Machine Learning for Regulatory Analysis and Transcription Factor Target Prediction in Yeast

Dustin T. Holloway[1], Mark Kon[2,3], Charles DeLisi[3§]

[1] Molecular Biology Cell Biology and Biochemistry, Boston University, Boston, MA 02215, U.S.A.
[2] Department of Mathematics and Statistics, Boston University, Boston, MA 02215, U.S.A.
[3] Bioinformatics and Systems Biology, Boston University, Boston, MA 02215, U.S.A.

[§]Corresponding author

Email addresses:
DTH: dth128@bu.edu
MK: mkon@bu.edu
CD: delisi@bu.edu

Abstract

High throughput technologies, including array-based chromatin immunoprecipitation, have rapidly increased in our knowledge of transcriptional maps—the identity and location of regulatory binding sites within genomes. Still, the full identification of sites, even in lower eukaryotes, remains largely incomplete. In this paper we develop a supervised learning approach to site identification using support vector machines (SVMs) to combine 26 different data types. A comparison with the standard approach to site identification using position specific scoring matrices (PSSMs) for a set of 104 *Saccharomyces cerevisiae* regulators indicates that our SVM-based target classification is more sensitive (73% vs 20%) and has double the precision (positive predictive values of 90% and 45% respectively).

We have applied our SVM classifier for each transcriptional regulator to all promoters in the yeast genome to obtain thousands of new targets, which are currently being analyzed and refined to limit the risk of classifier over-fitting. For the purpose of illustration we discuss several results, including biochemical pathway predictions for Gcn4 and Rap1. For both transcription factors SVM predictions match well with the known biology of control mechanisms, and possible new roles for these factors are suggested, such as a function for Rap1 in regulating fermentative growth.

We also examine the promoter melting temperature curves for the targets of YJR060W, and show that targets of this TF have potentially unique physical properties which distinguish them from other genes. The SVM output automatically provides the means to rank dataset features to identify important biological elements. We use this property to rank classifying *k*-mers, thereby reconstructing known binding sites for several TFs, and to rank expression experiments, determining the conditions under which Fhl1 and the factor responsible for expression of ribosomal protein genes is active. After identifying these conditions, we can see that targets of Fhl1 are differentially expressed in them, as compared to expressions of average and negative set genes. SVM-based classifiers provide a robust framework for analysis of regulatory networks. Processing of classifier outputs can provide high quality predictions and biological insight into functions of particular transcription factors. Future work on this method will focus on increasing the accuracy and quality of predictions using feature reduction and clustering strategies. Since predictions have been made on only 104 TFs in yeast, new classifiers will be built for the remaining 100 factors which have available binding data.

Background

Understanding transcriptional regulation is one of the key challenges of the post-genomic era. Transcription factors control the expression of their target genes by binding specific sequences of bases, typically 10-15nt in length, in a region upstream of transcription initiation. Sequences bound by a TF are not identical to each other and only represent a preferred pattern of nucleotides within a binding motif. The complete regulation of a gene will often depend on the cooperative or antagonistic effects of several transcription factors with potentially overlapping binding sites. Thus, the regulatory code for a gene is composed of a pattern of degenerate motifs concealed within the promoter.

Many methods for predicting additional target sites for a TF have been proposed. Given a set of genes known to be regulated by a certain factor and a set known not to be coregulated, supervised learning tools such as support vector

machines (SVM) can be used to categorize new genes. Unsupervised methods begin with less well-defined information, for example a set of co-expressed genes from a microarray study which are thought to contain some set of common but unknown patterns. New patterns can then be discovered by statistical overrepresentation or by local search algorithms such as Gibbs sampling. Several unsupervised techniques for predicting binding sites have been reported [1-8], and an excellent review of current motif-discovery methods is available [9]. Founding work in TF binding site representation involved the use of position specific scoring matrices (PSSMs) [10-13], which contain the frequency of nucleotide bases at each position in a possible binding site, or motif. New predictions are sites which match the PSSM based on a score threshold[10]. Later, clusters of predicted binding sites have been shown to be predictive of whether a gene is a target of a regulator or not[14-17].

The approach reported here is a supervised pattern classification scheme designed to integrate a large number of heterogeneous data sources in order to more accurately predict the association of a transcription factor and its target. In particular, we explore the use of support vector machines, which are able to incorporate high-dimensional data sets (many features). SVM classifiers have previously been used for the prediction of protein homology[18], secondary structure[19], and sub-cellular localization[20]. As sequence classifiers they have also been useful in predicting translation start sites[21], mRNA splice sites, and signal peptide cleavage sites[22]. More broadly they show good performance in the identification of normal and cancerous tissue samples[23] as well as prediction of gene function[24].

Few groups have published work on supervised classification schemes for predicting new transcription factor targets. We briefly reviewed some of these previously (submitted[25]). One method includes linear discriminant analysis (LDA) to select from a set of potentially co-regulated genes those that are most likely to share common transcription factors[26]. Another approach uses Bayesian networks to learn the combinatorial relationships of TFs and targets that underlie specific gene expression experiments[27]. Finally, in an approach similar to ours, SVMs have been applied to microarray data in order to predict TF-target associations[28].

Although some of these techniques work well, they either do not effectively incorporate the large amount of regulatory data available in ChIP-chip interactions or they base their classification on only one or two types of genomic data. Our approach easily combines 26 large genomic datasets, adaptively weighting each data source based on its ability to correctly classify a training set. The combination of heterogeneous data reduces false positive predictions while maintaining high accuracy. Genomic data combination using SVMs has been demonstrated before. Protein sequence similarity, protein-protein interactions, protein hydrophobicity, and gene expression data were successfully combined to predict the functional group of a set of proteins, and the combination of data was shown to significantly outperform individual methods[29].

SVMs: Background

We consider 26 different datasets sequentially, train a classifier on each, and then construct a composite classifier which is a weighted combination of the 26. For each training set, we develop an allocation rule for every TF. Let N be the size of the training set for a particular TF (the collection of positive and negative examples, i.e., genes which do and do not bind it). Each gene has a set of attributes forming a vector that contributes to the distinction between positive and negative sets. As an example, an attribute vector for a gene could be an ordered list consisting of the number of

times each possible 4-mer occurs in the upstream region. The collection of such vectors is the *feature space F*. Each gene would then be characterized by a 256 component *feature vector*. The SVM generates a hyperplane of $D = 255$ dimensions in the feature space separating positives from negatives (d will henceforth be an index over the features of the dataset). We write a vector in F as $\mathbf{x}_i = (x_{i1}, x_{i2}, x_{i3}...x_{id})$, the components x_{id} representing, for the example above, the count of the d^{th} k-mer in the i^{th} gene. Then the equation for a hyperplane has the form

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b = 0 \qquad (2)$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_d)$ and $\mathbf{w} \equiv (w_1, w_2, ..., w_d)$. For $D = 2$, this is a straight line in variables $\mathbf{x} = (x_1, x_2)$ with slope $-w_2/w_1$ and intercept $-b/w_1$.

Geometrically $\mathbf{w}$ is a vector perpendicular to the hyperplane H, the magnitude $|w_d|$ of its d^{th} component weighting the corresponding dimension. The function $f(\mathbf{x})$ is assumed normalized (through scaling of $\mathbf{w}$) so that the closest (positive, negative) pair $\mathbf{x}_i^+$ and $\mathbf{x}_i^-$ have values $f(\mathbf{x}^+) = 1$ and $f(\mathbf{x}^-) = -1$ respectively. Then the SVM problem is to find $\mathbf{w}$ and b such that the attribute vectors of all genes in the positive set are above the hyperplane H_1 defined by

$$\mathbf{w} \cdot \mathbf{x} + b = +1$$

and all in the negative set are below hyperplane H_2 defined by

$$\mathbf{w} \cdot \mathbf{x} + b = -1$$

and that the *margin* (distance between H_1 and H_2) is maximal. Thus the goal is to find a separator that maximizes the margin, or distance between the positive and negative classes. This construction is essentially a choice of scaling for $\mathbf{w}$, b, in particular requiring that the length $|\mathbf{w}|$ be minimal, since this maximizes the margin under the above normalization. Maximizing the margin is a *convex optimization* problem which is generally solved using standard Lagrangian methods[30]. Typically, as in our case, perfect separation cannot be achieved. When error-free decisions are not possible the method can be readily generalized to allow any specified amount of misclassification, with a suitable penalty function.

An important aspect of the solution is that the data enter only in the form of a *kernel matrix K*, whose entries K_{ij} are dot products of all pairs $\mathbf{x}_i$, $\mathbf{x}_j$ of feature vectors. In the case that all components of the feature vector are truly independent, the Lagrangian is a linear function of the elements of the kernel, and the linear dot product is used with $K_{ij} = \mathbf{x}_i \cdot \mathbf{x}_j$. When the elements are correlated, the Lagrangian is written as a nonlinear function of the inner products of the attribute vectors (see below). In particular, the nonlinear dot products are defined for data points by $K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$, where the given positive definite function $K(\mathbf{x}, \mathbf{y})$ is known as the *kernel function*. Such nonlinear products are equivalent to assuming that an unspecified higher dimensional feature space F_1 exists into which F is mapped and in which the separating hyperplane is linear. This yields a Lagrangian with matrix entries given by this alternative dot product. The implicit choice of F_1 is made by changing the type of inner product used (see Table 1). For a more detailed development of SVMs, see our Supplementary information or the excellent reference texts[30, 31]. For a detailed 2-dimensional example see [25].

Kernel	Parameters	Description
linear	none	$K(\mathbf{x},\mathbf{y})=\mathbf{x}\cdot\mathbf{y}$
polynomial	poly degree d	$K(\mathbf{x},\mathbf{y})=(\mathbf{x}\cdot\mathbf{y}+1)^d$

Gaussian radial basis function (RBF)	σ	$K(\mathbf{x},\mathbf{y}) = \exp\left(\frac{-\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}\right)$
Gaussian	σ	$K(\mathbf{x},\mathbf{y}) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$

Table 1 - Four common kernels tested

These are the four common kernel functions, the parameters which must be set by the user, and their mathematical description.

Post-processing can be an essential task in pattern classification problems, particularly if one wishes to extract the highest quality predictions from a classifier. A naïve way to extract the most significant (positive) prediction from an SVM classifier is to select those data points which are most distant from the separator (distance given by $\mathbf{w}\cdot\mathbf{x}_i + b$ for data point i). The interpretation is that those distant points are most unlike the negative set and contain the strongest positive character. A more informative method is to rank data by $P(y_i = 1 | \mathbf{w} \cdot \mathbf{x}_i + b)$; i.e. by the posterior probability of a positive classification, given the distance of example $\mathbf{x}_i$ from the hyperplane. Platt observed that these posterior probabilities could be well approximated by fitting the SVM output to the form of a sigmoid function[32], and developed a procedure to generate the best-fit sigmoid to an SVM output for any dataset. The result is the posterior probability $P(y_i = 1 | \mathbf{w} \cdot \mathbf{x}_i + b)$ for each data point in the training set (see [32] for further details). This probability places a confidence level on any new prediction made in the yeast genome and, most importantly, results in an ability to identify high-confidence predictions for future experiments.

Results and Discussion

After data pre-processing, the analysis begins with the independent evaluation of each dataset on each TF. Several kernel functions are tested and any necessary parameters are optimized before a final classifier is constructed (see Methods). A schematic of our procedure is given in Figure 1. Once parameter optimized classifiers are constructed for each TF-dataset pair, all of the datasets, represented by the optimized kernel matrices, are combined using a weighting scheme based on their F1 scores. The hypergeometric test is used to filter out datasets which do not perform better than random (accept p-value ≤ 0.05) for a particular TF. Accuracy estimates for the combined classifier are made using a final leave-one-out cross validation.

Figure 1

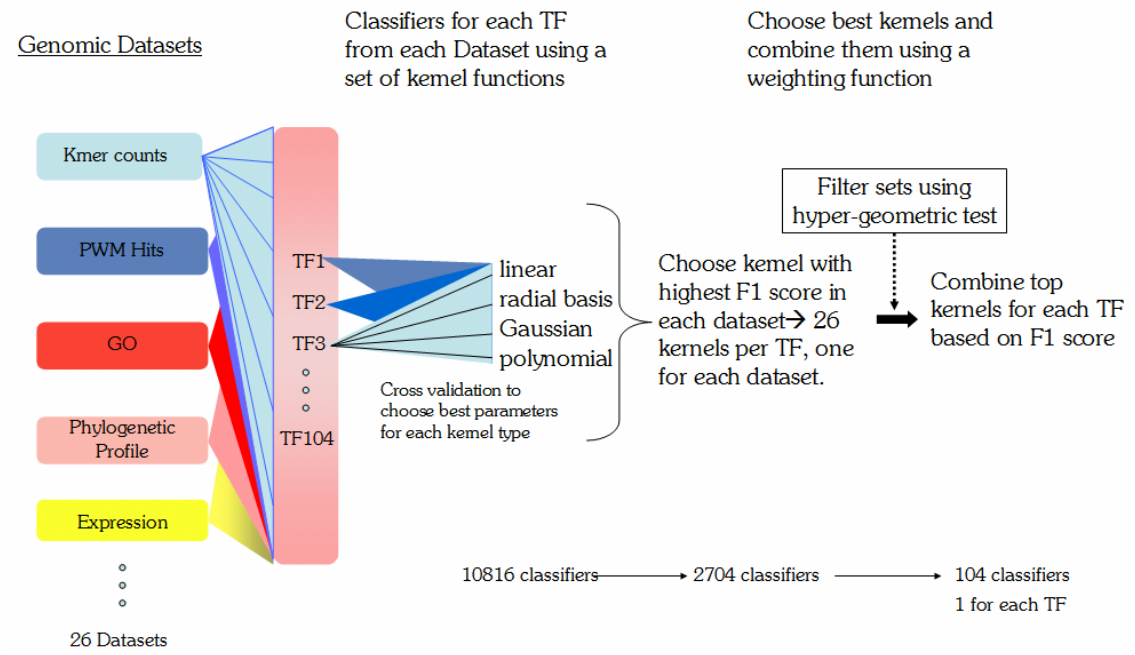

Figure 1 Flow diagram: synthesizing a single classifier for each TF from several data sets.

A classifier is constructed for each individual TF for each genomic dataset, using every one of 4 possible kernel functions (26 datasets × 104 TFs × 4 kernel functions = 10816 kernels from which SVM classifiers are built). For each of these classifiers optimal parameters are chosen by cross-validation. For each dataset and each TF, the best performing of the four kernel functions is selected, reducing the number of classifiers to 2704 (26 datasets x 104TFs). Finally, the datasets are combined based on F_1 score of their best performing kernel so that there is only one classifier per TF.

Three simple weighting schemes have been tried (see Methods), and the primary weight for a method is the ratio of its F_1 score with that of the best performing method. The first scheme simply multiplies all kernel matrices by their scaled F_1 scores and sums them. The second scheme squares the weights before multiplying. This has the effect of decreasing weights of poorly performing methods. Our third scheme uses the squared tangent of the primary weight. This will more severely penalize poor performers while boosting the weights of the best methods (e.g., instead of weight 1, the best method will have a weight of 2.43).

We have been able to accurately classify the known targets of many transcription factors in *S. cerevisiae*.

Figure 2

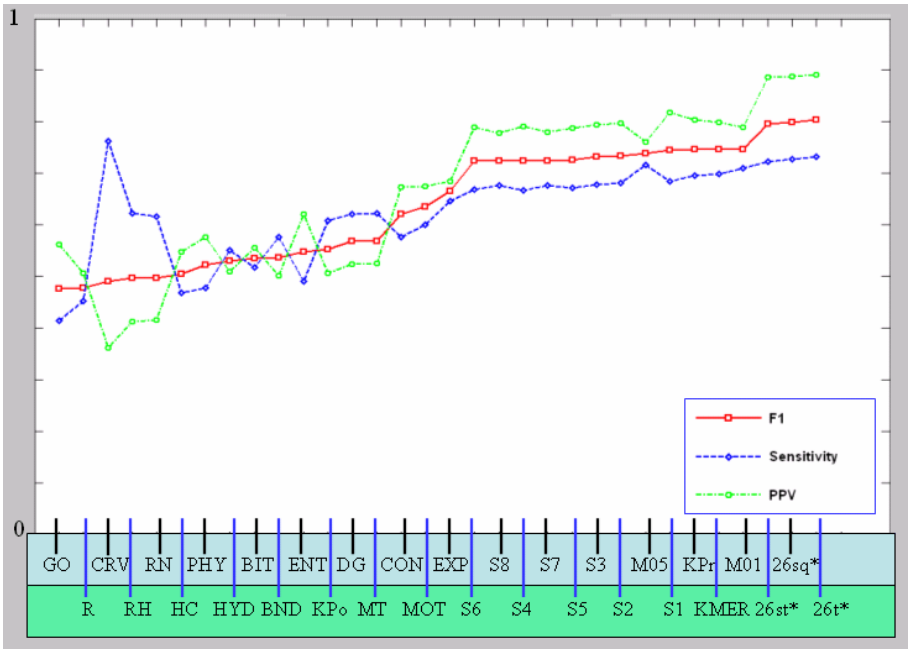

Figure 2 SVM Performance

Performance of each dataset and combined datasets ordered by increasing F_1 score. Cumulative results for all transcription factors were used to plot the sensitivity, positive-predictive-value, and the F_1 statistic for each dataset and data combination. Dataset abbreviations are given in Table 3. The combined classifiers, labeled 26st (linear weighting), 26sq (square weighting), and 26t (tangent square weighting) on the far right, perform better than any dataset alone, with the squared tangent weighting giving the best result overall. Three random datasets also appear in the table, R (randomized *k*-mer counts), RH (randomized 10% selection of each dataset), and RN (normally distributed random numbers).

Figure 2 shows the performance of classifiers generated on each individual dataset. The combination of datasets performs better than any individual type of data, but the best single method achieves a sensitivity of 71% and a positive predictive value of 0.82. The combined datasets are labeled STD for weighting based on simply the scaled F_1 measure, SQU for weighting based on squared, scaled F_1 measure, and TAN for weighting based on the tangent squared F_1 measure, as described in Methods. Other abbreviations can be found in Table 2. Almost all methods perform much better than random. The exceptions are GO term annotation and phylogenetic profiles. For phylogenetic profiles this is not unexpected, since only 30% of the yeast genome has an established ortholog in the COG database. This absence of data means that many positive examples can no longer contribute to classification, leading to poor performance for most TFs. The situation is similar for GO term annotation, where many genes are poorly annotated or have no known function.

Table 2

	Abbreviation	Description	

1	MOT	Motif hits in S.cerevisiae	
2	CON	Motif hits conservation 18 organisms	
3	PHY	Phylogenetic profile	
4	EXP	Expression correlation	
5	GO	GO term profile	
6	KMER	K-mers – 4,5,6-mers	
7	S1	Split 6-mer 1 gap kkk_kkk	
8	S2	Split 6-mer 2 gaps kkk__kkk	
9	S3	Split 6-mer 3 gaps kkk___kkk	
10	S4	Split 6-mer 4 gaps kkk____kkk	
11	S5	Split 6-mer 5 gaps kkk_____kkk	
12	S6	Split 6-mer 6 gaps kkk______kkk	
13	S7	Split 6-mer 7 gaps kkk_______kkk	
14	S8	Split 6-mer 8 gaps kkk________kkk	
15	M01	6-mer with 1 mismatch (count 0.1)	
16	M05	6-mer with 1 mismatch (count 0.5)	
17	ENT	Condition specific TF-target correlation	
18	BIT	Nucleotide sparse binary encoding	
19	CRV	Promoter Curvature prediction	
20	HC	Homolog Conservation	
21	HYD	Hydroxyl Cleavage	
22	KPo	Kmer median positions from start	
23	KPr	Kmer Probabilities (-log pval)	
24	MT	Promoter Melting Temperature-20bp window	
25	DG	Promoter Melting Delta G profile-20bp win	
26	BND	Promoter bend prediction	

Table 2 - Abbreviations of datasets used to generate classifers

Abbreviations for each dataset and a short description are given.

The performance statistics mentioned in Figure 2 are a summary of those for all 104 combined classifiers. Since there are 9104 known positives for all regulators, a sensitivity of 71% indicates that, considering all 104 classifiers, we recover 71% of the known data. This means that classifiers for some TFs have much higher sensitivities or PPVs while other classifiers perform no better than random.

The most powerful individual classification uses *k*-mer counts allowing 1-missmatch per *k*-mer. However, the combination of all of the methods shows increased sensitivity and precision over all individual methods. The squared-tangent weighting function performs the best overall, reaching a sensitivity of 73% and a positive predictive value of 0.89. Looking only at the top 20 TFs, we see a sensitivity and PPV of 88.2% and 0.9 respectively. Our results show that combining datasets increases sensitivity only incrementally over classifiers built on simple *k*-mer counts alone, and that it produces a small improvement in positive predictive value. Thus, combining methods results in the modest reduction of false positive classifications.

The use of the hypergeometric distribution to test the significance of a dataset for each TF allows us to assess how useful a particular data type is for target identification. Figure 3 plots the percentage of TFs for which each dataset has been found to be significant at $p \leq 0.05$. Overall, sequence based methods (*k*-mer counts,

mismatch and gapped *k*-mer counts, and *k*-mer likelihoods) show the best overall coverage, being significant for almost all transcription factors. Structural descriptions of the promoter region differ greatly in their usefulness, varying from DNA curve prediction, useful for ~15% of TFs, to melting temperature profiles and free energy values, significant for over 60% of TFs tested.

Figure 3

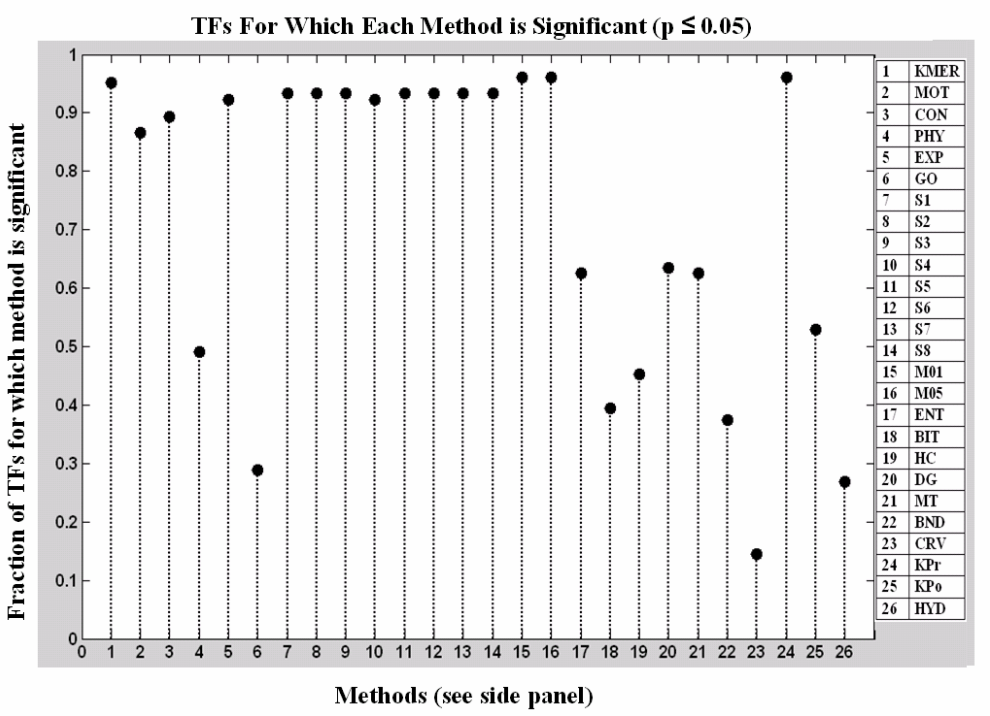

Figure 3 Percentage of TFs for which each dataset is significant ($p \leq 0.05$).

Percentage of TFs is on the left axis and datasets are numbered along the bottom with a key given to the right of the diagram (see Table 3 for descriptions of method abbreviations).

In work with genomic datasets with large numbers of features (e.g., *k*-mer counts, expression measurements) there is always an inherent risk of over-fitting when the number of positives and negatives are relatively small. To give a more practical portrayal of our method and prevent an overly optimistic view of the results, it is illuminating to compare our results with those obtained from classifiers obtained running the same training on random data. Thus three random datasets have been constructed as controls and their results displayed in Figure 2. The first, abbreviated R, is simply randomly shuffled *k*-mer count data. The second (RH) is created by shuffling a composite dataset composed of a random 10% selection of each individual dataset. The third (RN) is a normally distributed random set of numbers with mean zero and standard deviation one.

Although performance is much better than random it is doubtful from these results that predictions obtained by applying our classifiers to the entire genome would yield

truly reliable targets without further processing. A simple classification of all potential targets with our 104 classifiers returns, on average, ~800 new targets for each TF. The conditional probabilities given as output from Platt's method[32] allows the selection of possible targets[33] at a desired probability threshold. For instance, one can easily select predictions for which the probability of being a positive is greater than 0.99. In some of the examples below, the top targets were selected in this fashion and compared to the full set of known positive genes.

Another method to reduce the risk of over-fitting, which we reserve for our future work, is application of sophisticated dimension reduction techniques to discover significant features in different datasets based on classifier performance. Feature selection and clustering will allow the most relevant features from different datasets to be retained while large portions of redundant and irrelevant information are discarded. In some cases this has been shown to increase classifier accuracy. In other cases, the reduction in the complexity of the problem is worthwhile since other learning algorithms ,like *k*-nearest-neighbors or Bayes networks, which are difficult to train on large feature sets, could be compared efficiently on the smaller set of features.

The dynamics of the individual classifiers can also be examined based on distributions of sensitivity and F_1 score as compared to the random classifier. Figure 4a and Figure 4c show the distribution of F_1 score and sensitivity respectively for normal random data. Figure 4b and Figure 4d show the same distributions but for actual data (26 method combination with tangent weights). The sensitivities and F_1 scores for actual data have distributions heavily shifted to the right as opposed to those for random data. Although the majority of classifiers are comparatively good, several TFs have poor performance, something which warrants further inspection. There are 4 classifiers for which the F_1 score and sensitivity are zero (YHL020C, YNL139C, YER068W, and YER161C). These factors have comparatively few known targets compared to others. On average these four TFs have 10 targets each (one of them has only 3 positives) in their training sets compared to an average of 88 targets for most regulators. This low number of positive examples is likely the cause of the poor performance. Figure 5 shows a plot of sensitivity versus TF sorted by increasing number of positives for all classifiers. The general trend shows that classifiers having more positives give better performance.

Figure 4

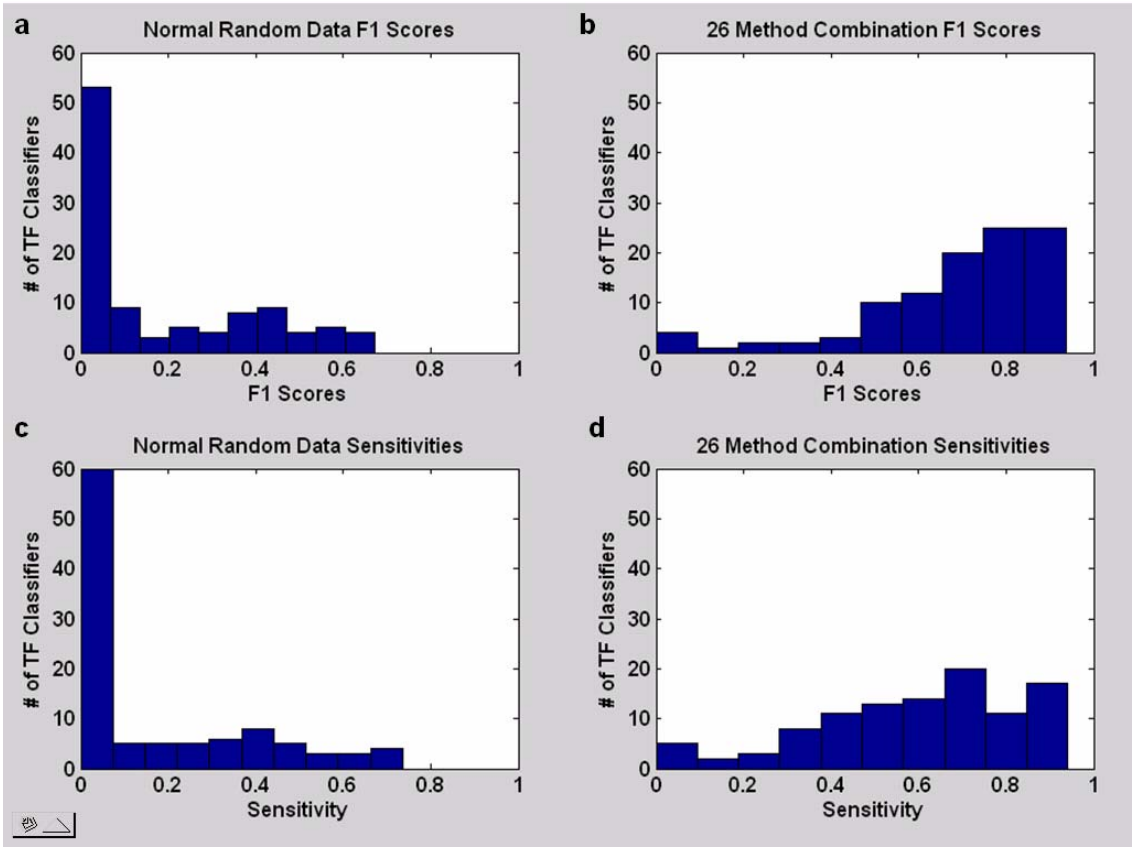

Figure 4 Random vs Combined Classifiers

6a is the distribution of F_1 scores for normal random classifiers, 6b the same distribution on classifiers made from 26 dataset combinations for all TFs. 6c is the sensitivity distribution for normal random classifiers and 6d the sensitivity distribution for the 26 dataset classifiers for all TFs.

Figure 5

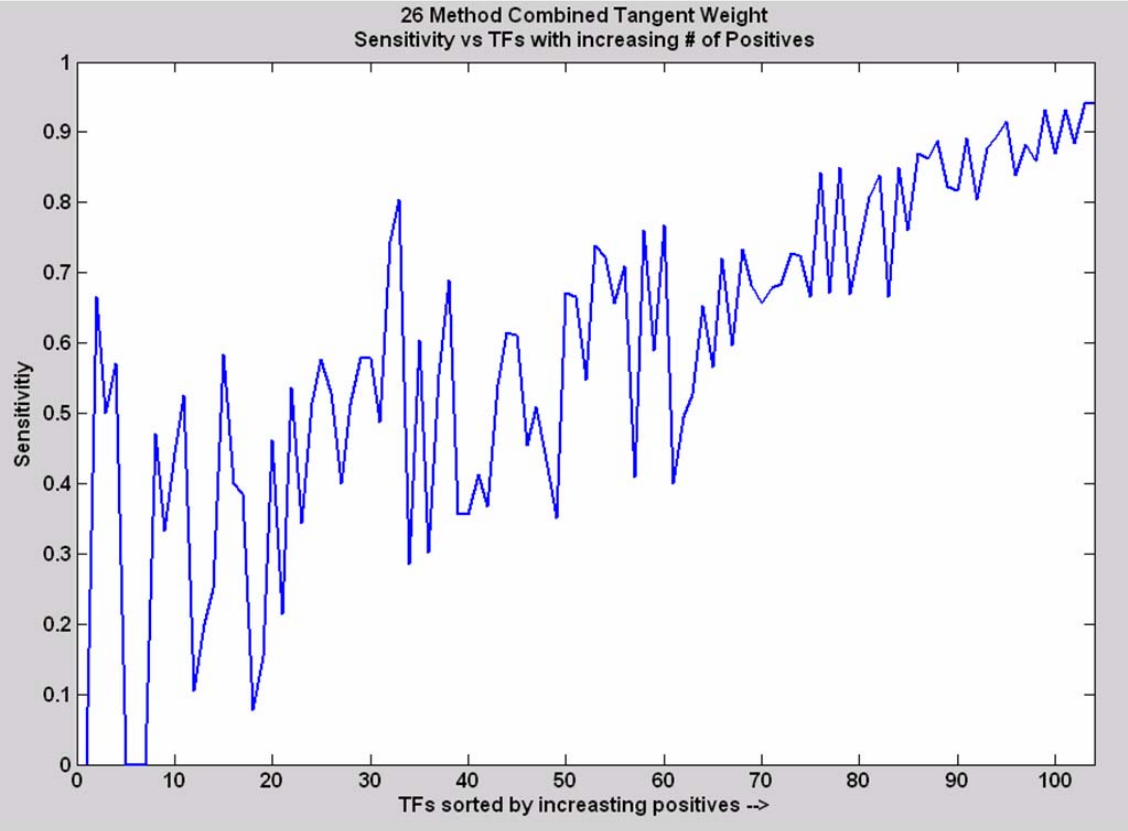

Figure 5 Sensitivity as a function of increasing positives

Classifiers for each TF were sorted according to increasing number of positives and the trend in their sensitivity is shown. Generally, classifiers with more positive examples perform better.

Biological insights-promoter melting

Beyond categorizing genomic datasets as useful or not for classification purposes, the significance of a particular dataset has potential biological implications for a TF. To see if this could be explored based on our results, the factor YJR060W was chosen for further examination, since the promoter melting temperature profile is significant for this TF at $p = 0.0037$. Figure 6 shows a plot of the average promoter melting temperature curve (calculated using a 20bp window and moving in steps of 1bp) over all genes in yeast (solid blue), the average curve for genes in this TF's negative set (dashed blue), the average in the TF's positive set (dashed red), and the average in the most significant 33 targets of the TF (solid red). The top 33 targets have Platt conditional probabilities $P($ positive | distance from separator$) \geq 0.99$ and are obtained from the predictions made using the combination of all datasets, thus representing the best predictions we can make for this TF. This is equivalent to choosing predictions significant with a p-value of 0.01. These most significant targets contain 18 new predictions which are not part of the original positive set.
Figure 6

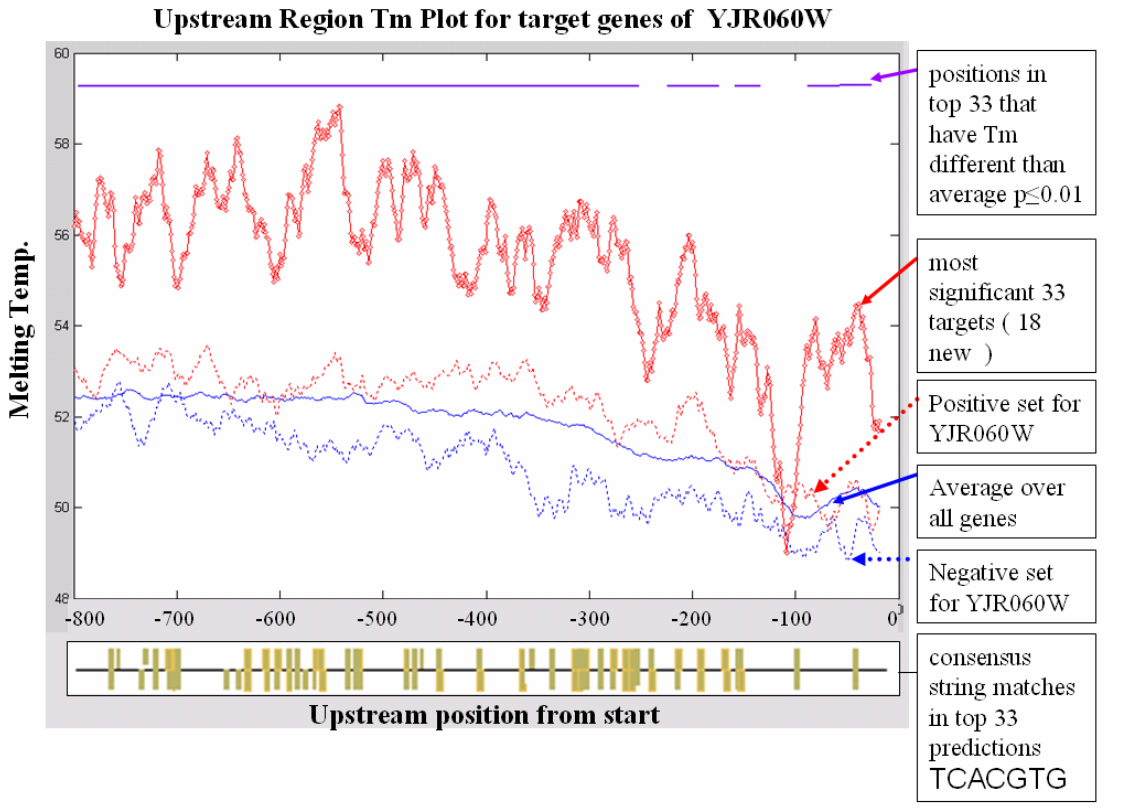

Figure 6 Melting Temperature Curves YJR060W

Using a 20bp window for DNA melting temperature calculation, the temperature plots are presented for the average over all 5571 yeast genes (solid blue), positive targets for YJR060W (dashed red), negatives for YJR060W (dashed blue), and high confidence targets (solid red—P(true|distance to separator) ≥ 0.99) determined using Platt's method for probability assignment to SVM output. Under the graph is an indicator displaying hits to the YJR060W consensus sequence in the top 33 targets. Consensus hits are distributed throughout the 800bp upstream space.

Clearly, the positive and negative groups for this TF contain average differences in promoter melting temperature. This difference is magnified when only the best targets are examined. The best 33 predictions have a very different melting signature from the negative set and the average yeast gene. A two-sample *t*-test was used to find the significance of this difference from the average curve. The purple overbar in Figure 4 shows the window positions where the best targets have an average value which is significant at $p \leq 0.01$. Almost all positions show a significant increase in melting temperature, with the exception of several positions proximal to the transcription start site. Considering that the transcription machinery must unwind the helix in this region, it is not unexpected that the melting temperature here would be smaller, as this would lower the activation energy needed to dissociate the strands.

As reviewed in Methods, there is ample support for the idea that melting temperature can influence transcription [34], and that torsional strain can affect the stability of the DNA duplex[35]. Experiments have also shown that sites susceptible to this kind of destabilization correlate well with regulatory regions[36]. In light of the high melting temperature of promoter targets of YJR060W, it is possible that

duplex destabilization plays a role in regulation by this TF. Indeed, experiments have shown that YJR060W functions largely in recruiting chromatin remodeling factors to proximal promoters[37]. The exact mechanism for this recruitment is not fully understood, but it is required for transcription at some promoters and complementary to additional binding factors at others[37]. In any case a possible hypothesis is that duplex stability is an important mechanism for regulation at these promoters and that YJR060W binding affects this stability either by conformational change induced by its binding or induced by the recruitment of chromatin remodeling factors. The conformational changes may alter the torsional strain on the DNA and thus affect the melting temperature prior to transcription.

Biological insights-binding site detection

Our results demonstrate that there is clearly a signal identifying ChIP-chip positives from other genes. Other groups have had less success confirming the validity of the ChIP-chip data, and this has led some to consider that as many as 50%[26] to 60%[38] of the targets produced by ChIP-chip are false positives in the assay. The fact that the high throughput results are chosen to be significant with $p \leq 0.001$ indicates that the transcription factors do in fact bind their targets. It is certainly possible that this binding does not always translate into changes in gene expression, that the changes are not large enough to be considered significant, or perhaps that the conditions under which binding would result in expression change were not tested. In any case, our classifier appears to pick up the information necessary to identify a target gene.

To find this signal we have looked at the results of various individual datasets and extracted the attributes which contribute most to a transcription factor's classifier. Support vector machines are often considered a "black box" method, since their results are not as readily interpretable as, for instance, the probability assessment of Bayesian classifiers. Nevertheless, the **w** vector described above can give an indication of which features in the data are important to the classification. Features whose components w_i are large correspond to dimensions in feature space where positives and negatives are more widely separated. Thus by examining a single dataset, e.g. *k*-mer counts, it is possible to determine the *k*-mer(s) most responsible for differences between positives and negatives. To this end, **w**-vectors from the *k*-mer count dataset have been calculated for each linear TF classifier and examined to determine which *k*-mers had the largest weights. We compare these *k*-mers to known binding sites for each factor. Results for the best 10 TFs can be seen in Table 3, where the highest ranked *k*-mers are manually assembled to show their correspondence with known binding motifs. In most cases the *k*-mers with the highest weights match closely the reported binding site for the TF, showing that the algorithm is choosing meaningful features for classification. For example, the DNA binding protein Cep1 is known the bind the consensus TCACGTG and regulate cell cycle and stress response genes. The highest weighted *k*-mer in the classifier is CACGT, and the top 4 *k*-mers all overlap precisely with the known site (CACGT,CGTG,TCACG,TCACGT) .

Table 3

Standard ID	Gene name	Known Motif (SGD)	Kmers labeled by rank
YKL112W	ABF1	RTCAYTNNNNACGW	1 CACT 2 ATCA 3 ACTAT 4 TCAC

			5 ATCAC ATCACT
YDR207C	UME6	TAGCCGCCSA	1 GCCG 2 TAAG 3 GCCGC 5 GCCGCC 6 AGCCGCC 7 TAGA TWAGCCGCC
YBR049C	REB1	CGGGTRR	1 TAAC 2 GGGTAA 3 GGTA 4 GGGTA GGGTAA
YLR182W	SWI6	CACGAAAA	No match 1,4,5,6,8 2 AACG 9 GGAA 3 ACGCG 7 CGCG AACGCG
YPR104C	FHL1	TGTAYGGRTG	No match 1-4,6 5 TGTA 7 GTACA 8 ATGTA ATGTA
YEL009C	GCN4	ARTGACTCW	1 ATGA 2 TGAC 3 TGACT 4 AACT 5 ACTC 7 ACTCA 8 GACT 9 ATGAC ATRACTCA
YJR060W	CEP1	TCACGTG	1 CACGT 2 CGTG 3 TCACG 4 TCACGT TCACGTG
YOL028C	YAP7	MTKASTMA	1 TAGA 2 GTAA 3 ATTA 4 ATATT 5 CGAA 6 CTTA AMTTASDAA
YER111C	SWI4	CACGAAAA CGC[G/C]AAA	1,2,3 match TATA box 4 GCGCA 5 CGCG 7 CGAA 10 GCGA CGCGMA
YNL216W	RAP1	CAYCCRTRCA RMACCCATACAYY	1 TAAAAT 2 ATTC 3 ATTAA 4 ACCCA 6 TACA 7 TAAAG 8 ACATC 9 ATTCC TAAARYCCATACATYMM

Table 3 - High ranking *k*-mer alignment and comparison to known binding site

Weight vectors for each TF classifier are used to rank all *k*-mers. Known TF motifs appear in the middle column and high ranking *k*-mers are assembled in the right column showing correspondence with the known motif. Standard nucleotide abbreviations are used. Some less common abbreviations are W = {A or T}, R = Purine, Y = Pyrimidine, S = {C or G}, K = {T or G}, M = {C or A}, D = not C.

Biological insights-Microarray Expression

The ability to identify the primary conditions under which a transcription factor exerts control would be a critical component of any focused study of gene regulation. As we have seen the **w** vector generated on a dataset indicates which of its components are most important for discriminating targets. In the case of gene expression classifiers, **w** elucidates which conditions are discriminatory. Intuitively, these are the conditions in which we would expect to see differential regulation of true targets. Given the predictions made using the combination of all methods, and the **w** obtained from the linear classifier built on expression data alone, we can see whether the predicted targets have differential regulation, and identify conditions where the TF is likely to act.

By the hypergeometric test, expression data is a significant predictor (p=6.12e-14) of targets for Fhl1, a forkhead-like TF known to be involved in rRNA processing and ribosomal protein gene expression. The **w** for this TF's classifier from expression data has been calculated and sorted to determine the conditions having the highest weight. Figure 7 shows a plot of expression values over the top 25 conditions for the average yeast gene (solid blue), the average for genes in Fhl1's negative set (dashed blue), the average in the positive set (dashed red), and the average in the most significant (P(true) ≥ 0.99) 48 targets of this TF (solid red).

Figure 7

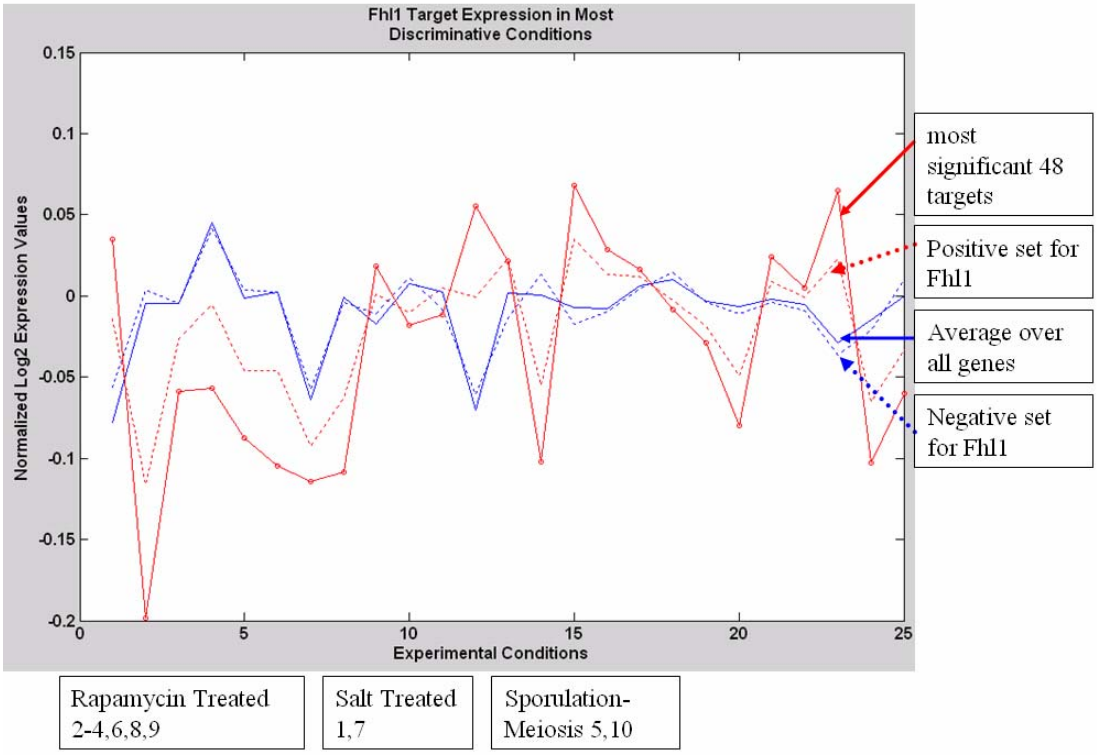

Figure 7 Expression plot of Fhl1 Targets over top 25 discriminative conditions.

Average expression is plotted over all 5571 yeast genes (solid blue), over the negative set for Fhl1 (dashed blue), the positive targets (dashed red), and the most significant targets (solid red), P(true | distance from classifier) ≥ 0.99. The best targets have expression significantly different than the average or negative genes. The chosen expression conditions, ranked by **w**-vector from the expression based classifer, are

shown under the graph with numbers indicating the position of the conditions in the graph. These conditions make sense since Fhl1 is regulated by the TOR signaling pathway, which is blocked by rapamycin. There is also some support in the literature for TOR having a role in meiosis and stress response.

For 23 of the 25 conditions the highly significant targets show expression which is different from both the average and the negative sets (*t*-test *p*-value $\leq$ 0.01). Most importantly, the best 10 ranked conditions contain 6 where yeast cells were treated with rapamycin and 2 involving meiosis/sporulation. This result is satisfying since rapamycin treatment specifically inhibits the TOR (**T**arget **O**f **R**apamycin) signaling pathway, which is known to activate ribosomal protein expression as well as regulate several other pathways in yeast. Inhibition of TOR directly prevents Fhl1 from binding at promoter sites, thereby down-regulating expression of ribosomal protein genes[39], explaining why Fhl1 targets show differential expression in these experiments. Furthermore, although Fhl1 has not been directly implicated in meiosis, TOR pathway kinases are required for meiosis[40], indirectly suggesting that Flh1 might be involved. This is a reasonable suggestion since Fhl1 has been shown to alter its activity in response to factors (mainly Sfp1 which is also under TOR control) controlling progression to Start in the yeast cell cycle. Thus the most highly ranked experiments seem to correlate well with the real biological roles of the TF, indicating that the SVM can correctly rank important experimental conditions. Our method can identify differential regulation as an important predictor of target genes (hypergeometric test) and use the SVM-based classifier to make testable hypothesis about which conditions show biological effects of transcription factor activity.

Biological insights-*PSSM comparison*

We have found that support vector classification performs better than a simple weight matrix scan, and the combination of 26 methods outperforms any one method by itself. In some sense, a direct comparison with these PSSMs is not entirely fair since a majority of these weight matrices were created by motif discovery procedures rather than directed experimentation (such as DNA footprinting). Also, carefully constructed variants of PSSMs, which may take into account motif conservation in multiple species or interdependence of bases, can offer state of the art motif detection. Unfortunately, sufficient data is not always available to build such detailed models. The purpose of our comparison is simply to highlight the improved performance of classification methods relative to the commonly available binding site models. Figure 8 shows the result of a comparison between simple PSSM scanning using the MotifScanner algorithm and predictions by SVM on combined data. The leftmost grouping is a result from scans using PSSMs for all 104 TFs against the positive and negative sets on which the SVMs were trained. A score threshold was chosen for each TF so that the specificity on the training set was held to 0.95. This makes comparison to the SVM classifiers more straightforward as overall specificity for the SVMs is 0.95. The grouping on the right restates the performance of the SVMs with 26 combined datasets on the full set of positives. The SVM classifiers outperform PSSMs in the number of detected positives. It is clear that loosening the thresholds for the PSSMs would allow for better coverage but degrade performance by increasing the number of false positive predictions. Support vector machine classifiers offer a good balance between sensitivity and false prediction.

Figure 8

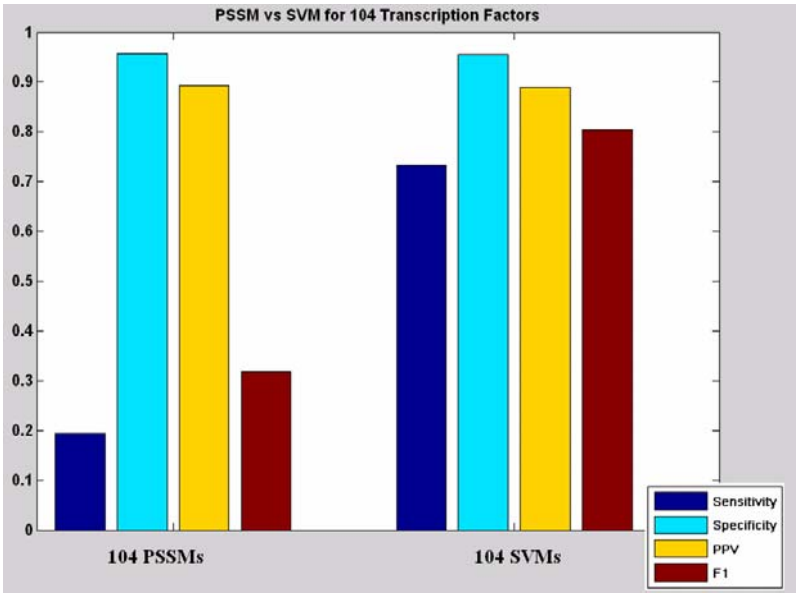

Figure 8 SVM vs PSSM scan.

Left: PSSMs for 104 TFs scanned against positive and negative sets. Overall specificity is held constant to 0.95 to match that of the SVM results.
Right: Overall results for SVM classifiers trained on weighted combination of 18 datasets.

Biological insights-*Pathway Control*

Finally, we have applied the combined classifier for each TF to all promoters in the yeast genome in order to expand the known binding repertoire of each factor. On average, each classifier produced approximately 884 new targets. Although it is unlikely that this set is free of false positives, examining the data in the context of biochemical pathways can shed light on significant predictions, which can quickly elucidate new sites which are good candidates for further study.

Gcn4 is a transcription factor in yeast known to control genes in the amino acid biosynthetic pathway[41], and SVM predictions match well with the known biology of Gcn4 control mechanisms. The final classifier for this TF has an F1 score of 0.89, sensitivity of 0.86, and PPV of 0.92. This TF is a master regulator which has known targets in at least 12 amino acid biosynthetic pathways and has been shown by gene expression to induce at least $1/10^{th}$ of the yeast genome[42]. Figure 9 highlights some known targets of Gcn4 in methionine/threonine biosynthesis in the aspartate family pathway. Branch-points from this pathway can ultimately lead to the amino acids methionine, threonine, lysine, and isoleucine. This group is of particular interest to humans since they are essential and not synthesized in the human metabolism. Gcn4 is known to regulate Hom3, Thr1 and Thr4 leading to threonine, lysine, and isoleucine. However predictions by SVM indicate it also directly targets committed steps of methionine biosynthesis by binding Met2, Met17, and Met6, which are interesting targets for further study.
Figure 9

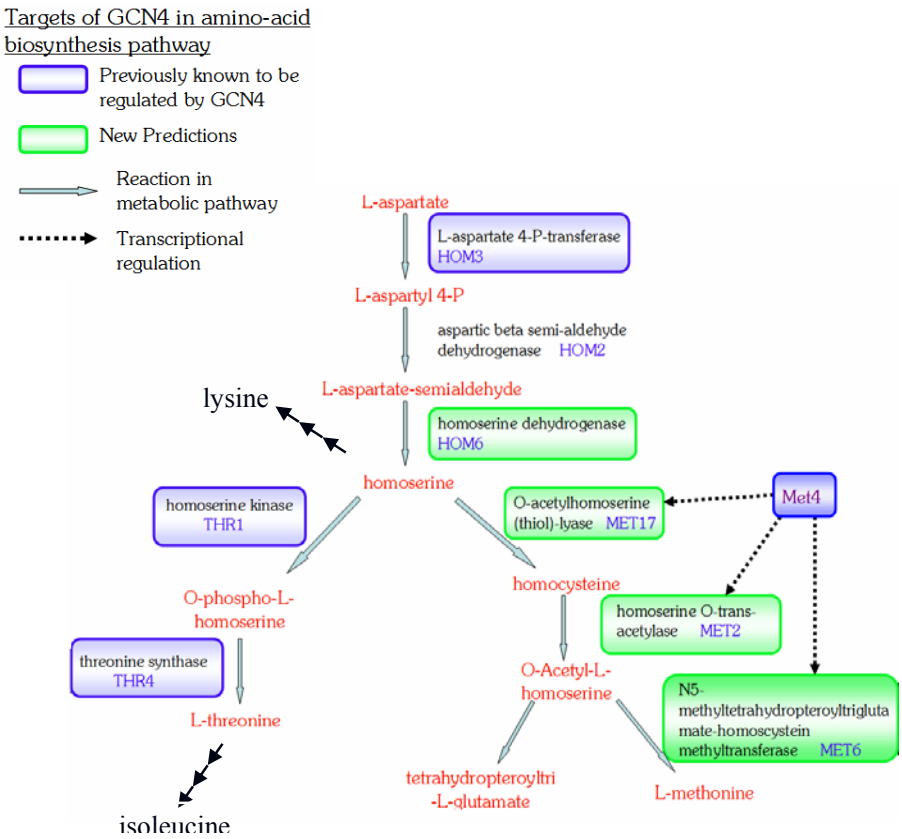

Figure 9 GCN4 and amino-acid biosynthesis.

Predictions by SVM match well with the known biology of Gcn4 control mechanisms. Pathway map generated taken from the Pathway Tool Omics Viewer at SGD[62].

Previously Gcn4 was known to indirectly influence synthesis of methionine by activating Met4, a transcription factor specific to methionine biosynthesis and sulfur metabolism[43]. It is feasible that regulation of these enzymes by both Gcn4 and target Met4 represents a transcriptional feed-forward loop. Such loops have been described before and can be advantageous to an organism by exhibiting sign-sensitive delay, since it may be useful to have a quick response when shifting to an OFF state and a slow response when turning back ON[44].

The Rap1 DNA binding factor is a widely known regulator in the cell cycle, acting as a repressor or activator depending on its context. Rap1 is also a key element in the structure of yeast telomeres, where it plays a role in telomere silencing[45]. In a seemingly contradictory role, Rap1 has also been shown to regulate several glycolytic enzymes, as shown in Figure 10. The specificity of this glycolytic regulation is dependent on a second factor, Gcr2, which binds to the Rap1/Gcr1 complex but does not contact DNA directly[46]. New predictions by SVM in the pathways of sugar metabolism show good correspondence with expectations for Rap1 (Figure 10). Most interestingly, the new predictions include both isoforms of the enzyme phosphofructokinase. This step, where fructose-6-phosphase is converted

into fructose-1,6-bisphosphate, is the crucial step in sugar breakdown where most metabolic flux through the pathway is controlled[47].
Figure 10

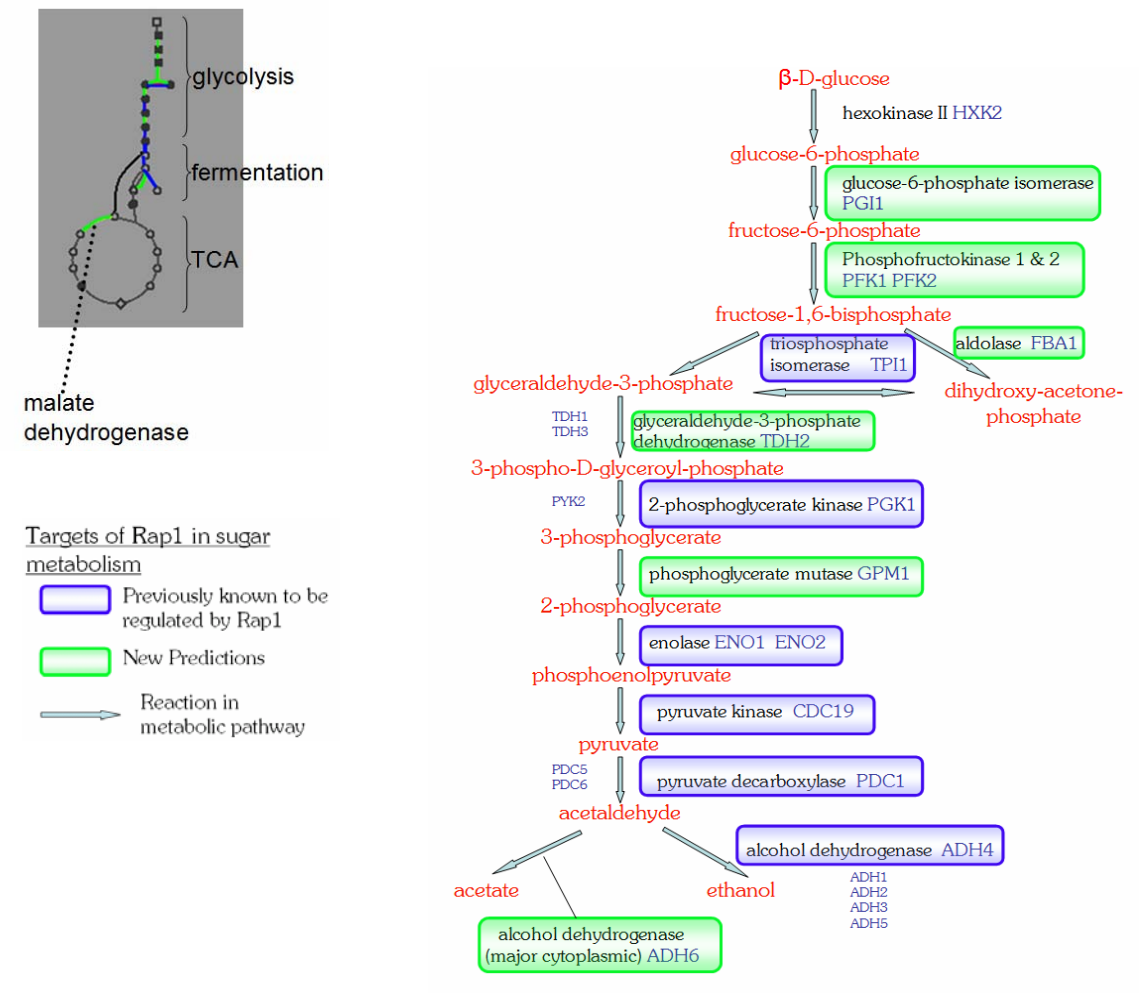

Figure 10 Rap1 and glycolytic/TCA cycle reaction.

Glycolysis leading to acetate and ethanol are shown. The gray box on the left contains a pathway overview of glycolysis, fermentation and the TCA cycle, where red connections are known and yellow are predicted. Rap1 can be seen to regulate key control points in glycolysis and the TCA cycle. Pathway map generated taken from the Pathway Tool Omics Viewer at SGD[62].

Also of significance is the prediction that Rap1 regulates malate dehydrogenase in the TCA cycle. Malate dehydrogenase is unique in the TCA cycle in that it has a very small equilibrium constant, meaning that the forward reaction from malate to oxaloacetate is highly unfavorable. This is generally overcome during aerobic growth since the subsequent reaction is extremely favorable (large free energy release). However, in the absence of oxygen the cell still requires certain

intermediates which can now not be made in the normal way. Running the malate dehydrogenase reaction in reverse, a favourable direction, can provide a way to synthesize these intermediates[47]. Rap1 is already known to regulate the conversion of acetaldehyde to ethanol via alcohol dehydrogenase, and the possible complementary control of malate dehydrogenase suggests a possible role for Rap1 in regulation of fermentative growth.

Conclusions

We have seen that support vector machines can accurately classify transcription factor binding sites using a wide range of genomic data types. Combining various information sources can reduce false positives and incrementally increase sensitivity, while post-processing of the data to assign posterior probabilities allows the selection of high confidence targets. Although the maximal margin of SVMs is resistant to over-fitting, it can be further abrogated by selecting the best features for classifier construction. Feature selection and clustering techniques can be used in future work to refine predictions and more efficiently compare the SVM to other learning machines (KNN, Bayes, and Random Forest) which don't easily handle high dimensional or correlated data..

Based on *k*-mer data, SVMs appear to be isolating appropriate features for classification where many known transcription factor binding sites overlap with highest ranked *k*-mers. Examination of melting temperature classifiers for YJR060W demonstrates the unique biological features of targets for that TF. Similarly, expression-based classifiers for Fhl1 show the conditions under which Fhl1 acts on its targets, pointing the way to testable hypotheses supported by data in the literature. Finally, targets of Gcn4 and Rap1, when put into the context of biological pathways, correspond well to published experiments and show the effectiveness of integrated classifiers for building system-wide gene regulatory networks. Future work will then involve development of methods to discover biologically significant features in different datasets based on classifier performance and intelligent dimension-reduction techniques to reduce noise and improve accuracy.

Methods

We have tested a variety of sequence and non-sequence based classifiers for predicting the association of TFs and genes ([33], submitted for publication). All together 26 separate data sources (each yielding a feature map and kernel) are combined to build classifiers for each transcription factor. The 26 data sources comprise a family of sequence-based methods (e.g., *k*-mer counts, TF motif conservation in multiple species, etc), expression data sets, phylogenetic profiles, gene ontology (GO) functional profiles, and DNA structural information such as promoter melting temperature, DNA bending, and DNA accessibility predictions (see Table 2).

Our positive and negative training sets are taken from ChIP-chip experiments[48, 49], Transfac 6.0 Public[50], and a list curated by Young *et al*., from which we have excluded indirect evidence such as sequence analysis and expression correlation (http://staffa.wi.mit.edu/cgi-bin/young_public/navframe.cgi?s=17&f=evidence). Only ChIP-chip interactions of *p*-value $\leq 10^{-3}$ (i.e., a high confidence level) are considered positives [48]. The Transfac and curated list represent a manually annotated set which will later be used

separately during SVM comparison to PSSM performance. For the purposes of SVM, however, all manually curated and high-throughput sets are grouped together, making a total of 9104 positive interactions.

Negative sets pose a greater challenge since no defined negatives exist in the literature; however, since a particular TF will regulate only a small fraction of the genome, a random choice of negatives seems acceptable. In fact, test cases with a few TFs show good classification performance with random negatives (unpublished work). Nevertheless, a safer set of negatives would be those showing no binding by experiment under some set of conditions. Along those lines, we have chosen for each TF 175 genes with the highest *p*-values (generally > 0.8) under all conditions tested in genomic ChIP-chip analyses[48, 49]. Clearly all experimental conditions have not been sampled and this does not guarantee that our choices are truly never bound by the TF, but this choice of negatives should maximize our chances of selecting genes not regulated by the TF of interest.

All promoter sequences have been collected from RSA tools, Ensembl, or the Broad Institute's Fungal Genome Anatomy Project[51-53]. For yeast, promoters are defined as the 800 base pairs upstream of the coding sequence. The motif hit conservation dataset required promoter regions from 17 other genomes. Those genomes, their sources, and the length of the promoter regions are described in our previous report[25]. Sequences are then masked using the dust algorithm and the RepeatMasker software[54, 55] where appropriate, to exclude low complexity sequences and known repeat DNA from further analysis. PSSM scans (for datasets 1 and 2, below) are performed with the MotifScanner algorithm[56]. MotifScanner assumes a sequence model where regulatory elements are distributed within a noisy background sequence[56]. The algorithm requires input of a background sequence model, which in this case is a transition matrix of a 3rd order Markov model generated from the masked upstream regions of each genome. MotifScanner only requires one parameter be set by the user, i.e. the threshold score for accepting a motif as a binding site. Several thresholds have been tested and the results we have used to create SVM kernels are all at a setting of 0.15, which has been found to be a reasonable middle ground, making approximately 560 predictions per TF. Settings beyond 0.2 produce too many false hits to be useful. The PSSMs themselves are obtained from Transfac 6.0 Public and from[57], which are a mix of experimentally derived motifs and those generated by motif-discovery procedures.

Datasets using *k*-mers rather than PSSMs are generated using the fasta2matrix[58] program which lists all possible *k*-mers and counts the occurrence of each within a set of promoters. Gapped *k*-mers are detected using custom scripts written as Matlab m-files. The expression data used include 1011 microarray experiments compiled by Ihmels and co-workers, which can be downloaded with permission from the authors[59].

Each data set is normalized so that each feature in the training set has mean zero and standard deviation one. Gene Ontology, phylogenetic profile, and TF-target correlation data are not normalized since their data are binary. Finally, since the ultimate goal is data integration the number of training examples for a given TF must be the same for every dataset used to make a classifier. When examples are missing in a dataset, as is the case with the GO and COG (phylogenetic profiles based on the Clusters of Orthologous Groups database) based classifiers, random values sampled from the rest of the training set are used to fill in the missing vectors.

All classifier construction and validation was performed in Matlab[60] using the Spider machine learning library[61]. Mapping of predicted binding targets to biological pathways was done using the Pathway Tools Omics Viewer at SGD[62].

Description of Analysis

A separate classifier is developed for each TF based on each independent dataset. The four kernel functions in Table 1 (linear, rbf, Gaussian, and polynomial) are tested using leave one out cross validation, and the function with the highest F_1 score (below) is chosen as best for that particular TF-dataset combination. A flow diagram of our method can be seen in Figure 1. Let *TP* denote the count of true positives, *FN* false negatives, etc. The F_1 statistic is a robust measure that represents a harmonic mean between sensitivity $S = \frac{TP}{TP + FN}$ and positive predictive value $PPV = \frac{TP}{TP + FP}$. It is defined by

$$F_1 = \frac{2 \times S \times PPV}{S + PPV} = \frac{2 \times TP}{2 \times TP + FP + FN}$$

If we choose the classifier with the best F_1 statistic, each TF now has one classifier for each type of genomic data (26 classifiers total). For every classifier the *C* parameter (the trade-off between training error and margin) must be specified, and some kernel functions require a second parameter, e.g., the polynomial degree *k* for a polynomial kernel or a standard deviation σ (which controls the scaling of data in the feature space) for a Gaussian or radial basis function (RBF) kernel. The values for these parameters are chosen by a grid-selection procedure in which many values are tested over a specified range using 5-fold cross validation. The ROC score is used to choose the best values. As an example for an RBF kernel a range of *C* values from 2^{-5} to 200 is tested with a range of σ values from 2^{-15} to 2^3. The best combination of values is then chosen to make the final classifier.

The performance of any parameter-optimized classifier is determined using leave-one-out cross validation. Once the best kernel function $K(\mathbf{x}, \mathbf{y})$ (with optimized parameter values) has been chosen for a particular TF-dataset pair, the next step is to combine the datasets to create a composite classifier. To that end, the $K(\mathbf{x}, \mathbf{y})$ is used to create a kernel matrix for each of the 26 datasets. Before weighting and combining kernels for each data set, all kernel matrices are normalized according to

$$\widetilde{K}(x, y) = \frac{K(x, y)}{\sqrt{K(x, x)K(y, y)}}$$

This normalization effectively adjusts all points to lie on a unit hypersphere in the feature space *F*, and the diagonal elements in every kernel matrix the will be 1. This assures that no single kernel has matrix values that are comparatively larger or smaller than other kernels, so all matrices initially have the same contribution to the combination.

Datasets can be combined by adding kernel matrices together; however, an unweighted linear combination ignores dataset dependent performance—in fact some datasets do not perform better than random for some TFs. To avoid this problem, we determine whether the number of true positives predicted using a particular dataset is significantly different ($p \leq 0.05$) than what would be achieved by random guessing. We calculate the probability of observing fewer than *g* true positives given the

training set size N, the total number of known positives L (i.e., $TP + FN$), and the number of positively classified examples, M (i.e., $TP + FP$).

$$p = P(g \geq x) = 1 - F(x-1 \mid N, L, M) = 1 - \sum_{i=0}^{x-1} \frac{\binom{L}{i}\binom{N-L}{M-i}}{\binom{N}{M}} \quad \text{for } x > 0;$$

$p = 1$ otherwise

Here p is the probability of drawing x or more true positives at random. Datasets that do not meet the p-value cutoff are eliminated from the analysis for a particular TF.

Finally, the significant datasets (each represented by a kernel matrix K_{ij}) must be weighted based on their performance. Using a scheme (described below) with weights equal to the F_1 score of each classifier, the underlying 26 kernel matrices are scaled and added into a single unified kernel corresponding to the given transcription factor. Once the weighting is complete, an overall leave-one-out cross-validation is employed to estimate the error of the combined classifier.

Three simple weighting schemes have been compared. In all cases the primary weight for a method is determined by computing its ratio with the best performing method. Our first weighting scheme is linear and simply multiplies the m^{th} matrix $K^m = K_{ij}^m$ by its scaled F_1 score α_m and computes a sum, yielding $K = \sum_{m=1}^{26} \alpha_m K^m$. A second scheme is nonlinear and squares the weights of the first method before multiplying, yielding $K = \sum_{m=1}^{26} \alpha_m^2 K^m$. This will not change the weight of the best performing method, which will be scaled to 1, but will decrease the relative weights of poorer methods. Our third scheme, which is the most nonlinear, takes the squared tangent (an effective sigmoidal function) of the primary weight, yielding $K = \sum_{m=1}^{26} \left(\tan^2 \alpha_m\right) K^m$. This more steeply penalizes poorly performing methods while increasing relative weights of the best methods (e.g., instead of weight 1, the best method will have a weight of 2.43).

Genomic Datasets

1 PSSM Motif counts (MOT,Table 2 item 1)

Position-specific weight matrices (PSSM) for 104 transcription factors have been used to scan 800bp promoters in S. cerevisiae for each gene in a training set, and the number of hits for each PSSM has been counted. These counts are the features (i.e., components) of 104 dimensional feature vectors. It is clear that a greater number of "hits" by a PSSM in the upstream region of a gene will imply a greater likelihood that the TF corresponding to the matrix will actually bind the gene. For each prediction there is a probability that it will be true, P(True|hit). If a certain upstream region of a gene has more than one hit, the probability that the TF binds to the gene should increase (Figure 11). This method aims to better predict TF binding by taking into account the number and types of binding motifs in a promoter.

Figure 11

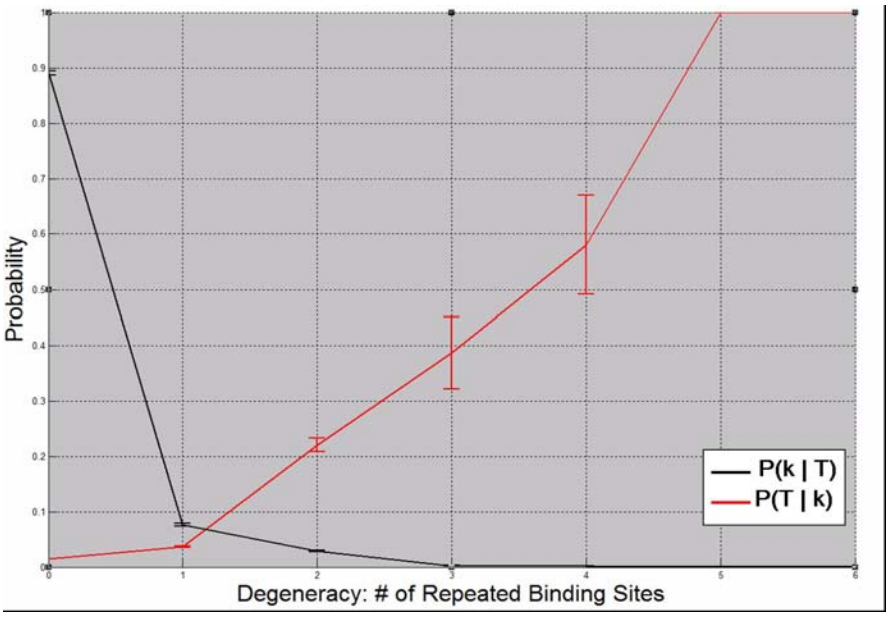

Figure 11 Motif Counts

Having more than one detected binding site for a TF in the upstream region of a gene increases the likelihood that the TF truly binds the gene. Higher counts of motifs yield fewer predictions; however, as the number of repetitions of a motif increases, the probability that the TF binds approaches 1.
$P(k|T)$ = Probability of motif count k given a set of True binding sites.
$P(T|k)$ = Probability of finding a True binding site given a motif count of k. Data is averaged over 104 TFs.

2 PSSM Hit Conservation (Table 2 item 2)

Comparative genomics tools have recently been applied with much success to the identification of transcription factor binding sites. Because most regulatory elements are in non-coding regions and show considerable variation in sequence even for the same TF, they aren't easily recognizable. However, binding sites are often preserved through evolution, and thus become apparent in what authors call a "footprint" in alignments of orthologous regions from different genomes. Cis-element conservation is a powerful way to detect functional non-coding elements, and, in this case, are modified and applied to 18 genomes ranging from yeast to human. Conservation of a TF binding site is determined by counting hits of the TF probability matrix in orthologous upstream regions from several organisms. Orthology information was taken mainly from the Homologene database[63] for all organisms except for *sensu stricto* and *sensu lato* yeasts, which was obtained from Washington University and the Whitehead Broad Institute [51, 64-66].

Previous studies have defined conservation as direct nucleotide correspondence in *aligned* orthologous regions. In previous publications[64] this analysis has involved manual inspection and modification of low scoring alignments, an approach that would be cumbersome and time consuming with a larger number of genomes. Other authors rely on whole genome alignments of closely related species to identify orthologs and conserved upstream regions[65]. This strategy would be

difficult if not impossible for genomes farther diverged than the few closely related yeast species. In this analysis, a hit by a PSSM in the upstream region of an ortholog is defined as a conserved motif. In this way, conservation of a *potential binding site* is being measured rather than the exact nucleotide string. This is because a PSSM may identify sequences that are different in nucleotide composition but still match the probability matrix. This is a looser conservation criterion that makes sense biologically, since natural selection will act to preserve a binding site, and not necessarily an exact nucleotide string.

The stronger the conservation of a potential binding site, the more likely the site is to be real (Figure 12). The empirical probability of a true site increases to 100% as the binding site conservation level reaches 15 genomes. Again, these data are assembled into a 104 dimensional feature vector for each gene in yeast. Each feature represents a transcription factor motif and the value of the attribute is the number of genomes in which the binding site is conserved.

Figure 12

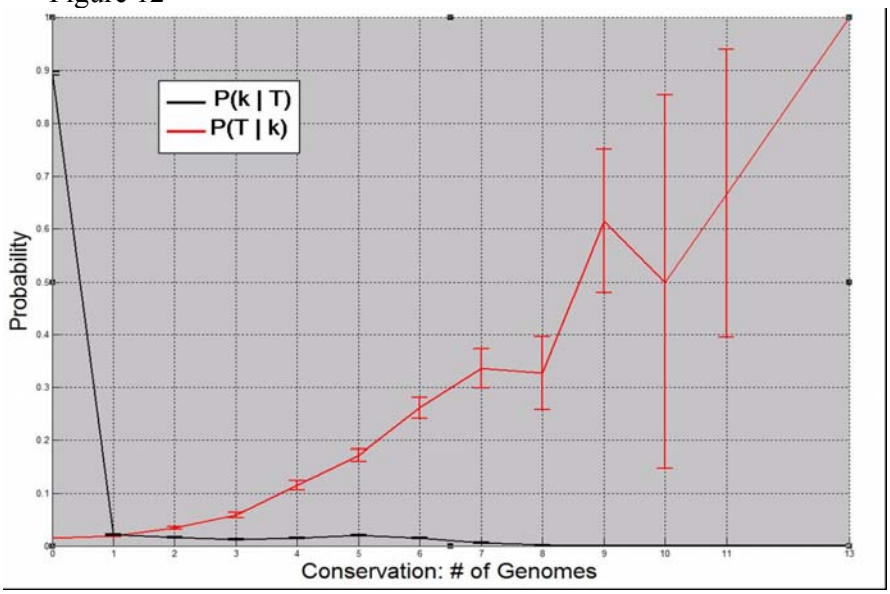

Figure 12 Conservation

Conservation of a TF binding site in several orthologous upstream regions increases the likelihood that a potential site is a True site. Data is averaged over 104 TFs.

$P(k|T)$ = Probability of site conservation in k genomes given a set of True binding sites.
$P(T|k)$ = Probability of finding a True binding site given that it is conserved in k genomes.

2 Kmers, Mismatch kmers, and Gapped kmers (Table 2 and 6-16)

PWMs may fail to detect binding sites if the binding site collection used to generate them is incomplete (in the case of experimental data) or if the motif discovery procedure is inaccurate (as may occur in the case of computationally generated matrices). In this case, the distribution of all *k*-mers in a gene's promoter may be used to predict whether it is bound or not-bound by a TF. *K*-mer counts in promoters have been used previously with SVMs to predict genes' functions[24]. Here, several strategies are used to generate a variety of datasets based on *k*-mer strings. First, one dataset of feature vectors is created by decomposing all yeast promoters into counts of all *k*-mers of length 4, 5, and 6. Similarly, 6-mers with variable length center gaps (of the form *kkk*-$\{x\}_n$-*kkk*) are counted in each promoter to form sequence datasets allowing gaps of size 1 through 8 (Table 2, items 4-11). This allows detection of split motifs such as the binding site for Abf1, RTCRYNNNNNACGR. Finally, we construct two datasets with 6-mer counts allowing one mismatch in any 6-mer (Table 2 items 12-13). A mismatched base pair is counted with a value of 0.1 in the first dataset, and 0.5 in the second.

Given a set of true positives and true negatives for each TF, the SVM classifies genes based on their complete promoter content as represented by these *k*-mer distributions. As we discuss below, *k*-mer counts are the single best performing method for distinguishing transcription factor targets.

It should be noted that our kernels derived from sequence data are very similar to sequence kernels used in previous work. The *k*-mer kernel produced from 4,5,and 6-mers is analogous to the spectrum kernel previously used to classify protein sequences[67]. Whereas the spectrum kernel would be calculated separately for *k*-mers of length 4, 5, and 6, we have concatenated these features into one kernel. Also proposed for sequence classification, the (*g*,*k*)-gappy kernel can represent any *g*-length sequence with *k* gaps[68] much like the gapped sequence kernel used in our work. The difference in this case is that a (g,k)-gappy kernel allows gaps in any sequence position while we allow only central gaps splitting a motif into two sub-strings. While the gappy kernel may be more appropriate for protein classification, the split *k*-mers used here more resemble transcription factor binding patterns. Mismatch kernels similar to ours have also been previously described for classification of protein function and detection of remote homology[69]. Finally, unlike kernels applied to protein data, the kernels used here take into account the reverse complements of each *k*-mer. This means, for instance, that the 3-mers "AAA", and "TTT" are counted together as one unit since the presence of one necessitates the other on the opposite strand of DNA.

3 GO Annotation (Table 2 item 5)

GO term annotation can be used to detect possible transcriptional targets. The targets of a transcription factor have often been shown to have similar function and a gene's GO annotation can be used to measure its functional similarity to known targets[70]. For this method, all GO Biological Process terms in yeast become features for genes, such that every gene will have a binary vector, with a 1 for the terms which are annotated to it, and 0 otherwise. Parent terms of direct annotations also receive a 1. There are 2155 possible terms for yeast, giving a vector of the same length. Since only about one-third of yeast genes are annotated with GO terms, a feature matrix generated with GO data is sparse, consisting mostly of zeros. Imputing zeros for genes unannotated in GO can potentially bias the result of the classifier (for instance, if many negatives are missing and hence become zero vectors it may be trivial to separate these from the positives). Instead, the binary vector is filled in with

random data according to the background distribution of term annotation in the yeast genome. Despite using random data, the vectors are still sparse and the best 800 GO terms are selected using the Fisher score criterion during the training of each TF. The Fisher criterion gives high scores to features that have large differences in mean between the positive and negative classes in relation to variance. For a specific feature *d*, calculate the mean and standard deviation of that feature in each class (μ_d+ σ_d+ and μ_d- σ_d-). The Fisher score for feature *d* is then

$$\text{Fisher score} = \frac{(\mu_d^+ - \mu_d^-)^2}{(\sigma_d^+)^2 + (\sigma_d^-)^2}$$

This feature selection is performed in the Spider data mining package[71].

4 Phylogenetic Profiles (Table 2 item 3)

Co-evolution of a transcription factor's targets may indicate regulation. A phylogenetic profile of a gene is simply the pattern of occurrence of its orthologs across a set of genomes. Genes with similar patterns have been shown to participate in the same physical complexes or have similar biochemical roles within the cell[72]. It has also been postulated that transcription factors and their targets co-evolve[73]. Therefore it seems reasonable that a group of commonly regulated genes could share a similar pattern of inheritance. Phylogenetic profiles here were parsed from the COG database, which contains orthology information between *S.cerevisiae* and 65 other microbial genomes. Each gene in the positive and negative set is represented by a 65 component binary vector, a component being 1 if the gene's ortholog is present in the corresponding genome, and zero otherwise. As with the GO data, gene attribute vectors are binary, containing 65 elements, one for each genome in COG. Also, since many genes have not been annotated to COG groups, it is necessary to generate random vectors for missing genes as described for the GO example above.

5 TF-Target Expression Correlation as a Method to Predict Regulation

Analysis of transcription factor motif-matching outputs shows that false positive predictions are numerous even in cases of low sensitivity. Expression analysis provides a means to discover targets missed by sequence based methods. Several studies have shown that genes with similar expression patterns are likely to share similar regulation and, conversely, genes regulated by the same TF are more likely to be co-expressed[70, 74].

Two strategies are often useful for discovering transcription factor targets using expression data. Often genes are turned on and off as the expression levels of their controlling TFs are altered. Thus one method is to find targets of some TFs by finding TF/gene pairs that have correlated expression patterns[6]. A second approach involves identifying groups of co-expressed genes, and hypothesizing that this co-expression is due to co-regulation by the same TF(s)[75, 76]. In the two sub-sections below, we describe how each of these strategies can be used to construct data vectors for SVM learning.

5.1 TF-Target Correlations Measured by Profile Entropy Minimization (Table 2 item 17)

Since genes may be combinatorially regulated by different TFs under different conditions, regulator-target relationships can be diluted in large expression profiles. Such relationships will only be discovered if their condition-specific correlation can be found.

A new approach [77] addresses this problem by searching for the conditions under which a regulator's profile is maximally associated with a target's profile, essentially choosing the set of experiments where the TF most clearly and significantly controls the expression of a potential target. This is accomplished by minimizing the Kullback-Liebler entropy between the TF and potential target's expression profiles when looking at different subsets of conditions (thereby choosing the set of conditions under which the TF/target have the most significant correlation: see references for further details[77]). In this analysis correlations with a *p*-value of 10^{-10} are chosen in order to extract the most significant regulatory relationships and reduce false predictions. Significant relationships are coded as 1's in gene's feature vector, so that every gene is described by a binary list whose length is the number of TFs (104 in this case).

5.2 Target-Target Correlations (Table3 item 4)

For purposes of representing expression correlation, we use normalized log2 ratios for each gene across 1011 experiments[78]. Each gene's expression profile is normalized to a mean of 0 and standard deviation of 1. This normalized expression profile is then the vector of features used by the SVM to represent any example gene (each gene will have 1011 features corresponding to expression coditions). In this case, the dot product between such gene vectors is analogous to a Pearson correlation and naturally fits into the SVM framework, which uses dot products to associate positive and negative examples. Given many known targets of a transcription factor as positive cases, the SVM can identify a new target based on how closely its expression resembles that of the known examples.

6 Sparse binary encoding of promoters (Table 2 item 18)

Efforts to encode strings into kernel representations have progressed for many applications. The mismatch, gap, and *k*-mer kernels mentioned above have been used mainly for protein classification, translation initiation site detection, and mRNA splice site identification. Another straightforward sequence representation is the sparse bit encoding[21]. In this simple scheme each nucleotide in a sequence is encoded by 4 bits, only one of which is set to 1. The nucleotide is identified as A, C, T, or G based on the position of the "1" in each such set. This leaves an $800 \cdot 4 = 3200$ dimensional vector to describe each example sequence, and the dot product of two vectors results simply in the number of nucleotides shared between the two sequences.

7 Promoter Curvature and Bend Predictions (Table 2 items 19 and 26)

It is well known that sequence-dependent DNA bending can be an important aspect of protein-DNA interactions. Some prominent examples of proteins that induce DNA bending are the TATA-binding protein (TBP)[79], catabolite activating protein (CAP), and the yeast Mcm1 transcription factor[80]. A specific sequence of nucleotides that is more prone to bending into the proper configuration would provide a ready-made site for transcription factor binding. The particular bend and curve properties of known target genes may help discriminate them from non-targets.

Using the "Banana" algorithm in the EMBOSS toolkit, bend and curvature predictions were made along the promoters of all yeast genes. These were used as two separate genomic methods from which to generate classifiers for all 104 TFs one based on bend predictions and one based on curve. Specifically, bending refers to the tendency of adjacent base pairs to be non-parallel (twists and short bends of ~3bp), whereas curvature refers to the tendency of the double-helix axis to follow a non-linear path for a distance of several base pairs (broad loops and arcs, ~9bp window).

Banana follows the method of Goodsell and Dickerson[81] which is consistent with published experimental data[82]. Briefly, base pairs in DNA are typically perpendicular to the helical axis, meaning that the curvature and bend of the DNA can be roughly described by the vectors normal to each base pair. Given a sequence and a table of the standard roll, tilt, and twist angles between base pairs (based on experimental measurements), the Banana algorithm calculates the magnitude of these normal vectors at each base pair relative to the previous pair. Bend is calculated from these normal vectors as an average over a 3bp window. These values become the feature vectors for examples in SVM classification based on DNA bend (i.e., each window is one feature). For curvature, an average over a 10bp window is used to smooth out variations in magnitudes (for a nucleotide position *n*, an average is taken from *n* - 4 to *n* + 4 with the -5 and +5 positions averaged in at half weight, determining the value assigned to position *n*). After this smoothing step, the curvature at *n* is calculated as the angle between base pairs at positions *n* - 15 and *n* + 15. This vector of angles is used as the feature vector for a training example in SVM classification. For more details on the method see [81] or reference the EMBOSS website (http://emboss.sourceforge.net/apps/banana.html).

8 Homolog Conservation (Table 2 item 20)

This method is akin to the phylogenetic profiles taken from the COG database described above. Because COG uses a strict definition of orthology, namely bi-directional best hits within a group of at least three organisms, many genes are not allocated to any ortholog group. The method described here relaxes the definition of orthology to allow a profile to be constructed for any gene, while still discriminating between well-conserved sequences and weakly conserved sequences[83]. These phylogenetic profiles are constructed using BLASTP to compare yeast proteins to 180 prokaryotic genomes. The resulting best hit E-values are then discretized by placing them into one of six bins based on empirically determined E-value cutoffs. The bin numbers range from 0 (no significant hit) to 5 (very significant). Thus, a typical example gene will have 180 features, each corresponding to a different genome, with values ranging from 0 to 5 indicating the strength of the best BLASTP hit of that gene's protein to another genome.

9 Hydroxyl Cleavage –DNA Accessibility (Table 2 item 21)

It is possible that strands of DNA sharing little sequence similarity may still share common structural motifs. Transcription factors may seek out these structural cues for binding, thereby identifying conserved structural motifs when no strong consensus sequence can be detected. Experiments show that hydroxyl (OH) radical cleavage is an effective probe for DNA structure, in that strand breaking mirrors the accessible surface areas of the sugar-phosphate backbone[84-86]. A database of DNA sequences and their hydroxyl cleavage patterns has been published[85]. This database allows accurate prediction of backbone accessibility for any sequence by sequentially examining every 3-mer in a sequence and looking up its experimental cleavage intensity as measured by phosphor imaging of cleaved, radio-labeled DNA separated by electrophoresis[84].

Predictions of this sort are generated for all sequences in the yeast genome and the individual 3-mer cleavage intensities along each promoter serve as feature vectors for TF-target classification by SVM. This method should prove useful in identifying potential targets when *k*-mer counts and other sequence based methods fail.

10 Kmer median positions from start (Table 2 item 22)

A potential transcription factor binding site may be functional only when within a certain distance from other binding motifs or from the start site of transcription. When such positional constraints exist, they can be used to filter out sites which would otherwise become false positive predictions.

For each *k*-mer in a sequence, we record its median distance from the transcription start. This dataset will be useful in classifying targets for a transcription factor only if the factor shows positional bias in promoter binding.

11 K-mer likelihoods (Table 2 item 23)

Although *k*-mer counts may describe promoter composition, the abundance of non-informative sequences may hide the few *k*-mers which meaningfully contribute to class separation. Those *k*-mers which are statistically over-represented in a promoter can often be transcription factor binding sites, and this fact has been effectively used to identify biologically significant patterns[87-89]. For every possible *k*-mer 4, 5, and 6 long we calculate the probability that the *k*-mer has *x* occurrences in a gene's promoter. The negative log of these probabilities are then the features used for SVM classification.

Background *k*-mer counts are obtained from RSA tools. The prior probability (*f*) for a *k*-mer to be found in any position is calculated by dividing the total number of counts in the background sequence set by the total number of possible positions in the background set (here, the background set is the full set of 800bp yeast promoters). Given this prior probability for a *k*-mer, the expected number of occurrences of the *k*-mer in any sequence can be calculated by

$$m = f(L - k + 1),$$

where L is the length of the sequence and k is the length of the *k*-mer.

The goal is then to calculate the probability of finding the observed number of counts by chance given the expected number for a promoter. This can be done simply by using the probability density function of the Poisson distribution with mean m. This method for calculating *k*-mer likelihoods is similar to the method described in[90]. Thus, for each gene, a *p*-value will be calculated for each *k*-mer which represents the likelihood that the *k*-mer appears as many times as observed by chance. A feature vector for a gene is then the vector of probabilities describing all *k*-mers.

12 Promoter Melting Temperature Profile and Promoter Delta G profile (Table 2 items 24 and 25)

It is widely known that the initiation of transcription by polymerase involves melting of the DNA double helix. Several experiments have indicated that differences in melting temperature (Tm) of DNA can influence the rate of transcription by assisting or obstructing DNA melting by polymerase[34], and there is evidence that torsional strain can play a role in duplex destabilization and opening[35]. Furthermore, it has been shown that sites thought to be susceptible to stress-induced duplex destabilization (SIDD) match well with gene regulatory regions[36]. It is therefore possible that transcription factors binding DNA may induce conformational adjustments in the promoter which slightly alter the stability of the helix. This change in stability may indirectly change the frequency or likelihood of transcription initiation. Indeed, recent models have shown correlation between sites of local promoter melting, regulatory sites, and initiation sites[91].

If certain transcription factors influence a target's expression by altering promoter stability, its targets may contain a specific melting temperature or free-energy signature in their promoter regions. This signature could potentially distinguish targets from non-targets much as sequence motifs do. To include this information in a classifier the EMBOSS[92] toolbox is used to calculate the melting and free energy profiles of all yeast promoters using a sliding window of 20bp. Thus, for every 20bp increment along each upstream region, a Tm value and a Gibbs free energy (ΔG at 25°C) is calculated. For these calculations EMBOSS uses the nearest-neighbor thermodynamics from [93, 94]. The Tm profile and the free energy profile become separate feature vectors for each gene, thereby providing two additional datasets which can be used for classification.

PSSM Comparision

Using the same positive and negative sets as for the SVM procedure, PSSMs are also used to make predictions across the yeast genome at various score thresholds to serve as a comparision to predictions made by SVM. The data in Figure 8 represent only one threshold, a value of 0.1 as the prior parameter in MotifScanner (low parameter values retain the best matches whereas values near 1 allow very loose hits). Other choices of threshold do not appear to improve performance. Loosening the threshold begins to dramatically increase false positive predictions beyond a prior of 0.2. By making detection more strict, false predictions are reduced along with sensitivity.

References

1. EM Conlon, XS Liu, JD Lieb, JS Liu: **Integrating regulatory motif discovery and genome-wide expression analysis**. *PNAS* 2003, **100**:3339-3344.
2. S Keles, MJ van der Laan, C Vulpe: **Regulatory motif finding by logic regression**. *Bioinformatics* 2004, **20**:2799-2811.
3. W Wang, JM Cherry, D Botstein, H Li: **A systematic approach to reconstructing transcription networks in Saccharomycescerevisiae**. *PNAS* 2002, **99**:16893-16898.
4. H Bussemaker, H Li, E Siggia: **Regulatory Element Detection Using Correlation with Expression**. *Nature Genetics* 2001, **27**:167-171.
5. K Birnbaum, PN Benfey, DE Shasha: **cis Element/Transcription Factor Analysis (cis/TF): A Method for Discovering Transcription Factor/cis Element Relationships**. *Genome Res.* 2001, **11**:1567-1573.
6. Z Zhu, Y Pilpel, G Church: **Computational Identification of Transcription Factor Binding Sites via a Transcription-Factor-Centric-Clustering (TFCC) Algorithm**. *Journal of Molecular Biology* 2002, **318**:71-81.
7. M Pritsker, Y-C Liu, MA Beer, S Tavazoie: **Whole-Genome Discovery of Transcription Factor Binding Sites by Network-Level Conservation**. *Genome Res.* 2004, **14**:99-108.
8. S Elemento, S Tavazoie: **Fast and systematic genome-wide discovery of conserved regulatory elements using a non-alignment based approach**. *Genome Biology* 2005, **6**.
9. M Tompa, et al.: **Assessing computational tools for the discovery of transcription factor binding sites**. *Nature Biotechnology* 2005, **23**:137-144.

10. GD Stormo: **DNA Binding Sites: Representation and Discovery.** *Bioinformatics* 2000, **16**:16-23.
11. CT Workman, GD Stormo: **ANN-Spec: a method for discovering transcription factor binding sites with improved specificity**. *Pac Symp Biocomput* 2000:467-78.
12. TD Schneider, GD Stormo, L Gold, A Ehrenfeucht: **Information content of binding sites on nucleotide sequences**. *Journal of Molecular Biology* 1986, **188**:415-431.
13. T Schneider, R Stephens: **Sequence logos: a new way to display consensus sequences**. *Nucl. Acids Res.* 1990, **18**:6097-6100.
14. MC Frith, MC Li, Z Weng: **Cluster-Buster: Finding Dense Clusters of Motifs in DNA Sequences**. *Nucleic Acids Research* 2003, **31**:3666-3668.
15. BP Berman, Y Nibu, BD Pfeiffer, P Tomancak, SE Celniker, M Levine, GM Rubin, MB Eisen: **Exploiting transcription factor binding site clustering to identify cis-regulatory modules involved in pattern formation in the Drosophila genome**. *PNAS* 2002, **99**:757-762.
16. D Dinakarpandian, V Raheja, S Mehta, E Schuetz, P Rogan: **Tandem machine learning for the identification of genes regulated by transcription factors**. *BMC Bioinformatics* 2005, **6**:204.
17. M Rebeiz, NL Reeves, JW Posakony: **SCORE: A computational approach to the identification of cis-regulatory modules and target genes in whole-genome sequence data**. *PNAS* 2002, **99**:9888-9893.
18. T Jaakola, M Diekhans, D Haussler: **Using the Fisher kernel method to detect remote protein homologies**. *Proc Int Conf INtell Syst Mol Biol* 1999:149-58.
19. Hua: **A novel method of protein secondary structure prediction with high segment overlap measure:support vector machine approach**. *Journal of Molecular Biology* 2001, **308**:397-407.
20. Hua., Sun.: **Support vector machine approach for protein subcellular localization prediction**. *Bioinformatics* 2001, **18**:721-728.
21. A Zien, G Ratsch, S Mika, B Scholkopf, T Lengauer, K-R Muller: **Engineering support vector machine kernels that recognize translation initiation sites**. *Bioinformatics* 2000, **16**:799-807.
22. M Wang, J Yang, K-C Chou: **Using string kernel to predict signal peptide cleavage site based on subsite coupling model**. *Amino Acids* 2005, **28**:395-402.
23. TS Furey, N Cristianini, N Duffy, DW Bednarski, M Schummer, D Haussler: **Support vector machine classification and validation of cancer tissue samples using microarray expression data**. *Bioinformatics* 2000, **16**:906-914.
24. P Pavlidis, WS Noble: **Gene Functional Classification from Heterogeneous Data**. *RECOMB Conference Proceedings* 2001:249-255.
25. D Holloway, M Kon, C DeLisi: **submitted: Machine Learning Methods for Transcription Data Integration**. *IBM Journal of Research and Development on Systems Biology* 2006.
26. N Simonis, SJ Wodak, GN Cohen, J van Helden: **Combining pattern discovery and discriminant analysis to predict gene co-regulation**. *Bioinformatics* 2004, **20**:2370-2379.
27. MA Beer, S Tavazoie: **Predicting Gene Expression from Sequence**. *Cell* 2004, **117**:185-198.

28. J Qian, J Lin, NM Luscombe, H Yu, M Gerstein: **Prediction of regulatory networks: genome-wide identification of transcription factor targets from gene expression data**. *Bioinformatics* 2003, **19**:1917-1926.
29. G Lanckriet, N Cristianini, M Jordan, WS Noble: **A Statistical Framework for Genomic Data Fustion**. *Bioinformatics* 2004, **20**:2626-2635.
30. B Sholkopf, AJ Smola: **Learning with Kernels**. *MIT Press* 2002.
31. P-N Tan, M Steinbach, V Kumar: **Introduction to Data Mining**. *Publisher:Pearson Education* 2005.
32. JC Platt: **Probabilisitic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods**. *Microsoft Research* 1999.
33. D Holloway, M Kon, C DeLisi: **submitted: Machine Learning and Data Combination for Regulatory Pathway Prediction**. *Synthetic and Systems Biology* 2006.
34. RA Flickinger: **Transcriptional frequency and cell determination**. *Journal of Theoretical Biology* 2005, **232**:151-156.
35. CJ Benham: **Energetics of the strand separation transition in superhelical DNA**. *Journal of Molecular Biology* 1992, **225**:835-847.
36. CJ Benham: **Duplex Destabilization in Superhelical DNA is Predicted to Occur at Specific Transcriptional Regulatory Regions**. *Journal of Molecular Biology* 1996, **255**:425-434.
37. NA Kent, SM Eibert, J Mellor: **Cbf1p Is Required for Chromatin Remodeling at Promoter-proximal CACGTG Motifs in Yeast**. *J. Biol. Chem.* 2004, **279**:27116-27123.
38. F Gao, B Foat, H Bussemaker: **Defining transcriptional networks through integrative modeling of mRNA expression and transcription factor binding data**. *BMC Bioinformatics* 2004, **5**:31.
39. DE Martin, A Soulard, MN Hall: **TOR Regulates Ribosomal Protein Gene Expression via PKA and the Forkhead Transcription Factor FHL1**. *Cell* 2004, **119**:969-979.
40. X-F Zheng, SL Schreiber: **Target of rapamycin proteins and their kinase activities are required for meiosis**. *PNAS* 1997, **94**:3070-3075.
41. A Hinnebusch: **General and Pathway-specific Regulatory Mechanisms Controlling the Synthesis of Amino Acid Biosynthetic Enzymes in Saccharomyces cerevisiae**. *The Molecular and Cellular Biology of the Yeast Saccharomyces: Gene Expression* 1992:319-414.
42. AG Hinnebusch, K Natarajan: **Gcn4p, a Master Regulator of Gene Expression, Is Controlled at Multiple Levels by Diverse Signals of Starvation and Stress**. *Eukaryotic Cell* 2002, **1**:22-32.
43. H Mountain, A Bytrom, C Korch: **The general amino acid control regulates MET4, which encodes a methionine-pathway-specific transcriptional activator of Saccharomyces cerevisiae**. *Molecular microbiology* 1993, **9**:221-223.
44. S Mangan, A Zaslaver, U Alon: **The Coherent Feedforward Loop Serves as a Sign-sensitive Delay Element in Transcription Networks**. *Journal of Molecular Biology* 2003, **334**:197-204.
45. B Pina, J Fernandez-Larrea, N Garcia-Reyero, F Idrissi: **The different (sur)faces of Rap1p**. *Molecular Genetic Genomics* 2003, **268**:791-798.

46. SJ Deminoff, GM Santangelo: **Rap1p Requires Gcr1p and Gcr2p Homodimers to Activate Ribosomal Protein and Glycolytic Genes, Respectively**. *Genetics* 2001, **158**:133-143.
47. G Zubay: **Biochemistry, Fourth Edition**. 1996:297-335.
48. C Harbison, E Fraenkel, R Young, et al: **Transcriptional Regulatory Code of a Eukaryotic Genome**. *Nature* 2004, **431**:99-104.
49. IT Lee, et al: **Transcriptional Regulatory Networks in Saccharomyces cerevisiae**. *Science* 2002, **298**:799-804.
50. V Matys, et al.: **TRANSFAC: Transciptional Regulation, from Patterns to Profiles**. *Nucleic Acids Research* 2003, **31**:374-378.
51. M Kellis, et al: **http://www.broad.mit.edu/annotation/fungi/comp_yeasts/**. 2003.
52. J van Helden: **Regulatory sequence analysis tools**. *Nucleic Acids Research* 2003, **31**:3593-3596.
53. B Ewan, et al.: **An Overvie of Ensembl**. *Genome Research* 2004, **14**:925-928.
54. RL Tatusov, DJ Lipman: **dust**. In: *Book dust* (Editor ed.^eds.). City.
55. A Smit, P Green: **Repeatmasker**.
56. S Aerts, G Thijs, B Coessens, M Staes, Y Moreau, B De Moor: **Toucan:Deciphering the Cis-Regulatory Logic of Coregulated Genes**. *Nucleic Acids Research* 2003, **31**:1753-1764.
57. C Harbison, E Fraenkel, R Young, et al: **http://jura.wi.mit.edu/fraenkel/download/release_v24/final_set/Final_InTableS2_v24.motifs**.
58. P Pavlidis, I Wapinski, WS Noble: **Support vector machine classification on the web**. *Bioinformatics* 2004, **20**:586-587.
59. J Ihmels, S Bergman, N Barkai: **http://barkai-serv.weizmann.ac.il/GroupPage/ .**
60. T Mathworks: **MATLAB: MATrix LABoratory**. *http://www.mathworks.com/*.
61. J Weston, A Elisseeff, G Bakir, F Sinz, et al: **SPIDER: object oriented machine learning library**. *http://www.kyb.tuebingen.mpg.de/bs/people/spider/*.
62. KR Christie, S Weng, R Balakrishnan, MC Costanzo, K Dolinski, SS Dwight, SR Engel, B Feierbach, DG Fisk, JE Hirschman, et al: **Saccharomyces Genome Database (SGD) provides tools to identify and analyze sequences from Saccharomyces cerevisiae and related sequences from other organisms**. *Nucl. Acids Res.* 2004, **32**:D311-314.
63. DL Wheeler, T Barrett, DA Benson, SH Bryant, K Canese, DM Church, M DiCuccio, R Edgar, S Federhen, W Helmberg, et al: **Database resources of the National Center for Biotechnology Information**. *Nucl. Acids Res.* 2005, **33**:D39-45.
64. PF Cliften, M Johnston, et al.: **Finding Functional Features in Saccharomyces Genomes by Phylogenetic Footprinting**. *Science* 2003, **301**:71-76.
65. M Kellis, N Patterson, M Endrizzi, B Birren, ES Lander: **Sequencing and Comparison of Yeast Species to Identify Genes and Regulatory Elements**. *Nature* 2003, **423**:241-254.
66. PF Cliften, et al.: **http://www.genetics.wustl.edu/saccharomycesgenomes/**. 2003.

67. C Leslie, E Eskin, WS Noble: **The Spectrum Kernel: A string kernel for SVM protein classification**. *Proceedings of the Pacific Symposium on Biocomputing* 2002:564-575.
68. C Leslie, R Kuang: **Fast kernels for inexact string matching**. *submitted for publication* 2005.
69. CS Leslie, E Eskin, A Cohen, J Weston, WS Noble: **Mismatch string kernels for discriminative protein classification**. *Bioinformatics* 2004, **20**:467-476.
70. D Allocco, I Kohane, A Butte: **Quantifying the Relationship Between Co-expression, Co-regulation, and Gene Function**. *BMC Bioinformatics* 2004, **5**.
71. C Bishop: **Neural Networks for Pattern Recognition**. *Oxford University Press* 1995.
72. J Wu, S Kasif, C DeLisi: **Identification of Functional Links Between Genes Using Phylogenetic Profiles**. *Bioinformatics* 2003, **19**:1-7.
73. A Gasch, A Moses, D Chiang, H Fraser, M Berardini, M Eisen: **Conservation and Evolution of Cis-Regulatory Systems in Ascomycete Fungi**. *PLOS Biology* 2004, **2**:2202-2219.
74. H Yu, N Luscombe, J Qian, M Gerstein: **Genomic Analysis fo Gene Expression Relationships in Transcriptional Regulatory Networks**. *Trends in Genetics* 2003, **19**:422-427.
75. J Ihmels, N Barkai, et al: **Revealing Modular Organization in the Yeast Transcriptional Network**. *Nature Genetics* 2002, **31**:370-377.
76. J Ihmels, S Bergman, N Barkai: **Defining Transcription Modules Using Large-Scale Gene Expression Data**. *Bioinformatics* 2004, **20**:1993-2003.
77. J Mellor, C DeLisi: **Inferring the Logic of Context-Dependent Transcription Regulation**. *Bioinformatics* 2004, **In Press**.
78. S Bergman, J Ihmels, N Barkai: **Iterative Signature Algorithm for the Analysis of Large-Scale Gene Expression Data**. *Physical Review* 2003, **67**.
79. KM Masters, KM Parkhurst, MA Daugherty, LJ Parkhurst: **Native Human TATA-binding Protein Simultaneously Binds and Bends Promoter DNA without a Slow Isomerization Step or TFIIB Requirement**. *J. Biol. Chem.* 2003, **278**:31685-31690.
80. T Acton, H Zhong, A Vershon: **DNA-binding specificity of Mcm1: operator mutations that alter DNA- bending and transcriptional activities by a MADS box protein**. *Mol. Cell. Biol.* 1997, **17**:1881-1889.
81. D Goodsell, R Dickerson: **Bending and curvature calculations in B-DNA**. *Nucl. Acids Res.* 1994, **22**:5497-5503.
82. S Satchwell, H Drew, A Travers: **Sequence periodicities in chicken nucleosome core DNA**. *Journal of Molecular Biology* 1986, **191**:659-675.
83. E Snitkin, A Gustafson, C DeLisi: *Unpublished work Personal Communication*.
84. B Balasubramanian, WK Pogozelski, TD Tullius: **DNA strand breaking by the hydroxyl radical is governed by the accessible surface areas of the hydrogen atoms of the DNA backbone**. *PNAS* 1998, **95**:9738-9743.
85. S Parker, J Greenbaum, G Benson, TD Tullius: **Structure-Based DNA Sequence Alignment**. *poster: 5th International Workshop in Bioinformatics and Systems Biology* 2005.
86. TD Tullius, JA Greenbaum: **Mapping nucleic acid structure by hydroxyl radical cleavage**. *Current Opinion in Chemical Biology* 2005, **9**:127-134.

87. D Cora, F Di Cunto, P Provero, L Silengo, M Caselle: **Computational identification of transcription factor binding sites by functional analysis of sets of genes sharing overrep-resented upstream motifs**. *BMC Bioinformatics* 2004, **5**.
88. J van Helden, J Collado-Vides: **Extracting Regulatory Sites from the Upstream Region of Yeast Genes by Computational Analysis of Oligonucleotide Frequencies**. *Journal of Molecular Biology* 1998, **281**:827-842.
89. P Haverty, U Hansen, Z Weng: **Computational Inference of Transcriptional Regulatory Networks from Expression Profiling and Transcription Factor Binding Site Identification**. *Nucleic Acids Research* 2004, **32**:179-188.
90. J van Helden: **Metrics for comparing regulatory sequences on the basis of pattern counts**. *Bioinformatics* 2004, **20**:399-406.
91. CH Choi, G Kalosakas, KO Rasmussen, M Hiromura, AR Bishop, A Usheva: **DNA dynamically directs its own transcription initiation**. *Nucl. Acids Res.* 2004, **32**:1584-1590.
92. P Rice, I Longden, A Bleasby: **EMBOSS: The European Molecular Biology Open Software Suite**. *Trends in Genetics* 2000, **16**:276-277.
93. KJ Breslauer, R Frank, H Blocker, LA Marky: **Predicting DNA Duplex Stability from the Base Sequence**. *PNAS* 1986, **83**:3746-3750.
94. F Baldino: **High-resolution in situ hybridization histochemistry**. *Methods in Enzymology* 1989, **168**:761-777.