..., Z_n)$, $Z_i = (Z_{i1}, Z_{i2})$ where Z_{ij} is an indicator: Z_{ij} = 1 if X_i was generated by model j and 0 otherwise. Then, X_i can be calculated by:

$$P(X_i|Z_i, \theta, \lambda) = \prod_{j=1,2} P(X_i|\theta_j)^{Z_{ij}} = [\prod_{k=1}^{l} f_{0,x_k^i}]^{Z_{i,2}}[\prod_{k=1}^{l} f_{k,x_k^i}]^{Z_{i,1}}$$

and Z_i is calculated by:

$$P(Z_i|\theta, \lambda) = \prod_{j=1,2} \lambda_j^{Z_{ij}}$$

therefore:

$$P(X_i, Z_i|\theta, \lambda) = P(X_i|Z_i, \theta, \lambda)P(Z_i|\theta, \lambda) = \prod_{j=1,2} [\lambda_j P(X_i|\theta_j)]^{Z_{ij}} = [\lambda_2 \prod_{k=1}^{l} f_{0,x_k^i}]^{Z_{i,2}}[\lambda_1 \prod_{k=1}^{l} f_{k,x_k^i}]^{Z_{i,1}}$$

and:

$$P(X, Z|\theta, \lambda) = \prod_{i=1}^{n} P(X_i, Z_i|\theta, \lambda)$$

What we want to calculate is:

$$L = logP(X|\theta, \lambda) = log \sum_{z} P(X, Z|\theta, \lambda)$$

But this is a mathematically difficuly calcultion. Thus, EM is used to learn the parameters. The EM algorithm iteratively maximizes the expected complete log likelihood over the conditional distribution of the hidden data Z given the observed data X, i.e. $E[logL(\theta, \lambda|X, Z)$. And then estimates the parameters θ and λ.

3.1.2 Outline of the EM algorithm

- Choose starting $\theta^{(0)}$, $\lambda^{(0)}$.

- Repeat until convergence of θ and λ:

  - E-step: Re-estimate Z from θ, λ, X, by computing the expected complete log likelihood.
  - M-step: Re-estimate θ, λ from X, Z, by maximization of the E-step product.

- Repeat all of the above for various $\theta^{(0)}$, $\lambda^{(0)}$ starting points.

E-step:
The expected value of the complete log likelihood over the values of the hidden data Z given X and the current parameters $\theta = \theta^{(i)}$ and $\lambda = \lambda^{(i)}$:

$$E[logL(\theta, \lambda|X, Z)] = \sum_{i=1}^{n} \sum_{j=1,2} Z_{ij}^{(i)} log(\lambda_j P(X_i|\theta_j)) =$$

$$= \sum_{i=1}^{n} \sum_{j=1,2} Z_{ij}^{(i)} log(P(X_i|\theta_j)) + \sum_{i=1}^{n} \sum_{j=1,2} Z_{ij}^{(i)} log(\lambda_j) = L_1 + L_2$$

Where:

$$Z_{ij}^{(i)} = E[Z_{ij}] = P(Z_{ij} = 1|\theta^{(i)}, \lambda^{(i)}, X_i) = \frac{P(Z_{ij} = 1, X_i|\theta^{(i)}, \lambda^{(i)})}{P(X_i|\theta^{(i)}, \lambda^{(i)})} =$$

$$= \frac{P(Z_{ij} = 1, X_i|\theta^{(i)}, \lambda^{(i)})}{\sum_{k=1,2} P(Z_{ik} = 1, X_i|\theta^{(i)}, \lambda^{(i)})} =$$

$$= \frac{\lambda_j^{(i)} P(X_i|\theta_j^{(i)})}{\sum_{k=1,2} \lambda_k^{(i)} P(X_i|\theta_k^{(i)})}$$

M-step:
Find θ and λ that maximize the expected log-likelihood. To find λ it is sufficient to find the value of λ that maximizes L_2 which is: $\lambda_j^{(l+1)} = \sum_{i=1}^{n} \frac{Z_{ij}^{(l)}}{n}$, $j = 1, 2$. To find θ, we need to maximize L_1 separately over θ_1 and θ_2.
The background model is calculated by:

$$f_{0b}^{(l)} = \sum_{i=1}^{n} \sum_{j=1}^{W} Z_{i2}^{(l)} I(b, X_ij)$$

where $I(b, a)$ is an indicator function which is 1 iff base a equals to b.
In the motif model:

$$f_{jb}^{(l+1)} = \frac{c_{jb}}{\sum_{b=1..4} c_{jb}}$$

where c_{jb} is the expected frequency that letter b is produced at column j:

$$c_{jb} = \sum_{i=1..n} Z_{i1}^{(l)} I(b, X_{ij})$$

Elaboration on the formulas can be found in Eq. 13-20 in [1].

3.2 Discriminative Motif Finding

Segal et al. [6] suggest a different probabilistic approach to the motif finding problem. Their goal is to find motifs that *discriminate* between promoter regions where the TF binds and those where it does not. This approach allows them to avoid the problem of learning the background distribution of the promoters. By doing that, abundant sequences that are not specific to the target set are not falsely declared as motifs. We denote the promoter sequence of a gene g as $S_1...S_n$. Each gene g has a binary variable $g.R(t)$ denoting whether t regulates g (from now on we will focus on a specific TF and use the notation $g.R$). We denote by θ_0 the probability distribution over nucleotides according to the background model. To model the motif we use a PWM model and use ψ_j to denote the distribution of characters in the jth position of the binding site. The motif length is k.

3.2.1 The Model

For a gene g we have:

$$P(S_1, ..., S_n|g.R = false) = \prod_l \theta_0[S_l]$$

$$P(S_1, ..., S_n|g.R = true) = \prod_l \theta_0[S_l] \sum_j \frac{1}{n-k+1} \prod_{i=1}^{k} \frac{\psi_i[S_{i+j}]}{\theta_0[S_{i+j}]}$$

where we assume a uniform prior over the binding position in case of regulation.
If we apply Bayes rule we get:

$$P(g.R = true|S_1, ..., S_n) = logit(x)$$

where

$$x = log\frac{P(S_1, ..., S_n|g.R = true)}{P(S_1, ..., S_n|g.R = false)} = log(\frac{P(g.R = true)}{P(g.R = false)} \frac{1}{n-k+1} \sum_j \prod_{i=1}^{k} \frac{\psi_i[S_{i+j}]}{\theta_0[S_{i+j}]})$$

and $P(R = true)$ is the prior on regulation occurrence. Now, we can see that the background effect cancels and the model can be simply parameterized by using k position-specific weights $w_j[l]$ ($l \in A, C, G, T$) and a threshold $v = log\frac{P(g.R=true)}{P(g.R=false)}$. Thus, we get:

$$P(g.R = true|S_1, ..., S_n) = logit(log(\frac{v}{n-k+1} \sum_j exp\{\sum_i w_i[S_{i+j}]\}))$$

These parameters can be learned heuristically using conjugate gradient ascent.

4 Cis-Regulatory Modules

Eukaryotic genes are often regulated by several transcription factors, whose binding sites are grouped in short sequences units called *cis-regulatory modules*. In order for multiple TFs to work together they probably need to be in close proximity on the promoter sequence (Figure 8). An important computational challenge is finding these cis-regulatory modules.

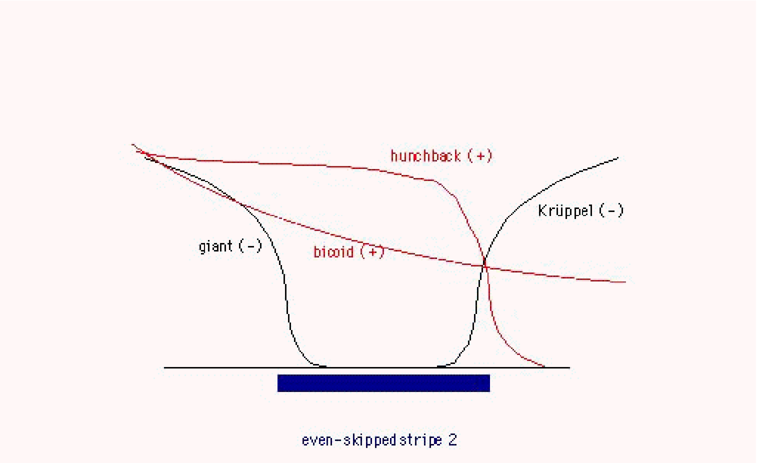


Figure 8: Source: [5]. Drosophila segment formation. Expression levels for four TFs according to their position on a drosophila body are displayed. Two of the TFs are repressors (giant and Krüppel) and two are enhancers (hunchback and bicoid). The lower stripe represents the area where the segmentation gene is transcripted. Within 430 bp in the promoter 12 BSs for the 4 TFs appear.

4.1 CREME: Cis-REgulatory Module Explorer

CREME [7] is a tool for identifying and visualizing cis-regulatory modules in the promoter regions of a given set of co-regulated genes. CREME relies on a database of putative TFBSs that have been carefully annotated across the whole genome. An efficient search algorithm is applied to this data set to identify combinations of transcription factors, whose binding sites tend to co-occur in close proximity within the promoter regions of the input gene set. These combinations are statistically evaluated, and significant combinations are reported and visualized.

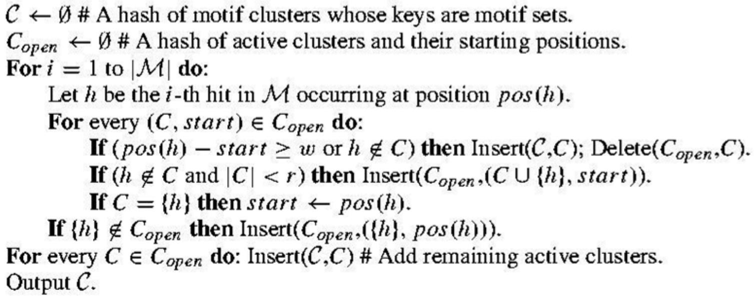
The algorithm receives as input all the promoters of the organism (the background), a set of promoters of co-regulated genes (the target) and a set of PWMs. The algorithm has four stages:

1. **Promoter annotation:** Under the biological assumption that BSs sequences are conserved among close species, the search for BSs can be limited only to conserved regions of the promoters. Hence, the algorithm searches for conserved regions in the promoters (both target and background) and ignores all the rest. This greatly reduces the search area for PWM hits and, hence, the rate of false positives. The hits of each PWM are annotated by calculating p-values and declaring all the subsequences that their score according to the PWM pass this threshold as hits.

2. **Finding key motifs:** Since many TFs are not relevant to a given gene set, only *key motifs* which are over or under-represented in the co-regulated set, are taken into account. Notice that the promoter lengths become variable after ignoring the non-conserved regions and that longer promoters have a higher probability of containing a hit. Hence, the PWMs cannot be scored by simply using the hypergeometric score (as in PRIMA). The approach taken is based on binning the promoters by their length. Let $b(p)$ denote the bin of promoter p. For each motif m, use the background set to estimate for each bin the expectation $E_m(b)$ and variance $V_m(b)$ of the number of hits in a promoter in bin b. Test the null hypothesis that for each bin b, the number of hits for m in promoters from the target set T also have this expectation and variance. By the Central Limit Theorem, the total number of hits for m in T is approximately normally distributed, where the expectation E_m and variance V_m of this distribution

can be estimated from the bin information: $E_m = \sum_{p \in T} E_m(b(p))$, $V_m = \sum_{p \in T} V_m(b(p))$. These estimations allow us to derive a normal-based p-value for m.

3. **Modules discovery:** CREME finds all the existing modules in the target set that consists only of the key motifs found in the previous stage. Let r be the number of PWMs in the module and w the maximal length of a region that contains the module (denote such modules as (r, w)-modules). It can be seen that if n equals the number of key motifs, there are n^r possible (r, w)-modules. Instead of enumerating all the possible modules, CREME scans the target set with a window of size w and uses hashing to discover the modules that actually have instances in the target set. CREME searches all instances of a (r, w)-module, but when consecutive instances appear, the efficiency is greater. i.e. an interval of length at most w in some promoter, that contains at least one hit for every motif in the module, and no hit for any other motif. The full algorithm is given in figure 9. The running time of the algorithm is $O(rH)$ where H is the total number of motif hits in all the promoters.



```
$\mathcal{C} \leftarrow \emptyset$ # A hash of motif clusters whose keys are motif sets.
$C_{open} \leftarrow \emptyset$ # A hash of active clusters and their starting positions.
For $i = 1$ to $|\mathcal{M}|$ do:
    Let $h$ be the $i$-th hit in $\mathcal{M}$ occurring at position $pos(h)$.
    For every $(C, start) \in C_{open}$ do:
        If $(pos(h) - start \geq w$ or $h \notin C)$ then Insert($\mathcal{C}$,$C$); Delete($C_{open}$,$C$).
        If $(h \notin C$ and $|C| < r)$ then Insert($C_{open}$,($C \cup \{h\}$, $start$)).
        If $C = \{h\}$ then $start \leftarrow pos(h)$.
    If $\{h\} \notin C_{open}$ then Insert($C_{open}$,($\{h\}$, $pos(h)$)).
For every $C \in C_{open}$ do: Insert($\mathcal{C}$,$C$) # Add remaining active clusters.
Output $\mathcal{C}$.
```

Figure 9: Source: [7]. An algorithm for identifying all motif clusters with at least one consecutive instance in a given sequence. M is the list of all hits, ordered by position. Procedures Insert(H,e) and Delete(H,e) insert/delete an element from a hash table H.

4. **Scoring the modules:** When trying to score the significance of each module, two problems are encountered. The first one is that a module can be over-represented in the target set merely because its component motifs are over-represented themselves. This problem is tackled by choosing a score that takes into account the frequencies of the module's component motifs in the target set. The second problem is the tendency of certain pairs of motifs to have frequent overlapping occurrences merely because certain positions within them have similar nucleotide distributions. This problem is tackled by creating *spaced promoters*. This is done by defining a set of motif hits within a promoter as *independent* if every pair of start positions among these hits differ by at least 4 bases. Then, each promoter is replaced by a set of t spaced promoters, each of which contains a randomly chosen maximal independent subset of the motif hits on the original promoter. These maximal independent subsets are generated by t executions of a randomized greedy algorithm. The idea is that these spaced promoters will contain a good sampling of the spaced occurrences of any module, while eliminating from consideration all module occurrences that are not spaced.
Now, the score of a module is calculated by checking how many times that module occurs on the spaced promoters relatively to the amount of times that would be expected by chance. The score is calculated using a Monte Carlo approach. Many Monte Carlo simulations are done, in each simulation, motif labels are randomly permuted while preserving number of promoters containing each motif. A p-value for each module is obtained by comparing the counts in the original spaced promoters and the permuted spaced promoters.

To validate the results, CREME's specificity was tested by shuffling the promotor model (Figure 11). The results clearly show that the real data scores much higher then the shuffled data, and thus the results of the algorithm are not false positives. We describe the results of CREME in simulation and on three biological data sets:

4.1.1 NFAT-AP1

The transcription factors NFAT and AP-1 coordinately regulate cytokine gene expression in activated T-cells. CREME was tested on a set of 10 genes that contains the module NFAT-AP1 where NFAT always precedes AP1. Only the correct module was reported ($p = 0.01$). After adding 10 promoters that does not contain the module, CREME still finds the NFAT-AP1 module. This demonstrates the high sensitivity of CREME.

4.1.2 Human cell cycle

The target set contained 336 genes. CREME found 16 non-redundant enriched PWMs and 7 non-redundant significant modules. The genes of five out of the seven were coherently expressed (Figure 10). We define the expression coherence of a set of genes S, as the median pairwise similarity calculated from the gene expression data in this set , where the Pearson correlation coefficient is used as a similarity measure. To score the coherence of S, 10000 subsets of size $|S|$ were randomly selected, and the coherence of each subset was computed. The p-value assigned to S was $\frac{k}{10000}$, where k is the number of subsets whose coherence was higher than that of S.

4.1.3 Stress response regulation

The target set contained 253 genes GO annotated as stress response genes. CREME found 20 enriched PWMs and 6 significant modules. Four out of the six were functionally enriched ($p < 0.05$) with some sub-categories of stress response. The functional enrichment was calculated by using a standard hyper-geometric score.

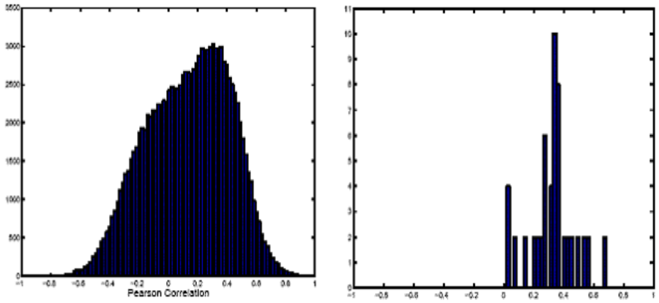


Figure 10: Source: [7]. Left: Histogram of similarity values for all 336 cell cycle regulated genes. Right: Histogram of similarity values for the genes containing a module found by CREME. The median similarity of genes in the module is significantly higher than in the whole set (p=0.04).

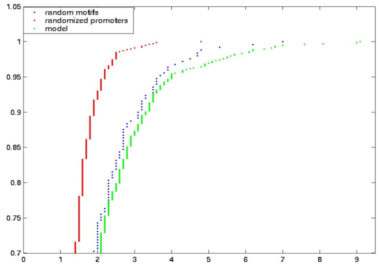


Figure 11: Source: [7]. CREME was tested on a) randomized set of promoters, b) a set of co-regulated genes, c) the same set as b but with random motifs instead of using the key motifs. The graph shows the cumulative distribution function of each test.

References

[1] T.L. Bailey and C. Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *ISMB*, pages 28–36, 1994.

[2] J. Buhler and M. Tompa. Finding motifs using random projections. In *RECOMB*, pages 69–76, 2001.

[3] R. Elkon, C. Linhart, R. Sharan, R. Shamir, and Y. Shiloh. Genome-wide in silico identification of transcriptional regulators controlling the cell cycle in human cells. *Genome Research*, 13:773–780, 2003.

[4] Whitfield M.L., Sherlock G., Saldanha A.J., Murray J.I., Ball C.A., Alexander K.E., Matese J.C., Perou C.M., Hurt M.M., Brown P.O., and Botstein D. Identification of genes periodically expressed in the human cell cycle and their expression in tumors. *Mol Biol Cell*, 13:1977–2000, 2002.

[5] Small S., Kraut R., Hoey T., Warrior R., , and Levine M. Transcriptional regulation of a pair-rule stripe in drosophila. *Gene Develop*, 5:827–839, 1991.

[6] E. Segal, Y. Barash, I. Simon, N. Friedman, and D. Koller. From promoter sequence to expression: A probabilistic framework. *RECOMB*, pages 263–272, 2002.

[7] R. Sharan, I. Ovcharenkoy, A. Ben-Hur, and R. M. Karp. Creme: A framework for identifying cis-regulatory modules in human-mouse conserved segments. *Bioinformatics*, 1:1–9, 2003.

[8] W. Thompson, E. C. Rouchka, and C. E. Lawrence. Gibbs recursive sampler: finding transcription factor binding sites. *Nucleic Acids Research*, 31:3580–3585, 2003.

[9] `http://forkhead2.cgb.ki.se/jaspar/`.

[10] `http://www.biobase-international.com/pages/index.php?id=transfac`.

[11] `http://www.cs.tau.ac.il/˜rshamir/prima/PRIMA.htm/`.

[12] `http://www.ncbi.nlm.nih.gov/geo/`.