BIOINFORMATICS

Vol. 1 no. 1 2003
Pages 1–10

Genome-wide Discovery of Transcriptional Modules from DNA Sequence and Gene Expression

E. Segal[1], R. Yelensky[1] and D. Koller[1]

[1]*Computer Science Dept., Stanford University, Stanford, CA 94305-9010, USA*

ABSTRACT

In this paper, we describe an approach for understanding transcriptional regulation from both gene expression and promoter sequence data. We aim to identify *transcriptional modules* — sets of genes that are co-regulated in a set of experiments, through a common *motif profile*. Using the EM algorithm, our approach refines both the module assignment and the motif profile so as to best explain the expression data as a function of transcriptional motifs. It also dynamically adds and deletes motifs, as required to provide a genome-wide explanation of the expression data. We evaluate the method on two *S. Cerevisiae* gene expression data sets, showing that our approach is better than a standard one at recovering known motifs and at generating biologically coherent modules. We also combine our results with binding localization data to obtain regulatory relationships with known transcription factors, and show that many of the inferred relationships have support in the literature.

Contact: eran@cs.stanford.edu

Keywords: probabilistic models, gene expression, transcriptional regulation.

INTRODUCTION

Many cellular processes are regulated at the transcriptional level, by one or more transcription factors that bind to short DNA sequence motifs in the upstream regions of the process genes. These co-regulated genes then exhibit similar patterns of expression. Given the upstream regions of all genes, and measurements of their expression under various conditions, we could hope to "reverse engineer" the underlying regulatory mechanisms and identify *transcriptional modules* — sets of genes that are co-regulated under these conditions through a common motif or combination of motifs.

In this paper, we take a genome-wide approach for discovering this modular organization, based on the premise that transcriptional elements should "explain" the observed expression patterns as much as possible. We define a probabilistic graphical model (Pearl, 1988), which integrates both the gene expression measurements and the DNA sequence data into a unified model. The model assumes that genes are partitioned into modules, which determine the gene's expression profile. Each module is characterized by a *motif profile*, which specifies the relevance of different sequence motifs to the module. A gene's module assignment is a function of the sequence motifs in its promoter region. However, our model does not assume that all motifs are necessarily active. In fact, as motifs are usually short, there are many genes where a motif is randomly present but does not play a role. Furthermore, our goal is to discover motifs that play a regulatory role in some particular set of experiments; a motif that is active in some settings may be completely irrelevant in others. Our model identifies *motif targets* — genes where the motif plays an active role in affecting regulation in a particular expression data set. These motif targets are genes that have the motif and that are assigned to modules containing the motif in their profile.

Our algorithm is outlined in Fig. 1. It begins by clustering the expression data, creating one *module* from each of the resulting clusters. As the first attempt towards explaining these expression patterns, it searches for a common motif in the upstream regions of genes assigned to the same module. It then iteratively refines the model, trying to optimize the extent to which the expression profile can be predicted transcriptionally. For example, we might want to move a gene g whose promoter region does not match its current module's motif profile, to another module whose expression profile is still a good match, and whose motif profile is much closer. Given these assignments, we could then learn better motif models and motif profiles for each module. This refinement process arises naturally within our algorithm, as a byproduct of the expectation maximization (EM) algorithm for estimating the model parameters.

In general, the motifs learned will not suffice to characterize all of the modules. As our goal is to provide a genome-wide explanation of the expression behavior, our algorithm identifies poorly explained genes in modules and searches for new motifs in their upstream regions. The new motifs are then added to the model and subsequently

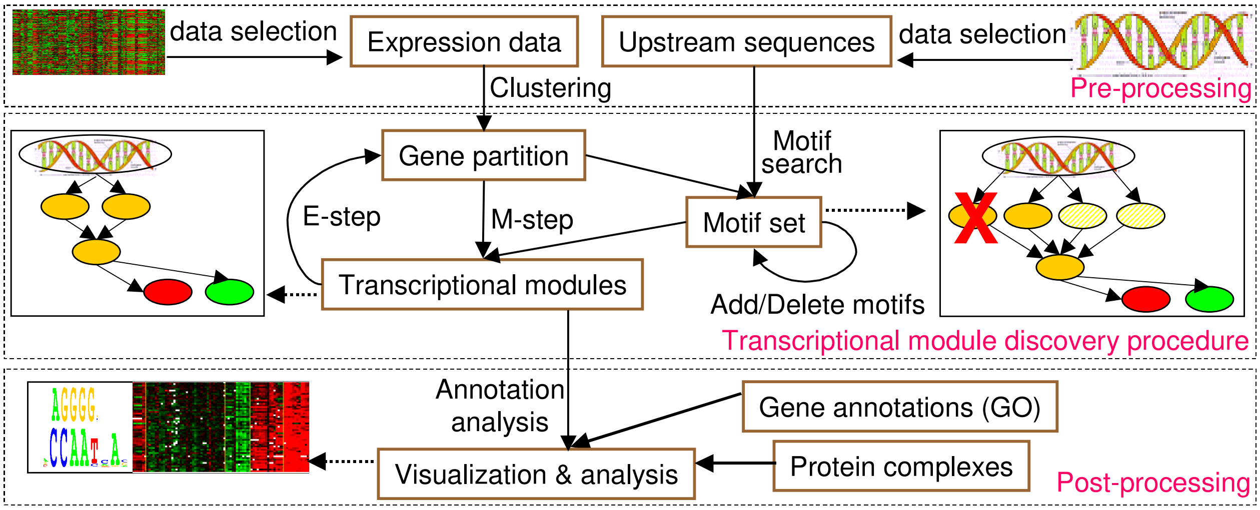


Fig. 1. Schematic flow diagram of our proposed method. The pre-processing step includes selecting the input gene expression and upstream sequence data. The model is then trained using EM, and our algorithm for dynamically adding and deleting motifs. It is then evaluated on additional data sets.

refined using EM. As part of this dynamic learning procedure, some motifs may become obsolete and are removed from the model. The algorithm iterates until convergence, adding and deleting motifs, and refining motif models and module assignments.

Our algorithm has several important advantages over other attempts to relate upstream sequences and expression data. First, we use both expression and sequence data together, requiring that modules display a coherent profile for both. This approach allows us to refine both the cluster assignments and motifs within the same algorithm. In contrast, many approaches (e.g., (Brazma et al., 1998; Liu et al., 2001; Roth et al., 1998; Sinha and Tompa, 2000; Tavazoie et al., 1999)) use gene expression measurements to define clusters of genes that are potentially co-regulated, and then search for common motifs in the upstream regions of the genes in each cluster. The expression analysis and motif finding are thus decoupled, and neither the clusters nor the motifs are re-evaluated once they are learned. Other approaches, e.g., (Bussemaker et al., 2001; Pilpel et al., 2001) work in the opposite direction, first identifying a set of candidate motifs, and then trying to explain the expression using these motifs. However, these approaches use a prespecified set of motifs, which are never adapted during the algorithm.

Our approach is based on the framework of Segal et al. (2002) but extends it in several important directions. First, their approach made use of DNA localization data, which is not widely available for all organisms and under multiple growth conditions. In contrast, we construct models that are based solely on sequence and expression data, which are much easier to obtain. Second, their approach used a predetermined number of motifs to construct the model. To allow a genome-wide analysis, our algorithm dynamically removes and adds motifs as needed to explain the expression data as a whole. Finally, while the models of Segal et al. (2002) allowed for detection of context specific regulation, the resulting structure is hard to interpret. Our model assigns each gene to one module, facilitating interpretability.

We tested our method on two distinct *S. cerevisiae* expression datasets. We show that our learned models find motifs that account for a much larger fraction of the observed expression patterns in comparison to standard approaches that first cluster the expression profiles and then search for motifs in the upstream regions of the genes in each cluster. Our approach also recovers a much larger number of known motifs. We evaluated the functional coherence of our transcriptional modules using a gene functional annotation database and two protein complex databases that were not given to the model as input. We found enrichment for many more groups in our models compared to standard approaches, suggesting that our transcriptional modules are biologically more accurate. Finally, we used the recent binding assays of Lee *et al.* (2002) to relate the actual transcription factors to the modules they regulate, resulting in a regulatory network; we show that many of the regulatory relationships discovered have support in the literature.

PROBABILISTIC MODEL

The basic entities in our model are the genes in some set $\mathbf{G}$. We assume that the genes are partitioned into a set of K mutually exclusive and exhaustive *transcriptional* *modules*. Thus, each gene is associated with an attribute $M \in \{1, \ldots, K\}$ whose value represents the module to which the gene belongs. We now describe how these modules are related to expression profiles and to motif profiles.

Gene Expression Model. For each gene g in $\mathbf{G}$, we have expression measurements $g.E_1, \ldots, g.E_J$, where $g.E_j$ represents the log ratio mRNA expression level measured for gene g in experiment j. We assume that all of the genes in a single module exhibit the same gene expression behavior, and use the simple yet powerful Naive Bayes model (Cheeseman and Stutz, 1995) to represent this behavior. In this model, as applied in our setting, we assume that the expression measurements are conditionally independent given the module assignment:

$$P(E_1, \ldots, E_J \mid M) = \prod_{j=1}^{J} P(E_j \mid M).$$

As the expression measurements are real-valued, we model each conditional probability distribution $P(E_j \mid M = m)$ using a Gaussian distribution $\mathcal{N}(\mu_{jm}; \sigma_{jm})$.

Motif Model. The second key component in our model is a set of variables that represent the regulation of the gene by motifs. For each gene g, we have a set of binary-valued *Regulates* variables $\mathbf{R} = \{R_1, \ldots, R_L\}$, where $g.R_i$ takes the value *true* if motif i appears in the promoter region of gene g, allowing the motif to play a regulatory role in the gene's expression. We model the motif using a standard *position specific scoring matrix (PSSM)* (Bailey and Elkan, 1994; Roth et al., 1998), which assigns a weight to each position in the motif and each nucleotide $\ell \in \{A, C, G, T\}$; this weight represents the extent to which the nucleotide's presence in this position is associated with the motif.

We use the discriminative PSSM approach of Segal et al. (2002), which trains the PSSM weights to discriminate as much as possible between the presence and the absence of the motif. This approach provides better predictions, and entirely avoids the problems arising from high-frequency but meaningless motifs that are common in many upstream sequences. This model is specified using a standard binary logistic model. We have p position-specific weights $w_j[\ell]$, one for each position j and each letter $\ell \in \{A, C, G, T\}$, and a threshold w_0. For a promoter sequence of length N, we assume that binding occurs once, and with equal probability at each of the $N - p + 1$ possible positions in the sequence. The probability of binding given the sequence is then specified simply as:

$$P(g.R = true \mid S_1, \ldots, S_n) =$$
$$logit\left(\log\left(\frac{w_0}{n-p+1}\sum_{j=1}^{n-p+1}\exp\{\sum_{i=1}^{p} w_i[S_{i+j-1}]\}\right)\right).$$

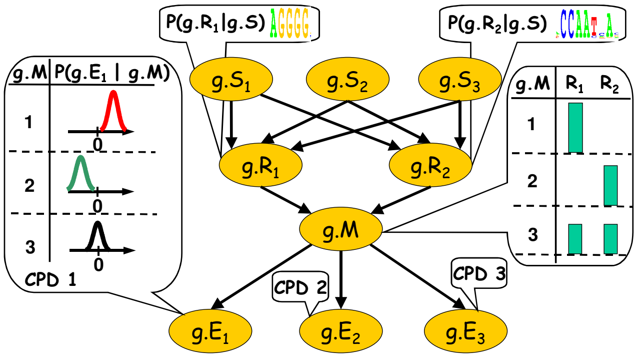


Fig. 2. Illustration of our unified probabilistic model, for a simple example with upstream regions of length 3, two sequence motifs, 3 possible module assignments and 3 expression measurements for each gene

Regulation Model. We define the *motif profile* of a transcriptional module to be a set of weights u_{mi}, one for each motif, such that u_{mi} specifies the extent to which motif i plays a regulatory role in module m. Roughly speaking, the strength of the association of a gene g with a module m is $\sum_{i=1}^{L} g.R_i u_{mi}$. The stronger the association of a gene with a module, the more likely it is to be assigned to it. We model this using a *softmax* conditional distribution, a standard extension of the binary logistic conditional distribution to the multi-class case:

$$P(g.M = \bar{m} \mid R_1 = r_1, \ldots, R_L = r_L) = \frac{\exp\{\sum_{i=1}^{L} u_{\bar{m}i} r_i\}}{\sum_{m'=1}^{K}\exp\{\sum_{i=1}^{L} u_{m'i} r_i\}}.$$

As we expect a motif to be active in regulating only a small set of modules in a given setting, we limit the number of weights $u_{1i}, \ldots, u_{Ki}$ that are non-zero to some $h \ll K$. This restriction results in a sparse weight matrix for $P(M \mid \mathbf{R})$, and ensures that each regulator affects at most h modules. In addition, for interpretability considerations, we require all weights to be non-negative. Intuitively, this means that a gene's assignment to specific transcriptional modules can only depend on features that correpond to the presence of certain motifs and not on the abscence of motifs. For a module m, the set of motifs u_{mi} that are non-zero are called the *motif profile* of m.

Unified Model. These three components, are put together as a probabilistic graphical model, as shown in Fig. 2; the model defines the following joint distribution:

$$P(g.\mathbf{R}, g.M, g.\mathbf{E} \mid g.\mathbf{S}) =$$
$$\prod_{i=1}^{L} P(g.R_i \mid g.\mathbf{S}) \cdot \mathbf{P}(\mathbf{g.M} \mid \mathbf{g.R}) \cdot \prod_{\mathbf{j=1}}^{\mathbf{J}} \mathbf{P}(\mathbf{g.E_j} \mid \mathbf{g.M}),$$

where each of the above conditional probability distributions is parameterized as described in the previous sections.

LEARNING THE MODELS

In the previous section, we presented our probabilistic model. We now turn to the task of learning this model from data. Our data set D consists of a set of genes $\mathbf{G}$, where for each gene $g \in \mathbf{G}$ we have a set of gene expression measurements $g.e_j$ for $j = 1, \ldots, J$ and a DNA sequence $g.\mathbf{S}$ in the upstream region of the transcription start site for g. For this section, we restrict attention to a fixed number of motifs, and address the problem of estimating the model parameters to fit the data. The model parameters to be estimated are: the means and variances of the normal distributions of the expression model, the softmax weights and structure of the module assignments (i.e., which sequence motifs each module depends on), and the PSSM weights for each sequence motif.

We follow the standard approach of *maximum likelihood* estimation: we find the parameters $\boldsymbol{\theta}$ that maximize $P(D \mid \boldsymbol{\theta})$. Our learning task is made considerably more difficult by the fact that both the module assignment $g.M$ and the *Regulates* variables $g.\mathbf{R}$ are unobserved in the training data. In this case, the likelihood function has multiple local maxima, and no general method exists for finding the global maximum. We thus use the *Expectation Maximization (EM)* algorithm (Dempster et al., 1977), which provides an approach for finding a local maximum of the likelihood function.

Starting from an initial guess $\boldsymbol{\theta}^{(0)}$ for the parameters, EM iterates the following two steps. The *E-step* computes the distribution over the unobserved variables given the observed data and the current estimate of the parameters. We use the *hard assignment* version of the EM algorithm, where this distribution is used to select a likely completion of the hidden variables. The *M-step* then re-estimates the parameters by maximizing the likelihood with respect to the completion computed in the E-step. This estimation task differs for the different parts of the model.

E-step: Inferring Modules and Regulation. Our task in the E-step is to compute the distribution over the unobserved data, which in our setting means computing $P(g.M, g.\mathbf{R} \mid g.\mathbf{E}, g.\mathbf{S})$. As genes are assumed to be independent, this computation can be done separately for each gene. However, although the softmax distribution for $P(g.M \mid g.\mathbf{R})$ has a compact parameterization, inference using this distribution is still exponential in the number of *Regulates* variables. Even if only a small number of these variables are associated with any single module, for the purpose of module assignment we need to consider all of the variables associated with any module; this number can be quite large, rendering exact inference intractable.

We devise a simple approximate algorithm for doing this computation, which is particularly well suited for our setting. It exploits our expectation that, while a large number of sequence motifs determine module assignment, only a small number of motifs regulate a particular transcriptional module. Consequently, given the module assignment for a gene, we expect a small number of *Regulates* variables for that gene to take the value *true*. Our approximate algorithm therefore searches greedily for a small number of *Regulates* variables to activate for each module assignment. For each gene g, it considers every possible module assignment m, and finds a good assignment to the *Regulates* variables given that $g.M = m$. This assignment is constructed in a greedy way, by setting $g.R$ variables to *true* one at a time, as long as $P(g.M, g.\mathbf{R}, g.\mathbf{E} \mid g.\mathbf{S})$ improves. The joint setting for $g.M$ and $g.\mathbf{R}$ which gives the overall best likelihood is then selected as the (approximate) most likely assignment. For the remainder of this section, let $g.\bar{m}$ and $g.\bar{r}_1, \ldots, g.\bar{r}_L$ represent the values selected for $g.M$ and $g.R_1, \ldots, g.R_L$ respectively by the E-step. Full details of the algorithm are given in Fig. 3(a).

M-step: Expression Model. Given the assignments of genes to modules as computed in the E-step, the maximum likelihood setting for the parameters of the expression model Gaussian distributions has a closed form solution. Letting N_m be the number of genes assigned to module m, we have that the mean and variance of the Gaussian for experiment j given module assignment m are

$$\mu_{mj} = \frac{1}{N_m} \sum_{g \in \mathbf{G} \,:\, g.\bar{m} = m} g.e_j$$

and

$$\sigma^2_{mj} = \frac{1}{N_m} \sum_{g \in \mathbf{G} \,:\, g.\bar{m} = m} g.e_j^2 - \mu^2_{jm}.$$

M-step: Motif Model. We want the motif model to be a good predictor of the assignment $\bar{r}$ to the *Regulates* variables computed in the E-step. Thus, for each R_i, we aim to find the values of the parameters $w_0, w_j[\ell]$ that maximize the conditional log probability $\sum_{g \in \mathbf{G}} \log P(g.\bar{r}_i \mid g.S_1, \ldots, g.S_n)$. Unfortunately, this optimization problem has no closed form solution, and there are many local maxima. We therefore use a conjugate gradient ascent to find a local optimum in the parameter space. Conjugate gradient starts from an initial guess of the weights $\vec{w}^{(0)}$. As for all local hill climbing methods, the quality of the starting point has a huge impact on the quality of the local optimum found by the algorithm. We therefore initilize the weights using

For each gene $g \in G$
  Set $g.M = 1$
  Set $g.R_i =$ *false* for $1 \leq i \leq L$
  Set $p = P(g.M, g.\mathbf{R} \mid g.\mathbf{S}, \mathbf{g}.\mathbf{E})$
  For $m = 1$ to K // for all modules
    Repeat // Find $g.R_i$ that increases p
      Set $p_{best} = p$
      For $i = 1$ to L // for all regulates variables
        Set $g.R_i =$ *true*
        $p' = P(g.M = m, g.\mathbf{R} \mid g.\mathbf{S}, \mathbf{g}.\mathbf{E})$
        if $p' > p$
          Set $g.M = m$
          Set $p = p'$
        else
          Set $g.R_i =$ *false*
    Until $p_{best} = p$

(a)

Set $U = \{\}$
Set *iteration* $= 0$
Let $V = \{v_{mi}\}_{1 \leq m \leq K, 1 \leq i \leq L}$
Set *MaxScore* $= \max_V$ *Score*$[V]$
  // *MaxScore* = score of unconstrained fit
Set $T =$ *Threshold for closeness to MaxScore*
Repeat
  Set *iteration* $=$ *iteration* $+ 1$
  Let $U' = \{u'_{mi}\}_{1 \leq m \leq K, 1 \leq i \leq L} - U$
  Set $U' = \text{argmax}_{U' \geq 0}$ *Score*$[U', U]$
    // Optimize weights not in U; weights in U fixed
  For $i = 1$ to L // for all regulates variables
    Let $m = \text{argmax}_m \{u'_{mi}\}_{1 \leq m \leq K}$
    Set $U = U \bigcup \{u'_{mi}\}$ // Add new non-zero weight
  Set $U = \text{argmax}_{U \geq 0}$ *Score*$[U, \mathbf{0}]$
    // Reoptimize weights in U; other weights $= 0$
Until *iteration* = *max iteration* or *Score*$[U] >=$ *MaxScore* $- T$

(b)

Delete:
For $i = 1$ to L // for all regulates variables
  Set $U' = U$
  Set $u'_{mi} = 0$ for $1 \leq m \leq K$
  If *Score*$[U] -$ *Score*$[U'] \leq$ *threshold*
    Delete R_i
    Set $U = U'$
Add:
For $m = 1$ to K // for all modules
  Let $\mathbf{G}' = \{\}$
  For each g s.t. $g.\bar{m} = m$
    Set $g.\bar{\mathbf{r}}' = \text{argmax}_{\bar{\mathbf{r}}} P(g.\bar{\mathbf{r}} \mid g.\mathbf{S})$
    Set $g.m' = \text{argmax}_m P(g.M = m \mid g.\mathbf{R} = g.\bar{\mathbf{r}}')$
    If $m' \neq m$
      Set $\mathbf{G}' = \mathbf{G}' \bigcup \{g\}$
  Learn motif with positive set $\mathbf{G}'$
  Add new *Regulates* variable with learned PSSM

(c)

Fig. 3. (a) Search procedure for E-step of EM. (b) Learning the softmax distribution for $P(g.M \mid g.\mathbf{R})$ in the M-step. (c) Procedure for dynamically deleting and adding regulates variables. In (b) and (c), U denotes the non-zero weights of $P_U(g.M \mid g.\mathbf{R})$, and *Score*$[U] = \sum_{g \in \mathbf{G}} \log P_U(g.\bar{m} \mid g.\bar{\mathbf{r}})$.

the method of Barash *et al.* (2001), which efficiently generates motif seeds of length 6–15 and then scores them using the hypergeometric significance test. Each seed produced by this method is then expanded to produce a PSSM of the desired length, whose weights serve as an initialization point for the conjugate gradient procedure.

M-step: Regulation Model. Finally, we consider the task of estimating the parameters for the distribution $P(g.M \mid g.\mathbf{R})$. Our goal is to find a setting for the softmax weights $\{u_{mi}\}_{1 \leq m \leq K, 1 \leq i \leq L}$ so as to maximize the conditional log probability $\sum_{g \in \mathbf{G}} \log P(g.M = g.\bar{m} \mid g.\mathbf{R} = g.\bar{\mathbf{r}})$. Although this optimization does not have a closed form solution, the function is convex in the weights of the softmax. Thus, a unique global maximum exists, which we can find using gradient ascent.

However, as we discussed in the previous section, we also constrain this weight matrix to be sparse and each weight to be non-negative. These constraints lead to more desirable models, but also turn our task into a hard combinatorial optimization problem. We use a greedy selection algorithm, that tries to include non-zero weights for the most predictive motifs for each *Regulates* variable R_i. The algorithm, shown in Fig. 3(b), first finds the optimal setting to the full weight matrix; as we discussed, the optimal setting can be found using gradient ascent. For each variable R_i, it then selects the most predictive motif — the one whose weight is largest — and adds it to the motif profile U, which contains motifs that have non-zero weight. The optimal setting for the weights in U is then found by optimizing these weights, under the constraint that each weight in U is non-negative and the weights not in U must be zero. This problem is also convex, and can be solved using gradient methods. The algorithm then continues to search for additional motifs to include in the profile U: It finds the optimal setting to all weights while holding the weights in U fixed; it then selects the highest weight motifs not in U, adds them to U, and repeats. Weights are added to U until the sparseness limit is reached, or until the addition of motifs to U does not improve the overall score.

DYNAMICALLY ADDING AND REMOVING SEQUENCE MOTIFS

In the previous section, we showed how to optimize the model parameters given a fixed set of motifs. We now wish to devise a dynamic learning algorithm, capable of both removing and adding sequence motifs as part of the learning process. As we learn the models, some motifs may not turn out to be predictive, or redundant given the newly discovered motifs. Conversely, some modules may not be well explained by sequence motifs, so that new motifs should be added.

We add and remove motifs after each completion of the EM algorithm. (Note that EM itself iterates several times between the E-step and the M-step.) To determine whether R_i should be deleted, we compute the conditional log probability $\sum_{g \in \mathbf{G}} \log P(g.m \mid g.\bar{\mathbf{r}})$ both with and without $g.R_i$, leaving the values of other *Regulates* variables fixed. This computation tells us the contribution that R_i makes towards the overall fit of the model. Variables that contribute below a certain threshold are subsequently removed from the model.

We try to add motifs when the current set of motifs does not provide a satisfactory explanation of the expression data: when there are genes for which the sequence predictions do not match the expression profile. We define the *residual* for a transcriptional module m to be the set of genes that are assigned to module m in the E-step, but would not be assigned to m based on the sequence alone. We determine the sequence-only assignment of each gene by computing

$$g.\bar{\mathbf{r}}' = \text{argmax}_{\mathbf{r}} P(g.\mathbf{r} \mid g.\mathbf{S})$$

and

$$g.m' = \text{argmax}_m P(g.M = m \mid g.\mathbf{R} = g.\bar{\mathbf{r}}').$$

We then attempt to provide a better prediction for the residual genes by adding a sequence motif that is trained to match these genes. Once a new *Regulates* variables is added, it becomes part of the model and its assignment and parameterization is adapted as part of the next EM iteration, as described in the previous section. This process tests whether a new motif contributes to the overall model fit, and may assign it a non-zero weight. Importantly, a motif that was trained for the residuals of one module often gets non-zero weights for other modules as well, allowing the same motif to participate in multiple modules. Full details of the algorithm are given in Fig. 3(c).

RESULTS

Models Learned. We evaluated our method separately on two different *S. Cerevisiae* gene expression datasets, one consisting of 173 microarrays, measuring the responses to various stress conditions (Gasch *et al.*, 2000), and another consisting of 77 microarrays, measuring expression during cell cycle (Spellman *et al.*, 1998). We also obtained the 500bp upstream region of each gene (sequences were retrieved from SGD (Cherry *et al.*, 1998)).

The EM algorithm requires an initial setting to all parameters. We use the standard procedure for learning motifs from expression data to initialize the model parameters: we first cluster the expression profiles, resulting in a partition of genes to clusters, and then learn a motif for each of the resulting clusters. For clustering the expression, we use the probabilistic hierarchical clustering algorithm of Segal *et al.* (2001). For learning motifs, we use the motif finder described above. To specify the initial parameterization of our model, we treat these clusters and motifs as if they were the result of an E-step, assigning a value to all of the variables $g.M$ and $g.\mathbf{R}$, and learn the model parameters as described above.

For the stress data, we use 1010 genes which showed a significant change in expression, excluding members of the generic stress response cluster (Gasch *et al.*, 2000). We initialized 20 modules using standard clustering, and learned the associated 20 sequence motifs. From this starting point, the algorithm converged after 5 iterations, of an EM step and a motif addition/deletion step, resulting in a total of 49 motifs. For the cell cycle data, we learned a model with 15 clusters over the 795 cell cycle genes defined in (Spellman *et al.*, 1998). The algorithm converged after 6 iterations, ending with 27 motifs.

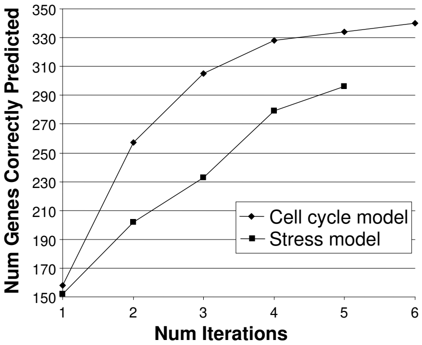


Fig. 4. Number of genes whose module assignment can be correctly predicted based on sequence alone, where a correct prediction is one that matches the module assignemnt when the expression is included. Predictions are shown for each iteration of the learning procedure.

Predicting Expression from Sequence. Our approach aims to explain expression data as a function of sequence motifs. Hence, one metric for evaluating a model is its ability to associate genes with modules based on their promoter sequence alone. Specifically, we compare the module assignment of each gene when we consider only the sequence data to its module assignment considering both expression and sequence data. Fig. 4 shows the total number of genes whose expression-based module assignment is correctly predicted using only the sequence, as the algorithm progresses and sequence motifs are added. As can be seen, the predictions improve across the learning iterations, and significantly outperform the standard approach (which is iteration 1). Ultimately, our model converges to 340 and 296 genes correctly predicted in the cell cycle and stress models, respectively, compared to 158 and 152 for the standard approach.

Gene Expression Coherence. These results indicate that our model assigns genes to modules such that genes assigned to the same module are generally enriched for the same motifs. However, we can achieve such an organization by simply assigning genes to modules based only on their sequence, while entirely ignoring the expresssion data. To verify the quality of our modules relative to gene expression data, we define the *expression coherence* of a module to be the average Pearson correlation between each pair of genes assigned to it, where the Pearson correlation is

$$\mathit{Pearson}(g_i.\mathbf{E}, g_j.\mathbf{E}) = \frac{1}{L}\sum_{l=1}^{L}\frac{(g_i.E_l - \mu_i)}{\sigma_i}\frac{(g_j.E_l - \mu_j)}{\sigma_j},$$

where μ_i, σ_i are the mean and standard deviation of

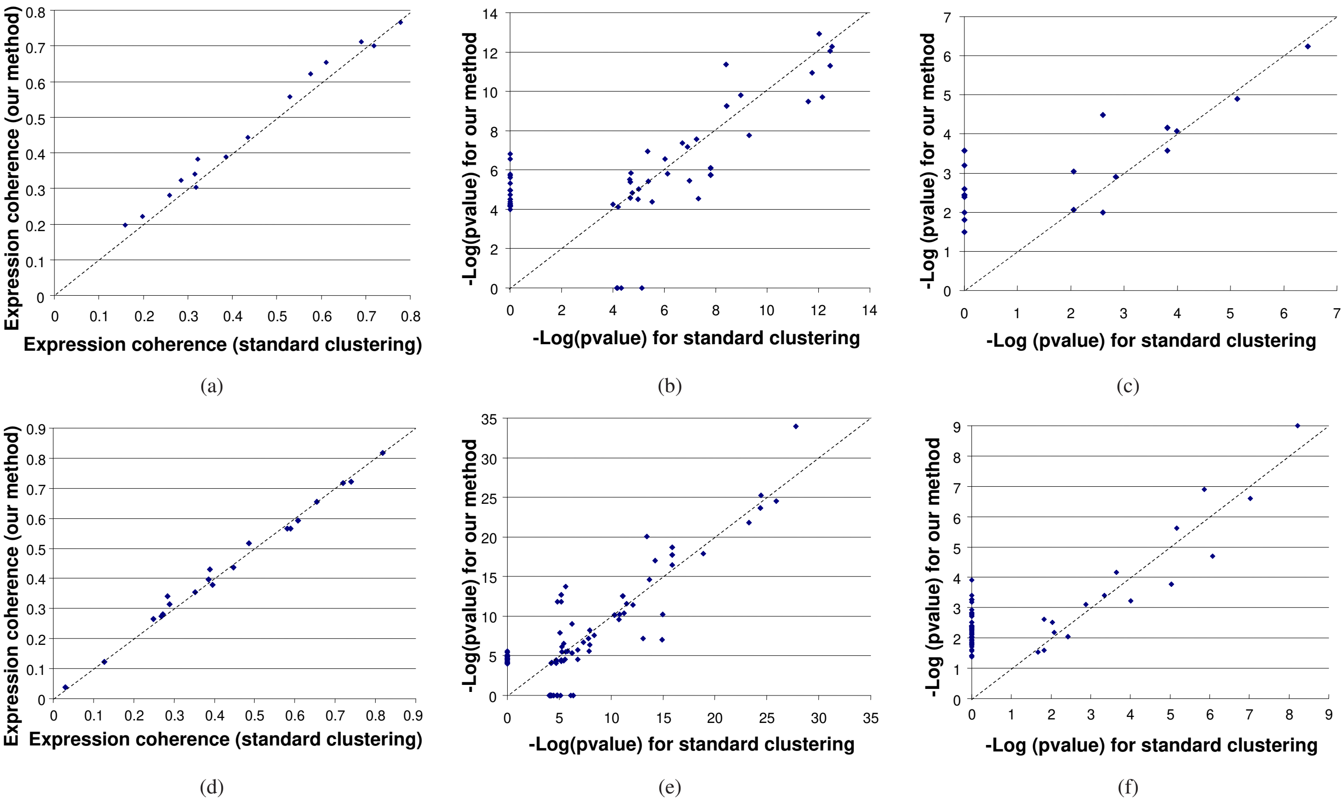


Fig. 5. Comparison of standard clustering and the proposed method. (a)-(c) are for the cell cycle dataset (Spellman *et al.*, 1998) and (d)-(f) are for the stress expression dataset (Gasch *et al.*, 2000). (a),(d) Comparison of the expression coherence for each inferred module (or cluster in the standard clustering model). (b),(e) Comparison of enrichment of the targets of each motif for functional annotations from the GO database. For each annotation, the largest negative log p-value obtained from analyzing the targets of all motifs is shown. (c),(f) Comparison of enrichment of the targets of each motif for protein complexes. For each protein complex, shown is the largest negative log p-value obtained from any of the motifs.

the entries in $g_i.\mathbf{E}$. Fig. 5(a,d) compares the expression coherence of our modules to those built from standard clustering for the cell cycle and stress data, showing identical coherence of expression profiles. For the cell cycle data, there was even a slight increase in the coherence of the expression profiles for our model. Thus, our model results in clusters that are more enriched for motifs, while achieving the same quality of expression patterns as standard clustering which only tries to optimize the expression score.

Coherence of Motif Targets. As we discussed, the motif profile characterizing a module allows us to define a notion of *motif targets* — genes that contain the motif, and where the motif plays a role in its expression profile, i.e., those assigned to a module whose motif profile contains the motif. In the standard clustering model, we can define the targets of a motif to be those genes that have the motif and belong to the cluster from which the motif was learned.

We tested whether our motif targets correspond to functional groups, by measuring their enrichment for genes in the same functional category according to the gene annotation database of GO (Ashburner *et al.*, 2000). We used only GO categories with 5 or more genes associated with them, resulting in 537 categories. For each annotation and each motif, we computed the fraction of genes in the targets of that motif associated with that annotation and used the hypergeometric distribution to calculate a p-value for this fraction, and took p-value $<$ 0.05 to be significant. We compared, for both expression data sets, the enrichment of the motif targets for GO annotations between our model and standard clustering. We found many annotations that were enriched in both models. However, there were 24 and 29 annotations that were significantly enriched in our cell cycle and stress models, respectively, that were not enriched at all in the standard clustering model, compared to only 4 and 14 categories only enriched in the standard clustering model for these respective models. Among those categories enriched only in our model were carbohydrate catabolism, cell wall organization and galactose metabolism, all of which are processes known to be active in response to various stress conditions that we can now characterize by sequence motifs. A full comparison of the GO enrichment for both datasets is shown in Fig. 5(b,e).

Since functional categories do not necessarily correspond to co-regulation groups, we also tested the enrichment of our motif targets for protein complexes, as compiled experimentally in the assays of Gavin *et al.* (2002) and Ho *et al.* (2002), consisting of 590 and 493 complexes, respectively. The member genes of protein complexes are often co-regulated and we thus expect to find enrichment for them in our motif targets. We associated each gene with the complexes it is assigned to in each protein complex dataset and computed the *p*-value of the enrichment of the targets of each motif for each complex, as we did above for the GO annotations. The results for the cell cycle and stress datasets are summarized in Fig. 5(c,f), showing much greater enrichment of our motif targets than the targets of the motifs identified using the standard approach, with 63 and 10 complexes significantly enriched only in our model, and no complexes only enriched in the standard approach, for the stress and cell cycle models, respectively.

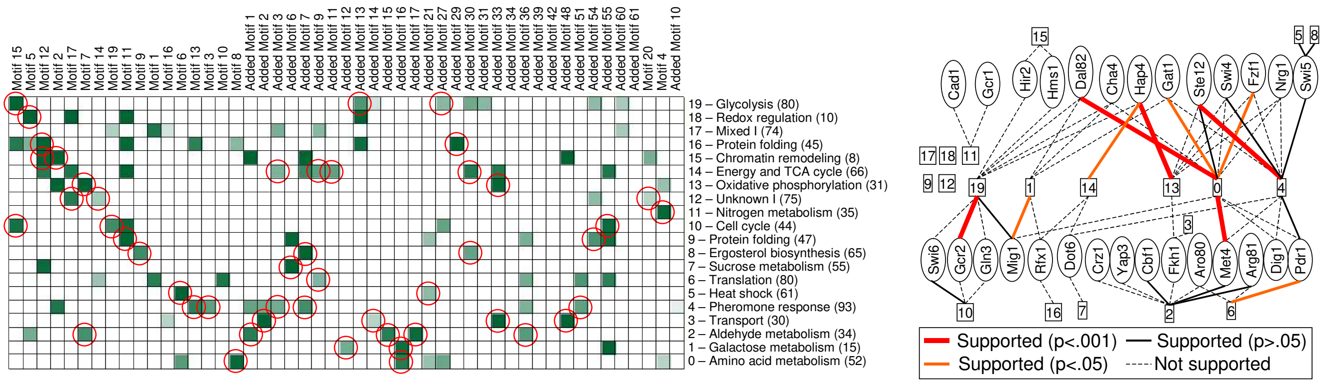
Motifs and Motif Profiles. We compared the motifs we identified with to motifs from the literature. Of the 49 motifs learned for the stress model, 22 are known, compared to only 10 known motifs learned using the standard approach. For the cell cycle model, 15 of the 27 learned motifs are known, compared to only 8 known motifs learned using the standard approach. Many of the known motifs identified, such as the stress element STRE, the heat shock motif HSF and the cell cycle motif MCM1, are also known to be active in the respective datasets.

A powerful feature of our approach is its ability to characterize modules by motif profiles. This ability is particularly important for higher eukaryotes, in which regulation often occurs through multiple distinct motifs. To illustrate the motif profiles found by our approach, we found for each motif all modules enriched for the presence of that motif. This was done by associating each gene with the motifs in its upstream region, and then computing the *p*-value of the enrichment of the member genes of each module. Fig. 6(a) shows all the module-motif pairs in which the module was enriched for the motif with *p*-value < 0.05. In addition, the figure indicates (by red circles) all pairs in which the motif appears in the module's motif profile. As can be seen, many of profiles contain multiple motifs, and many motifs were used by more than one module. Even though modules share motifs, each module is characterized by a unique combination of motifs.

Inferring Regulatory Networks. Identifying the active motifs is a significant step towards understanding the regulatory mechanisms governing gene expression. However, we would also want to know the identity of the transcription factor (TF) molecules that bind to these sequence motifs. We used the DNA binding assays of Lee *et al.* (2002), that directly detect to which promoter regions a particular TF binds *in vivo*, and associated TFs with the motifs we learned. For each motif, we computed the fraction, among the motif targets, of genes bound by each TF, as measured in the data of Lee *et al.*. We used the hypoergeometric distribution to assign a pvlaue to each such fraction and took p-value < 0.05 to be significant. Inspection of the significant associations showed that, in most cases, there was a unique motif that was significant for the TF and that a high fraction ($> .5$) of the TF's binding targets were among the motif target genes.

Based on this strong association between TFs and motifs, for each such TF-motif pair, we predicted that the TF regulates all the modules that are characterized by the motif. By combining all associations, we arrived at the regulatory network shown in Fig. 6(b). Of the 106 transcription factors measured in Lee *et al.*, 28 were enriched in the targets of at least one motif and were thus added to the resulting network. Of the 20 modules, 16 were associated with at least one TF. To validate the quality of the network, we searched the biological literature and compiled a list of experimentally verified targets for each of the 28 TFs in our network. We then marked each association between a TF and a module as *supported* if the module contains at least one gene that the TF is known to regulate from biological experiments. As current knowledge is limited, there are very few known targets for most TFs. Nevertheless, we found support for 21 of the 64 associations. We also computed the *p*-value for each supported association between a TF and a module, using the binomial distribution with probability of success $p = t/N$, where K is the total number of known targets for the TF and N is the total number of genes (1010). The *p*-value is then $P(X \geq \ell \mid X \sim B(p, n))$, where ℓ is the total number of known targets of the regulator in the supported module and n is the number of genes in the supported module. The resulting *p*-values are shown in Fig. 6(b) by edge thickness and color.

We assigned a name to each module based on a concise summary of its gene content (compiled from both gene annotation and literature). The regulatory network thus contains predictions for the processes regulated by each TF, where for each association the prediction includes the motif through which the regulation occurs. In many cases, our approach recovered coherent biological processes along with their known regulators. Examples of such associations include: Hap4, the known activator of oxidative phosphorylation, with the oxidative phosphorylation module (13); Gcr2, a known positive regulator of glycolysis, with the glycolysis module (19); Mig1, a glucose repressor, with the galactose metabolism module



(a) (b)

Fig. 6. (a) Matrix of motifs vs. modules for the stress data, where a module-motif entry is colored if the member genes of that module were enriched for that motif with p-value < 0.05. The intensity corresponds to the fraction of genes in the module that had the motif. Entries in the module's motif profile are circled in red. Modules were assigned names based on a summary of their gene content. (b) Regulatory network inferred from our model using the DNA binding assays of Lee *et al.*. Ovals correspond to transcription factors and rectangles to modules (see (a) for module names).

(1); Ste12, involved in regulation of pheromone pathways, with the pheromone response module (4); and Met4, a positive regulator of sulfur amino acid metabolism, with the amino acid metabolism module (0).

CONCLUSIONS

We presented a unified probabilistic model over both gene expression and sequence data, whose goal is to identify transcriptional modules and the regulatory motif binding sites that control their regulation within a given set of experiments. Our results indicate that our method discovers modules that are both highly coherent in their expression profiles and significantly enriched for common motif binding sites in upstream regions of genes assigned to the same module. A comparison to the common approach of constructing clusters based only on expression and then learning a motif for each cluster shows that our method recovers modules that have a much higher correspondence to external biological knowledge of gene annotations and protein complex data.

ACKNOWLEDGEMENTS

This work was supported by the National Science Foundation, grant ACI-0082554. Eran Segal was also supported by a Stanford Graduate Fellowship (SGF).

REFERENCES

Ashburner,M., C.A.Ball, J.A.Blake, D.Botstein, H.Butler, J.M.Cherry, A.P.Davis, K.Dolinski, S.S.Dwight, J.T.Eppig, M.A.Harris, D.P.Hill, L.Issel-Tarver, A.Kasarskis, S.Lewis, J.C.Matese, J.E.Richardson, M.Ringwald, G.M.Rubin, and G.Sherlock (2000) Gene ontology: tool for the unification of biology. the gene ontology consortium. *Nature Genetics 25*, 25–9.

Bailey,T.L. and C.Elkan (1994) Fitting a mixture model by expectation maximization to discover motifs in biopolymers. In *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, Volume 2, pp. 28–36.

Barash,Y., G. Bejerano, and N. Friedman (2001) A simple hypergeometric approach for discovering putative transcription factor binding sites. In *Algorithms in Bioinformatics*, Number 2149 in LNCS, pp. 278–293.

Brazma,A., I. Jonassen, J. Vilo, and E. Ukkonen (1998) Predicting gene regulatory elements in silico on a genomic scale. *Genome Res. 8*, 1202–15.

Bussemaker,H., H. Li, and E. Siggia (2001) Regulatory element detection using correlation with expression. *Nature Genetics 27*, 167–71.

Cheeseman,P. and J. Stutz (1995) Bayesian classification (AutoClass): Theory and results. In *Advances in Knowledge Discovery and Data Mining*, pp. 153–180. Menlo Park, CA: AAAI Press.

Cherry,M.J. *et al.* (1998) Sgd: Saccharomyces genome database. *Nucleic Acid Research 26*, 73–9.

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

Gasch,A. P., P.T.Spellman, C.M.Kao, O.Carmel-Harel, M.B.Eisen, G.Storz, D.Botstein, and P.O.Brown (2000) Genomic expression program in the response of yeast cells to environmental changes. *Mol. Bio. Cell 11*, 4241–4257.

Gavin A.C., M.Bosche, R.Krause, P.Grandi, M.Marzioch, A.Bauer, J.Schultz, J.M.Rick, A.M.Michon, C.M.Cruciat, M.Remor, C.Hofert, M.Schelder, M.Brajenovic, H.Ruffner, A.Merino, K.Klein, M.Hudak, D.Dickson, T.Rudi, V.Gnau, A.Bauch, S.Bastuck, B.Huhse, C.Leutwein, M.A.Heurtier, R.R.Copley, A.Edelmann, E.Querfurth, V.Rybin, G.Drewes, M.Raida, T.Bouwmeester, P.Bork, B.Seraphin, B.Kuster, G.Neubauer, G.Superti-Furga (2002) Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature 415*,

141–7.
Ho Y., A.Gruhler, A.Heilbut, G.D.Bader, L.Moore, S.L.Adams, A.Millar, P.Taylor, K.Bennett, K.Boutilier, L.Yang, C.Wolting, I.Donaldson, S.Schandorff, J.Shewnarane, M.Vo, J.Taggart, M.Goudreault, B.Muskat, C.Alfarano, D.Dewar, Z.Lin, K.Michalickova, A.R.Willems, H.Sassi, P.A.Nielsen, K.J.Rasmussen, J.R.Andersen, L.E.Johansen, L.H.Hansen, H.Jespersen, A.Podtelejnikov, E.Nielsen, J.Crawford, V.Poulsen, B.D.Sorensen, J.Matthiesen, R.C.Hendrickson, F.Gleeson, T.Pawson, M.F.Moran, D.Durocher, M.Mann, C.W.Hogue, D.Figeys, M.Tyers (2002) Systematic identification of protein complexes in saccharomyces cerevisiae by mass spectometry. *Nature 415*, 180–3.
Lee,T. N.J.Rinaldi, F.Robert, D.T.Odom, Z.Bar-Joseph, G.K.Gerber, N.M.Hannett, C.T.Harbison,C.M.Thompson, I.Simon, J.Zeitlenger, E.G.Jennings, H.L.Murray, D.B.Gordon, B.Ren, J.J.Wyrick, J.B.Tagne, T.L.Volkert, E.Fraenkel, D.K.Gifford, and R.A.Young (2002) Transcriptional regulatory networks in saccharomyces cerevisiae. *Science 298*, 824–7.
Liu,X., D. Brutlag, and J. Liu (2001) Bioprospector: discovering conserved dna motifs in upstream regulatory regions of co-expressed genes. In *Pac. Symp. Biocomput.*, pp. 127–38.
Pearl,J. (1988) *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann.
Pilpel,Y., P. Sudarsanam, and G. Church (2001) Identifying regulatory networks by combinatorial analysis of promoter elements. *Nature Genetics 29*, 153–9.
Roth,F., P. Hughes, J.D. Estep, and G. Church (1998) Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation. *Nat. Biotechnol. 16*, 939–945.
Segal,E., Y. Barash, I. Simon, N. Friedman, and D. Koller (2002) From sequence to expression: A probabilistic framework. In *Proc. RECOMB*, pp. 263–272.
Segal,E., B. Taskar, A. Gasch, N. Friedman, and D. Koller (2001) Rich probabilistic models for gene expression. *Bioinformatics 17*(Suppl 1), S243–52.
Sinha,S. and M. Tompa (2000) A statistical method for finding transcription factor binding sites. In *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, Volume 8, pp. 344–54.
Spellman,P.T., G.Sherlock, M.O.Zhang, V.R.Iyer, K.Anders, M.B.Eisen, P.O.Brown, D.Botstein, and B.Futcher (1998) Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Mol. Biol. Cell 9*(12), 3273–97.
Tavazoie,S., J. D. Hughes, M. J. Campbell, R. J. Cho, and G. M. Church (1999) Systematic determination of genetic network architecture. *Nat Genet 22*(3), 281–5. Comment in: Nat Genet 1999 Jul;22(3):213-5.