MODULEFINDER: A TOOL FOR COMPUTATIONAL DISCOVERY OF *CIS* REGULATORY MODULES

ANTHONY A. PHILIPPAKIS[†,1,3,4], FANGXUE SHERRY HE[†,1,3],
MARTHA L. BULYK[*,1,2,3,4]

[1]Division of Genetics, Department of Medicine, [2]Department of Pathology,
[3]Harvard/MIT Division of Health Sciences & Technology (HST), and
[4]Harvard University Biophysics Program
Brigham & Women's Hospital and Harvard Medical School
Boston, MA 02115
Email: {aphilippakis, mlbulyk}@receptor.med.harvard.edu, sherryhe@mit.edu

Regulation of gene expression occurs largely through the binding of sequence-specific transcription factors (TFs) to genomic binding sites (BSs). We present a rigorous scoring scheme, implemented as a *C* program termed "ModuleFinder", that evaluates the likelihood that a given genomic region is a *cis* regulatory module (CRM) for an input set of TFs according to its degree of: (1) homotypic site clustering; (2) heterotypic site clustering; and (3) evolutionary conservation across multiple genomes. Importantly, ModuleFinder obtains all parameters needed to appropriately weight the relative contributions of these sequence features directly from the input sequences and TFBS motifs, and does not need to first be trained. Using two previously described collections of experimentally verified CRMs in mammals and in fly as validation datasets, we show that ModuleFinder is able to identify CRMs with great sensitivity and specificity.

1. Introduction

Recent technological advances have enabled both the sequencing of a large number of genomes and the generation of expansive gene expression datasets. Still, little is known about how these gene expression patterns are precisely regulated through the binding of sequence-specific transcription factors (TFs) to their DNA binding sites (BSs). Of particular interest is the organization of TF binding sites (TFBSs) into *cis* regulatory modules (CRMs) that coordinate the complex spatio-temporal patterns of gene expression, and to use that information to identify the CRMs themselves. Mapping TFs to their target CRMs, however, is significantly complicated in higher eukaryotic genomes by the large proportion of non-protein-coding sequence. Since a typical TFBS can be as short as ~5 base pairs (bp), matches to its motif occur frequently by chance alone, with many of these occurrences presumably not acting to modulate gene expression. Therefore, a central challenge that must be overcome is distinguishing functional TFBSs from spurious motif matches.

[†] These authors contributed equally.
[*] Corresponding author.

To date, three indicators have been used to identify functional TFBSs. First, functional BSs for some TFs tend to occur in clusters, with multiple BSs occurring in close proximity (homotypic clustering). Second, searching for clusters containing BSs for 2 or more TFs that are believed to co-regulate can enrich for likely CRMs (heterotypic clustering). Finally, functional TFBSs are frequently conserved across evolutionarily divergent organisms[1]. Cross-species sequence conservation in particular has enormous potential for filtering sequence space, as many genomes have recently been sequenced, and many more are slated to be sequenced (http://www.genome.gov/10002154). The discriminatory power of phylogenetic footprinting for identifying *cis* regulatory elements is therefore expected to continue to increase through the use of more genomes[2-6]. In order to appropriately incorporate information on conservation across multiple genomes, however, a measure of TFBS conservation is required that weights each alignment genome according to its evolutionary distance not only from the query genome, but also relative to the other alignment genomes. For example, given a candidate TFBS in the human genome, observing conservation in chicken should be weighted more heavily than conservation in mouse, as mouse is evolutionarily closer to human. Moreover, if the candidate site were also conserved in rat, then this additional conservation should be weighted only slightly, given the evolutionary proximity of mouse and rat.

While numerous groups have developed approaches for the prediction of CRMs, none is optimized for practical applications. Specifically, many approaches[7-9] have been based on binary scoring schemes, wherein all regions containing a threshold number of occurrences for a given combination of TFBSs are returned. These approaches suffer from the limitation that they do not prioritize among the predictions, an important feature for experimentalists as only a limited number of candidate CRMs can feasibly be validated. Additionally, the threshold value determined in any given biological system is unlikely to be generalizeable from one set of TFs and CRM type to another; thus, the appropriate discriminatory criterion must be re-discovered with each application. Alternatively, among existing continuous scoring schemes, many require large training sets[10,11]. Such approaches cannot be applied to a system in which there are only a handful of known examples, as is frequently the case in practical applications. Finally, among approaches that employ continuous scoring schemes and do not require training[12-15], most do not systematically integrate BS clustering and conservation. We are aware of only one other approach that combines all three indicators[16], but it is computationally rather slow and requires the user to specify a single sequence window size for the search. Since CRMs are known to vary greatly in size, a scoring scheme is needed that evaluates clustering and conservation over windows of varying sizes[15].

We have developed a statistically rigorous scoring scheme that for any given genomic region integrates into a single score the degree of: (1) homotypic clustering; (2) heterotypic clustering; and (3) evolutionary conservation across multiple genomes. Similar to programs such as BLAST[17], our score is an objective measure of the statistical significance of the observed degree of clustering and conservation that is independent of the genome and TFBSs under consideration. Thus, the scoring scheme obtains all parameters needed to appropriately weight the relative contribution of each input alignment and TFBS motif directly from the sequences and motifs themselves, and so does not need to first be trained. We have implemented this scoring scheme as a *C* program called "ModuleFinder," that is algorithmically efficient and has an intuitive interface. Using two previously described collections of experimentally verified CRMs (mammalian skeletal muscle[18] and *D. melanogaster* segmentation genes[7]), we show that ModuleFinder is able to identify CRMs with ~95% sensitivity and ~95% specificity.

2. Methods

Methods that evaluate the overall degree of conservation for a given region have been successful in identifying *cis* regulatory elements in metazoan genomes[2,6]; they do not, however, necessarily identify the CRMs through which a given set of TFs exert their regulatory roles (i.e., the TFs' "target" CRMs). Since our ultimate goal is to identify candidate CRMs that are bound by a given set of TFs, we have developed a scoring scheme that specifically considers the conservation of a particular set of TFBSs comprising a given transcriptional regulatory model. For this, we developed a novel statistical framework that builds on earlier work. Blanchette *et al.* stated the substring parsimony problem and presented a rigorous and efficient algorithmic procedure for solving it[19]; this model was applied to the identification of candidate DNA motifs. Moses *et al.* used mixture models to evaluate conservation within a tree, and applied it to the identification of candidate DNA motifs from sets of co-expressed genes[20]; this was similar to an approach given by Prakash *et al.*[21] Here we present a related approach for identifying candidate CRMs from input TFBS motifs.

2.1. Scoring Scheme

We define a *word* to be a short sequence on the DNA alphabet {A,C,G,T}, and a *motif* to be a collection of words all of the same length. ModuleFinder takes as input a collection of arbitrarily many motifs $\{m_1 \ldots m_m\}$, where each motif m_i is composed of arbitrarily many words of length l_i. It also takes as input a set of sequences $G = \{g_1,\ldots g_n\}$ corresponding to genomic regions that are to be searched for instances of these motifs, as well as two sets of genomic sequences, $A = \{a_1,\ldots,a_n\}$ and $B = \{b_1,\ldots,b_n\}$, extracted from evolutionarily divergent organisms and then aligned to the sequences of G. Here, we primarily illustrate the scoring scheme for the case of two alignment genomes, but include comments on the extension to fewer or more alignments. For any g_j, let $g_{j,k}$ denote the base at the kth position and $(g_{j,k} \ldots g_{j,k+l})$ denote the subsequence of length l beginning at position k. If there is a match to a given motif m_i at position k of sequence g_j, we define it to be *conserved in A* (respectively, *B*), if it is true that the subsequence ($a_{j,k}...a_{j,k+l}$) (respectively, ($b_{j,k}...b_{j,k+l}$)) is also a word in motif m_i. Note that we are not assuming that $g_{j,k}...g_{j,k+l} = a_{j,k}...a_{j,k+l}$, but merely that they are both words in m_i.

Our basic approach is to scan each sequence in G with a series of nested windows (i.e., overlapping windows of differing sizes). In each window we count the number of occurrences of each motif and the number of these that are conserved in A and B. We then evaluate the likelihood of observing this number of matches and conserved matches under the appropriate null hypothesis, and return those windows that are statistically significant. Specifically, let $X = (X_1,\ldots, X_m)$ be the vector whose components indicate the number of occurrences for each motif individually in a given window, and let $Y = (Y_1,\ldots, Y_m)$ and $Z = (Z_1,\ldots, Z_m)$ be the corresponding vectors indicating that Y_i and Z_i out of X_i occurrences are conserved in A and B, respectively. The window score is obtained by finding the probability of observing (X,Y,Z). This quantity will vary according to the likelihood of conservation in A and/or B, the motif frequency, and the window width. Thus, this probability can be represented by:

$$P_{\Gamma,\alpha,w}(X,Y,Z) \tag{1}$$

where Γ parameterizes conservation likelihood, α parameterizes motif frequencies, and w is the window width. Observe that:

$$P_{\Gamma,\alpha,w}(X,Y,Z) = P_{\Gamma}(Y,Z \mid X)P_{\alpha,w}(X) \tag{2}$$

where the relevant parameters can be split between terms in the Markov decomposition, as $P_{\alpha,w}(X)$ is unaffected by conservation likelihood, and $P_{\Gamma}(Y,Z \mid X)$ is unaffected by motif frequency and window size.

For a single motif m_i, the term $P_{\alpha_i,w}(X_i)$ of Eq. (2) is the likelihood of observing X_i occurrences under the null hypothesis that the motif matches are distributed at random. This has been proved to be well-approximated by a Poisson distribution, provided the motif occurs infrequently and the words comprising it do not exhibit extensive self-overlap.[22] Thus, $P_{\alpha_i,w}(X_i) = e^{-\lambda_i}(\lambda_i^{X_i} / X_i!)$, where $\lambda_i=\alpha_i*w$. The value of α_i will itself be determined by both the words comprising m_i, as well as genomic word frequencies. To obtain it, we estimate the frequency of each word in m_i by a seventh order Markov approximation based on genomic word frequencies, and then sum these frequencies for all words in the motif.

For multiple motifs, the joint probability is given by assuming independence:

$$P_{\alpha,w}(X_1,\ldots X_m) = \prod_{i=1}^{m}\left(P_{\alpha_i,w}(X_i)\right)$$

This is a simplifying assumption to make the computation tractable; the error in this approximation has, however, been proved to be bounded[22].

The computation of the second term of Eq. (2), $P_\Gamma(Y,Z \mid X)$, is complicated by two factors. First, the score must reflect not only the evolutionary distances of *A* and *B* to *G*, but also the distances of *A* and *B* to each other. Thus, *Γ* must re-parameterize $P_\Gamma(Y,Z \mid X)$ so that it becomes smaller as *A* and *B* grow more distant from *G*, and as the correlation between *A* and *B* decreases. Second, the quantity $P_\Gamma(Y,Z \mid X)$ will depend not only on the phylogeny of *A*, *B* and *G*, but also on the degeneracy of the motifs m_i. Since we have defined a given motif match to be conserved in *A* or *B* if there is a motif occurrence (but not necessarily an exact word match) at the same position in these aligned sequences, a more degenerate motif has a greater likelihood of being conserved.

We account for these difficulties as follows. Define $\Gamma^1_{A,B}$ to be the covariance matrix representing the relative proportions of *A* and *B* that can be aligned against *G*; thus, $\Gamma^1_{0,0}$ gives the proportion of sequence in *G* for which neither *A* nor *B* could be aligned, $\Gamma^1_{1,0}$ and $\Gamma^1_{0,1}$give the proportion for which either *A* or *B* (but not both) could be aligned, and $\Gamma^1_{1,1}$ gives the proportion for which both *A* and *B* could be aligned. Similarly, for each motif m_i, define $\Gamma^{i,2}_{A,B}$ to be the covariance matrix representing the relative likelihoods of exact conservation of l_i positions (i.e., $(g_{j,k}\ldots g_{j,k+l}) = (a_{j,k}\ldots a_{j,k+l})$) in *A* and/or *B*. Here, we have observed non-independence of exact conservation likelihood between adjacent positions, so we model it as a first order Markov chain.

Conservation of a completely degenerate motif is parameterized by $\Gamma^1_{A,B}$, and conservation of a motif composed of a single word is parameterized by $\Gamma^{i,2}_{A,B}$. The parameterization of a generic motif is between these extremes; for this, let $P_{i,j,k}$ be the matrix giving the frequency of nucleotide $j \in \{\text{A,C,G,T}\}$ at position $k \in \{1,\ldots,l_i\}$ in motif m_i, and let E_i be the average entropy of the motif:

$$E_i = -\frac{1}{2l_i}\sum_{k=1}^{l_i}\sum_{j\in\{A,C,G,T\}} P_{i,j,k}\log_2 P_{i,j,k}$$

Hence, E_i=1 for a completely degenerate motif, E_i=0 for a motif composed of a single word, and E_i increases monotonically and smoothly between these extremes as the motif degeneracy increases. Therefore, we take our parameterization of Γ_i for m_i to be a weighted average of $\Gamma^1_{A,B}$ and $\Gamma^{i,2}_{A,B}$:

$$\Gamma^i = E_i\Gamma^{i,1}_{A,B} + (1-E_i)\Gamma^{i,2}_{A,B}$$

We then use Γ_i to compute $P_{\Gamma^i}(Y_i, Z_i \mid X_i)$. In a sequence window containing X_i matches to motif m_i, let a_i be the number that are not conserved in either *A* or *B*, let b_i and c_i be the number conserved in either *A* or *B* (but not both), and let d_i be the number that are conserved in both *A* and *B*. The following equations hold:

$$a_i + b_i + c_i + d_i = X_i \tag{3-5}$$

$$b_i + d_i = Y_i \qquad c_i + d_i = Z_i$$

$P(Y_i,Z_i|X_i)$ is therefore given by the following multinomial:

$$P_{\Gamma^i}(Y_i, Z_i \mid X_i) = \sum \left(\frac{X_i!}{a_i!b_i!c_i!d_i!} \right) \left(\left(\Gamma^i_{0,0}\right)^a \cdot \left(\Gamma^i_{1,0}\right)^b \cdot \left(\Gamma^i_{1,0}\right)^c \cdot \left(\Gamma^i_{1,1}\right)^d \right) \qquad (6)$$

where the summation is performed over all values of a_i, b_i, c_i and d_i satisfying Eqs. (3)-(5). To achieve computational efficiency, we make use of the following 1-dimensional parameterization, where X_i, Y_i and Z_i remain fixed as d_i is varied:

$$a_i = X_i - Y_i - Z_i + d_i \qquad (7\text{-}9)$$

$$b_i = Y_i - d_i \qquad c_i = Z_i - d_i$$

Thus, the summation of Eq. (6) can be performed by simply taking each value of d_i in the range $0 \le d_i \le \min(Y_i, Z_i)$.

If one desires to only input one genome, it is sufficient to set $A=B$. The relevant parameters then simplify, and the preceding multinomial distribution collapses to a binomial distribution with parameter $\gamma_i = \Gamma^i_{1,1}$:

$$P_{\gamma_i}(Y_i \mid X_i) = \binom{X_i}{Y_i} \gamma^{Y_i} (1-\gamma_i)^{X_i - Y_i}$$

This parameterization can also be easily generalized to more than 2 alignment genomes by replacing the matrix Γ^i with an appropriate tensor.

This derived value of $P_{\Gamma,\alpha,w}(X,Y,Z)$ alone is insufficient for determining statistical significance, since a measurement of distance into the appropriate tail of the distribution is also required. Therefore, we perform a summation of $P_{\Gamma,\alpha,w}(X,Y,Z)$ extending from the observed value of (X,Y,Z) and including all values of (X,Y,Z) with an increased degree of clustering and conservation (we use log values to simplify the numerical analysis):

$$\begin{aligned} S_{\Gamma,\alpha,w}(X,Y,Z) &= \log_{10}\left(\sum_{\tilde{X}=X}^{\infty} \sum_{\tilde{Y}=Y}^{\tilde{X}} \sum_{\tilde{Z}=Z}^{\tilde{X}} \prod_{i=1}^{m} \left(P_{\Gamma^i,\alpha_i,w}(\tilde{X}_i, \tilde{Y}_i, \tilde{Z}_i) \right) \right) \\ &= \sum_{i=1}^{m} \log_{10}\left(\sum_{\tilde{X}_i=X_i}^{\infty} \sum_{\tilde{Y}_i=Y_i}^{\tilde{X}_i} \sum_{\tilde{Z}_i=Z_i}^{\tilde{X}_i} P_{\Gamma^i,\alpha_i,w}(\tilde{X}_i, \tilde{Y}_i, \tilde{Z}_i) \right) = \sum_{i=1}^{m} S_{\Gamma^i,\alpha_i,w}(X_i, Y_i, Z_i) \end{aligned} \qquad (10)$$

Therefore, the output score $S_{\Gamma,\alpha,w}(X,Y,Z)$ for a given window is the linear sum of scores for the input motifs, $S_{\Gamma_i,\alpha_i,w}(X_i,Y_i,Z_i)$, where each such term has been automatically weighted so that more degenerate motifs contribute less. Observe also that $S_{\Gamma_i,\alpha_i,w}(X_i,Y_i,Z_i)$=0 if and only if X_i=0, and that $S_{\Gamma_i,\alpha_i,w}(X_i,Y_i,Z_i)$ increases monotonically with increasing values of (X_i,Y_i,Z_i), as desired.

2.2. Implementation and Availability

ModuleFinder has been implemented in *C*. To minimize runtime, we pre-process each sequence of *G* with suffix arrays[23] for efficient searching; additionally, as the algorithm proceeds, a look-up table is kept that contains a list of scores for all observed window sizes w and motif matches (X,Y,Z). ModuleFinder can scan ~120 Mb/hr using window sizes of 300-700 bp with an increment size of 50 bp and one alignment genome on a Pentium 4 computer. The compiled code, along with README files and appropriately formatted genomes and alignments for human, mouse, rat, fly, worm and yeast based on the latest UCSC assemblies[24] are available for download at our website (http://the_brain.bwh.harvard.edu).

Two additional features were included for improved practical applicability. First, it is known that TFs frequently bind to DNA as homo- and hetero-dimers. We have added to ModuleFinder the ability to take pairs of TFBSs as input, along with minimum and maximum spacer lengths between sites. The score of the dimer is computed by evaluating the probability of each component motif as in Eq. (1), then taking the product of these probabilities and summing them over all input spacings. Second, ModuleFinder allows a certain amount of 'wiggle room' to compensate for the potential existence of local misalignments. Specifically, given an input value r, a motif match $(g_{j,k}\ldots g_{j,k+l})$ is considered conserved in A if there is any subsequence of $(a_{j,k-r}\ldots a_{j,k+r+l})$ that is a word in m_i. Although this does increase the likelihood of conservation, the effect is miniscule for small values of r ($1\leq r \leq 5$) and has frequently identified potentially conserved sites that would have been missed otherwise.

3. Results

3.1 Validation of ModuleFinder on human skeletal muscle CRMs

In order to evaluate ModuleFinder, we used a set of positive control regions previously compiled by Wasserman *et al.*[18] This test dataset comprises 27[a] skeletal muscle CRMs that have been demonstrated to direct transcription in skeletal muscle or a suitable cell-culture model system[18]. Each region contains a validated BS for at least one of the following 5 TFs: the Myf family (total of 39 TFBSs in the positive control set), Mef2 (26 TFBSs), SRF (20 TFBSs), Tef (12 TFBSs) and Sp1 (13 TFBSs). Of these 27 regions, 23 are located within 5 kb upstream of translational Start, and 2 within introns. As negative controls, 1000 regions of size 200 bp were randomly selected to positionally match the positive control regions: 852 (=(23/27)*1000) regions were within 5 kb of translational Start for a randomly chosen RefGene[24] gene, and the remaining 148 were within introns. This matching of chromosomal locations was performed as ModuleFinder accounts for local word frequencies, which vary throughout the genome; in particular, promoter regions are known to be GC-rich.

We ran ModuleFinder on the positive and negative control regions with window sizes of 100-200 bp (increment size = 10 bp), using human sequence alone,

[a] The original collection gave 28 genes, but we removed the gene *Rb1* as there were no confirmed TFBSs for the listed TFs.

human/mouse/rat (H/M/R) alignments and human/mouse/chicken alignments (H/M/C) obtained from UCSC Genome Browser (hg16, mm3, rn3, galGal2)[24]. Currently, two alternative strategies for representing TFBSs have been used by various groups in computational searches for CRMs: exact word matches to known BSs[9,15], and position weight matrices (PWMs)[7,10-13], which allow for extrapolation to additional BSs. To determine which of these representations had greater discriminatory power, we performed our searches both ways, using a PWM threshold value of 1 standard deviation (SD) below the motif average[25]. We used a "jack-knife" strategy[11] for these searches, whereby the BSs for each CRM were excluded from the construction of the PWM used to search that CRM, and similarly the exact word matches from each CRM were excluded in the search of that CRM. In addition, since *in vitro* binding experiments had been performed for Mef2[26] and SRF[27], we also added those BSs to both searches.

	Human Alone		**Human/Mouse/Rat**		**Human/Mouse/Chicken**	
	Exact	**PWM**	**Exact**	**PWM**	**Exact**	**PWM**
Sens.	88.9%	92.6%	92.6%	96.3%	92.6%	92.6%
Spec.	90.1%	89.2%	88.8%	94.4%	87.4%	94.4%
***p*-val**	1.19×10^{-8}	6.5×10^{-10}	2.5×10^{-10}	1.4×10^{-10}	4.5×10^{-10}	7.1×10^{-10}

Table 1. ModuleFinder was run on a human skeletal muscle dataset using both exact word matches and PWMs. The searches were done using no alignments, mouse/rat alignments, and mouse/chicken alignments. For each search, sensitivity ("Sens."), specificity ("Spec."), and a t-test on the means ("*p*-val") were computed, as compared to matched random regions.

The results of these evaluations are shown in **Table 1**. Here, we have reported those values for sensitivity and specificity which maximally discriminate between the positive and negative control sets (i.e., using the threshold score such that the difference between the sensitivity and specificity is minimized). Since there was great variability in score among the positive control regions (see **Figure 1**; i.e., the top positive control region received a score of -11.23 and the worst positive control region scored only -1.22 (positive controls: mean = -4.69, SD = 2.24; negative controls: mean = -0.42, SD = 0.81)), we also performed a t-test on the positive and negative control region means, in order to measure the effectiveness of ModuleFinder on regions falling far from the threshold score.

On this dataset, ModuleFinder achieved a maximum sensitivity of 96.3% and specificity of 94.4% on the H/M/R PWM searches. Moreover, the PWM approach consistently gave better discrimination than exact word matches. Much of this improved discrimination, however, is an artifact of the jack-knife procedure, which has a stronger effect on exact match searches. Here, using the complete set of BSs (i.e., without the jack-knife), exact word matching achieves 100% sensitivity and 95.1% specificity (we removed degenerate flanking sequences for all searches with exact words). In addition, these results indicate that the H/M/R searches reliably outperformed the H/M/C searches. There are two possible explanations for this: 1) the chicken genome is not yet complete, and the appropriate alignment regions may not have been sequenced yet; 2) the underlying mechanisms of transcriptional regulation are not actually conserved in an organism as distant as chicken. Neither of these hypotheses can be ruled out until the completion of the chicken genome.

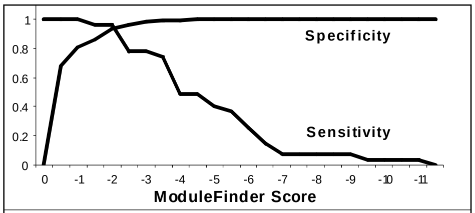

Figure 1: Sensitivity and specificity of ModuleFinder on skeletal muscle test regions, versus randomly selected control regions.

Since ModuleFinder was specifically developed to integrate homotypic clustering, heterotypic clustering, and conservation, we wanted to determine which of these features were most contributory to discriminatory power. In order to assess this, we ran ModuleFinder on the positive and negative control regions using no alignments, one alignment (each of mouse, rat and chicken), and two alignments (H/M/R and H/M/C). These searches were repeated with each TF individually, as well as with all 5 TFs together. In **Figure 2**, we show the negative logarithm of the *p*-values obtained from t-tests on the positive versus negative control regions for each of these searches. Here the mouse and rat alignments improved discriminatory power, but little was gained by using both genomes, because of their evolutionary proximity. Somewhat surprisingly, using chicken actually reduced discrimination relative to human alone. This was unexpected, as it implies that our negative controls are more likely to be conserved than these 27 regions. However, this effect could be an artifact of the small size of the positive controls and gaps in the chicken genome (only 13/27 positive controls had any alignable chicken sequence).

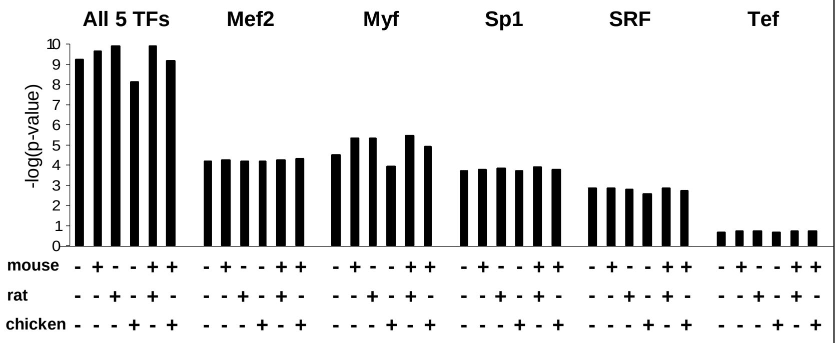

Figure 2: Negative log of p-values obtained from t-test on means between positive and negative controls. ModuleFinder was run with various combinations of mouse, rat, and chicken alignments (indicated by +/-), using all 5 TFs together, and each TF alone.

Finally, at least four other algorithms have used overlapping subsets of this dataset as positive controls[11-13,28], achieving sensitivities between 59% and 66%, and specificities between 95.3% and 97.1% (see **Table 2**). Thus, ModuleFinder appears to have comparable specificity but greater sensitivity. However, note the following caveats for this comparison. First, because ModuleFinder uses evolutionary conservation as a central component and because few vertebrate genomes have been sequenced, we limited our searches to the subset of the original compilation for which human/rodent alignments were available[18]. The other algorithms tested on this dataset did not consider conservation, and thus used the original, larger compilation that included CRMs obtained from diverse organisms including chicken, hamster, rabbit, pig and cow[11]. Frith *et al.*[12,13] trimmed this larger set[11] to a subset of 27 regions, but their subset overlapped with ours by only 15 genes. Second, each group used a different set of negative controls. The original paper by Wasserman *et al.*[11] used a set of negative control regions similar to our set; it was composed of 200 bp regions selected from the Eukaryotic Promoter Database. Comet and Cister were each tested on 300 bp regions that were selected to overlap well-characterized transcriptional Starts[12,13]. Finally, MSCAN[28] measured specificity by looking at the "hit rate" in contiguous stretches of the *Fugu* genome.

Algorithm	**Sensitivity**	**Specificity**
Logistic Regression[11]	60%	96%
Cister[12]	59%	97.1%
COMET[13]	59%	95.3%
MSCAN[28]	66%	NA
ModuleFinder	96%	94%

Table 2. Relative performance of ModuleFinder: Sensitivities and specificities, as reported by groups using overlapping subsets of the skeletal muscle dataset. Logistic regression, Cister, Comet and ModuleFinder specificities refer to 200-300bp portions of the human genome; the MSCAN specificity was ascertained using large stretches of *Fugu* sequence.

3.2 Other validations of ModuleFinder

In addition to the mammalian skeletal muscle set, we have also tested ModuleFinder on a *D. melanogaster* dataset that comprises 20 transcriptional enhancers from 9 genes known to be co-regulated during anterior-posterior segmentation of fly embryos[7]. Using the *D. melanogaster/D. pseudoobscura* alignments and a protocol similar to that described in Section 2.1, ModuleFinder was able to discriminate this collection of CRMs from randomly chosen noncoding regions with 95% sensitivity and 95% specificity (Philippakis *et al.*, manuscript in preparation). In addition to these *in silico* confirmations, we have also successfully applied ModuleFinder to predict CRMs in three biological systems: (1) development of the fly pericardium (Michaud *et al.*, manuscript in preparation), (2) development of fly muscle founder cells (Philippakis *et al.*, manuscript in preparation), and (3) mammalian myogenesis (Warner *et al.*, manuscript in preparation). For mammalian myogenesis, we applied the same 5

TFs and their BSs as described above; indeed much of the work presented here was done for the explicit purpose of selecting optimized BSs and sequence alignments before attempting to predict novel mammalian CRMs.

4. Discussion and Future Directions

We have presented a statistically rigorous approach for scoring windows of genomic sequence according to their likelihood of containing BSs for a collection of input TFs. The approach systematically integrates homotypic clustering, heterotypic clustering and evolutionary conservation across multiple genomes into a single, objective scoring scheme that does not require training. Additionally, our algorithm, implemented as a *C* program called "ModuleFinder," is publicly available for download, along with pre-processed genomes and alignments for yeast, worm, fly, mouse, rat, and human, at our lab website (http://the_brain.bwh.harvard.edu). The current version of ModuleFinder considers up to two alignment genomes as input, and we are currently expanding it to accept arbitrarily many genomes.

We have tested ModuleFinder on a set of human skeletal muscle CRMs using a variety of genome alignments and TFBSs, and have achieved a maximum sensitivity and specificity of 96% and 94%. On this dataset, improved sensitivity and specificity were achieved by using mouse and rat alignments in the searches, whereas chicken alignments actually decreased sensitivity and specificity. Furthermore, PWMs resulted in improved sensitivity and specificity over exact TFBS matches. Preliminary results indicate that ModuleFinder can successfully predict novel CRMs in human myoblasts (Warner *et al.*, manuscript in preparation). In addition, on a *D. melanogaster* segmentation gene dataset with *D. pseudoobscura* as the alignment genome, ModuleFinder achieved sensitivity and specificity of 95% and 95%. We have also predicted and experimentally validated several novel CRMs in the developing fly mesoderm (Philippakis *et al.*, manuscript in preparation). We expect that in the future we and others will use ModuleFinder to further refine transcriptional regulatory models for CRMs in particular biological systems and thus discover how the associated TFBSs are organized to confer specific gene expression patterns.

5. Acknowledgments

The authors thank Bertrand Huber and Jason Warner for helpful discussion, and Mike Berger, Mark Umbarger, and Roman Yelensky for critical reading of the manuscript. This work was funded in part by a PhRMA Foundation Informatics Research Starter Grant (M.L.B.), a William F. Milton Fund Award (M.L.B.), and NIH/NHGRI R01 HG02966-01 (M.L.B.). A.A.P. was supported in part by a National Defense Science and Engineering Graduate Fellowship from the Department of Defense, and an Athinoula Martinos Fellowship from HST.

References

1. Bulyk, M.L. *Genome Biol.* **5**, 201 (2003).
2. Boffelli, D. et al. *Science* **299**, 1391-4 (2003).
3. Cliften, P. et al. *Science* **301**, 71-6 (2003).
4. Kellis, M., Patterson, N., Endrizzi, M., Birren, B. & Lander, E.S. *Nature* **423**, 241-54 (2003).
5. McGuire, A.M., Hughes, J.D. & Church, G.M. *Genome Res.* **10**, 744-57 (2000).
6. Thomas, J.W. et al. *Nature* **424**, 788-93 (2003).
7. Berman, B.P. et al. *Proc. Natl. Acad. Sci. USA* **99**, 757-62 (2002).
8. Halfon, M.S., Grad, Y., Church, G.M. & Michelson, A.M. *Genome Res.* **12**, 1019-28 (2002).
9. Markstein, M., Markstein, P., Markstein, V. & Levine, M.S. *Proc. Natl. Acad. Sci. USA* **99**, 763-8 (2002).
10. Krivan, W. & Wasserman, W.W. *Genome Res.* **11**, 1559-66 (2001).
11. Wasserman, W.W. & Fickett, J.W. *J. Mol. Biol.* **278**, 167-81 (1998).
12. Frith, M.C., Hansen, U. & Weng, Z. *Bioinformatics* **17**, 878-89 (2001).
13. Frith, M.C., Spouge, J.L., Hansen, U. & Weng, Z. *Nucleic Acids Res.* **30**, 3214-24 (2002).
14. Rajewsky, N., Vergassola, M., Gaul, U. & Siggia, E.D. *BMC Bioinformatics* **3**, 30 (2002).
15. Rebeiz, M., Reeves, N.L. & Posakony, J.W. *Proc. Natl. Acad. Sci. USA* **99**, 9888-93 (2002).
16. Sinha, S., van Nimwegen, E. & Siggia, E.D. *Bioinformatics* **19 Suppl 1**, i292-301 (2003).
17. Altschul, S.F., Gish, W., Miller, W., Myers, E.W. & Lipman, D.J. *J. Mol. Biol.* **215**, 403-10 (1990).
18. Wasserman, W.W., Palumbo, M., Thompson, W., Fickett, J.W. & Lawrence, C.E. *Nat. Genet.* **26**, 225-8 (2000).
19. Blanchette, M., Schwikowski, B. & Tompa, M. *J. Comput. Biol.* **9**, 211-23 (2002).
20. Moses, A.M., Chiang, D.Y. & Eisen, M.B. *Pac. Symp. Biocomput.*, 324-35 (2004).
21. Prakash, A., Blanchette, M., Sinha, S. & Tompa, M. *Pac. Symp. Biocomput.*, 348-59 (2004).
22. Reinert, G., Schbath, S. & Waterman, M.S. *J. Comput. Biol.* **7**, 1-46 (2000).
23. Irving, R.W. & Love, L. *Technical Report no. TR-2001082 of the Computing Science Department of Glasgow University* (2001).
24. http://www.genome.ucsc.edu.
25. Stormo, G.D. *Bioinformatics* **16**, 16-23 (2000).
26. Andres, V., Cervera, M. & Mahdavi, V. *J. Biol. Chem.* **270**, 23246-9 (1995).
27. http://www.cognia.com.
28. Johansson, O., Alkema, W., Wasserman, W.W. & Lagergren, J. *Bioinformatics* **19 Suppl 1**, i169-76 (2003).