Transcription factor binding site identification using the Self-Organizing Map

Bioinformatics Advance Access published January 12, 2005

Transcription factor binding site identification using the Self-Organizing Map

Shaun Mahony[1]*, David Hendrix[2,4,5], Aaron Golden[1,3], Terry J. Smith[1], Daniel S. Rokhsar[4,5]

[1]*National Centre for Biomedical Engineering Science, NUI Galway, Galway, Ireland*
[2]*Department of Physics, University of California, Berkeley, CA 94720, USA*
[3]*Department of Information Technology, NUI Galway, Galway, Ireland*
[4]*Joint Genome Institute, Walnut Creek, CA 94598, USA*
[5]*Center for Integrative Genomics, University of California, Berkeley, CA 94720, USA*

Running title: Self-Organizing Map for motif discovery

***To whom correspondence should be addressed. Contact details:**
Email: shaun.mahony@nuigalway.ie
Phone: +353-86-8765617 or +353-91-524411 ext. 3849
Fax: +353-91-750596

ABSTRACT

Motivation: The automatic identification of over-represented motifs present in a collection of sequences continues to be a challenging problem in computational biology. In this paper, we propose a self-organizing map of position weight matrices as an alternative method for motif discovery. The advantage of this approach is that it can be used to simultaneously characterize every feature present in the data set, thus lessening the chance that weaker signals will be missed. Features identified are ranked in terms of over-representation relative to a background model.

Results: We present an implementation of this approach, named SOMBRERO, which is capable of discovering multiple distinct motifs present in a single data set. Demonstrated here are the advantages of our approach on various data sets and SOMBRERO's improved performance over two popular motif-finding programs; MEME and AlignACE.

Availability: SOMBRERO is available free of charge from http://bioinf.nuigalway.ie/sombrero.

Contact: shaun.mahony@nuigalway.ie

INTRODUCTION

Every eukaryotic genome sequenced thus far has shown vast quantities of DNA which do not appear to contain protein-coding regions. Although such non-coding DNA can play important structural roles, much of it also harbors intricate gene regulatory information, including short (6-20 bp) motifs that serve as transcription factor binding sites. Cracking the so-called "*cis*-regulatory code" has become an important goal in the decoding of genomic data, and an integral part of this challenge is the identification of transcription factor binding sites.

A common computational approach to finding transcription factor binding sites involves identifying sequence motifs which are shared between genes that are known to be co-regulated, followed by an attempt to distinguish functional binding sites from other motifs. Motifs are often identified using probabilistic models, and this can be achieved through the application of standard statistical learning theory methods, such as maximum likelihood estimation (e.g. MEME (Bailey and Elkan, 1994)) or Gibbs sampling (e.g. AlignACE (Hughes *et al.*, 2000) and BioProspector (Liu *et al.*, 2001)).

Many alternative motif identification methods have also been proposed, including word enumeration, winnowing, and dictionary construction based methods (Bussemaker *et al.*, 2000; Gupta and Liu, 2003; Pevzner and Sze, 2000; Rigoutsos and Floratos, 1998; Sinha and Tompa, 2002).

In this work, we present a new approach to the identification of conserved motifs in biological sequences based on a self-organizing map (SOM) of position weight matrices (PWMs). The SOM (Kohonen, 1995) is a competitive learning network, which seeks to characterize the distribution of vectors in input space and represent it as a lattice of feature vectors. The SOM has been previously applied to such problems in biological sequence analysis as the study of codon usage (Abe *et al.*, 2003; Kanaya *et al.*, 2001; Mahony *et al.*, 2004; Wang *et al.*, 2001) and the clustering of similar protein sequences (Kohonen and Somervuo, 2002; Yang and Chou, 2003).

The current study seeks to use the clustering properties of the SOM to exhaustively characterize all motif features present within a set of input sequences. This is achieved through the iterative updating of PWMs at each node in the SOM neural network. As described in the Methods below, the training process clusters similar subsequences at each node, allowing each PWM to become a representation of a given motif in the input sequences. In order to identify motifs that have a high probability of having a functional regulatory role, over-represented motifs are found by examining each node in comparison with a background probabilistic model.

We hypothesise that simultaneously characterizing all motif features may allow weaker motifs to be more easily distinguished in noisy datasets. Our approach is demonstrated here using an implementation named SOMBRERO (Self-Organizing Map for Biological Regulatory Element Recognition and Ordering). SOMBRERO is a command-line driven software program written in C++, with a Perl-Tk interface for viewing results (see Fig. 1). SOMBRERO's motif-finding performance is evaluated here in comparison with two popular motif-finding algorithms, MEME and AlignACE.

SYSTEM AND METHODS

Our goal is to characterize all the features of a given length ℓ that are present in the input sequences while still affording some flexibility to allow for more variability than a simple consensus sequence. This is accomplished by mapping the input

sequences onto a 2-D Self-Organizing Map of probabilistic models of sequence motifs. Hence, we represent a motif as a position weight matrix (PWM), $W_{ib} = \log\left(\frac{f_{ib}}{p_b}\right)$, where f_{ib} is an entry of a profile matrix, i.e. the probability that the base b occurs in position i of the motif. Thus, $b \in \{A, C, G, T\}$ and $i \in \{1,2,..., \ell\}$. The probabilities p_b represent the background frequency of the base b occurring in the intergenic regions of the genome under investigation. A score function *S(x)* will be used to determine how similar a string x is to a motif defined by a profile matrix. The log-likelihood ratio of a DNA string being an instance of the motif provides such a score and is computed by:

$$S(x) = \sum_{b=A}^{T} \sum_{i=1}^{\ell} x_{ib} W_{ib} \qquad (1)$$

where x_{ib}, known as the indicator matrix for the string x, is 1 if base b is at position i of the string and 0 otherwise. A high score *S(x)* indicates that the string x is more similar to the motif characterized by the profile matrix f_{ib} than the background model.

SOM Architecture

The general structure of a self-organizing map is a two dimensional network of interconnected nodes, whose algorithmic approach to learning involves the mapping of input vectors representing some feature or pattern onto specific nodes over the training phase. Our novel application of the two-dimensional SOM uses a PWM at each node to represent features present in the input sequences. During the training phase, each PWM evolves to portray a different feature of the data. Nodes that are located close to one another on the network will strongly influence each other's PWM, and this enforces the similarity of neighboring nodes. Arranging the nodes on a grid allows the PWMs to spread out over the entire input space in an ordered and topological way.

To this end, we create an M x N grid of such nodes, and denote the coordinates of the nodes by $\mathbf{z} = (z_1, z_2)$. Each node represents a feature of the input sequence, and contains the number of occurrences $n_{\mathbf{z}}$ of that feature, as well as the number of occurrences of base b at position i in the instances of that feature, denoted $c_{ib}^{\mathbf{z}}$. Clearly, $\sum_b c_{ib}^{\mathbf{z}} = n_{\mathbf{z}}$, for all i. Each node also keeps track of a profile matrix $f_{ib}^{\mathbf{z}}$ that is used to define the PWM. In typical applications, the profile matrix is defined as

$$f_{ib}^{z} = \frac{c_{ib}^{z} + \beta p_b}{n_z + \beta} \qquad (2)$$

In order to avoid zero probabilities, an unbiased pseudo-count βp_b is added to each term, in proportion to the background frequency p_b, and scaled down by the factor $\beta \approx 0.1$. The value of β has little effect on the outcome and serves only to make the pseudo-count small. This definition is slightly altered below in order to make the definition more amenable to the SOM architecture. A PWM $W_{ib}^{z} = \log\left(\frac{f_{ib}^{z}}{p_b}\right)$ is also defined for each node. Likewise, a log-likelihood ratio score $S_{\mathbf{z}}(x)$ can be computed with Eq(1) for each string x and each node $\mathbf{z}$.

Initializing the SOM

During the initialization of a SOM, each node's PWM must be set to represent a unique value. By virtue of the smoothing and ordering effect of the SOM algorithm, the initialization of the PWMs can be carried out randomly without significantly affecting the motif-finding accuracy of the SOM (see Supplementary Table 1).

For SOMBRERO, a more structured approach to PWM initialization is favored. After the motif length ℓ is chosen, typically from 8 to 20, the profile matrices $f_{ib}^{\mathbf{z}}$ are initialized to a value between 0.1 and 0.4 such that the sum of all probabilities in any given position of the motif is 1. We also constrain the nodes to have a degree of preference for a certain base, as determined by the quadrant of the SOM which the node lies in. For example, the top left corner of the SOM is given a preference for 'A', the bottom left corner a preference for 'T', the top right corner a preference for 'C', and the remaining bottom right corner a preference for 'G'. All nodes between the corners have gradients of preference.

The pre-ordering of the SOM lattice allows training to converge smoothly and rapidly on features of the input data set, and also allows a known ordering to be maintained throughout training. That is, after training on the data set, motifs in the upper left corner still maintain a bias toward 'A', etc. The conservation of the grid topology should increase ease of use and user familiarity with the results. It should be stressed, however, that the choice of PWM initialization parameters has no algorithmic significance. In tests carried out using random initial conditions, we observe no substantial decrease in accuracy, even though the final ordering of nodes on the SOM is quite different (Supplementary Table 1).

Training the SOM

SOM training is carried out via the so-called batch version of the SOM, in contrast to a slower incremental regression process. For the initial training iteration, the input sequences are segmented into every overlapping string x^j of length ℓ, and each string is then loaded and assigned to the node with the maximum likelihood, i.e. the highest score $S_{\mathbf{z}}(x^j)$. Unlike the EM algorithm, which probabilistically weights each string as being generated from a profile matrix, SOMBRERO makes hard associations. The process continues until each string is likewise processed so that the raw base-count matrix $c_{ib}^{\mathbf{z}}$ is updated for the winning node for each string.

In addition to the initial training on the input sequences, there is also a neighborhood update step where the base counts at each node contribute fractionally to the PWMs of all other nodes. Nodes that are located close to one another contribute more to each other than distal nodes, enforcing the similarity of nearby nodes. This is achieved computationally by augmenting the profile matrix $f_{ib}^{\mathbf{z}}$ as defined in Eq(2), to include contributions from other nodes. More precisely,

$$f_{ib}^{z} = \frac{\sum_{z'} \Phi\left(\left|z - z'\right|\right) c_{ib}^{z'} + \beta p_b}{\sum_{b'} \sum_{z'} \Phi\left(\left|z - z'\right|\right) c_{ib}^{z'} + \beta}, \tag{3}$$

where $\Phi(|\mathbf{z} - \mathbf{z}'|)$ is a neighborhood function that defines the proportion that a node will contribute to another node that is a distance $|\mathbf{z} - \mathbf{z}'|$ away on the SOM. For our purposes, the Gaussian neighborhood function

$$\Phi\left(\left|z - z'\right|\right) \equiv e^{-\frac{\left|z-z'\right|^2}{\gamma}} = e^{-\frac{[(z_1-z_1')^2+(z_2-z_2')^2]}{\gamma}} \tag{4}$$

works quite well. Here the term γ in the exponential is a measure of the sharpness of the neighborhood function and is defined as $\gamma \equiv \frac{1}{\log(\delta)}$ so that adjacent nodes will contribute $1/\delta$ of their counts to each other. In practice, δ ranges from 4 to 15 in the course of the training. Thus, the contributions to $f_{ib}^{\mathbf{z}}$ from the counts of neighboring nodes initially strongly enforce the similarity of nearby nodes, and end up contributing little at the end of training.

This new profile matrix $f_{ib}^{\mathbf{z}}$ is used to then characterize the feature associated with the node $\mathbf{z}$. It will be used to define the PWM $W_{ib}^{\mathbf{z}}$, and the training cycle begins anew with this updated PWM. This procedure continues until a convergence criterion is met, or until a specified number of training iterations are completed.

Once SOM training is complete, each string x^j is assigned to its most similar node. In the case where two or more strings at a given node are overlapping strings in the

input sequences, only the most similar string to the node's PWM, i.e. the string with the larger $S_{\mathbf{z}}(x^j)$, is kept. After this deletion, $n_{\mathbf{z}}$ will refer to the remaining number of strings at the node **z**.

Ranking the PWMs and identifying potential regulatory motifs

At the conclusion of the training phase, the two dimensional grid of PWMs will represent a characterization of the various motif features present in the input sequences. In other words, each node in the SOM will contain a different motif present in the input sequences. Since the focus of this study is on finding potential regulatory elements, the aim is to distinguish those motifs that are over-represented in relation to a background model. In practice, many of the motifs characterized by the SOM may not be over-represented, despite numerous occurrences in the input sequences, and are in fact due to the random occurrence of similar strings in the data set. One would like to rank the various motifs found by the SOM in terms of over-representation against a stochastic model of occurrence.

The z-score is used here for the purpose of ranking the motifs. To this end, a data set of random sequences of the same number and lengths as the input data set is created using a Markov chain based model of the intergenic DNA in the genome being studied. Each string x^j of length ℓ in this random set is assigned to the most similar node in the SOM in the same manner as the input sequence strings. Once again, overlapping instances are resolved. This process is repeated for 100 random data sets, and in this way we find the expected number of occurrences $< n_{\mathbf{z}} >$ of each node's profile matrix as well as the standard deviation $\sigma_{\mathbf{z}}$ at each node. The statistical significance value of each node's profile matrix is then calculated using the z-score:

$$Z_{score} = \frac{n_z - < n_z >}{\sigma_z} \qquad (5)$$

In order to justify the suitability of z-scores in the context of ranking the SOM's PWMs, we consider the following. Under the simplest conditions, where only one strand is considered and the overlapping instances of each node are not deleted, the distribution of z-scores is approximately Gaussian. This is understood from the fact that n_z can be viewed as the sum of many similar binary random variables associated with each ℓ-mer in the data set. Inasmuch as the data set is large, the values of n_z will have a Gaussian distribution by the central limit theorem. This interpretation is demonstrated in Figure 2, which shows the empirical distribution of z-scores

computed from a 60x30 SOM trained on a random data set consisting of 20 sequences of 1000bp each, plot alongside the Gaussian distribution with the same mean and variance. In this calculation a purely random background model with $p_b = 0.25$ for all bases b was used to both generate the input sequences and their randomized counterparts used in computing the z-score. Considering both strands and deleting overlaps does not significantly distort the distribution. Though the mean of this distribution shown in Figure 2 is essentially 0, the standard deviation is greater than 1 (about 1.7 in this data set), indicating that in practice there is more variability in the n_z values than σ_z. Nevertheless, the z-score can be interpreted as Gaussian, and is thus a practical measure of statistical significance when separating the over-represented motifs from those due to chance.

As the focus of this study is to identify regulatory elements, certain repetitive of 'simple' genomic features (such as poly-A sequences or repeats) may be disregarded. For this purpose, a complexity filter is employed, and a suitable complexity score for a profile matrix is developed. The complexity score, which is a natural extension of a common single-string score (Wan *et al.*, 2003), is given by

$$C(z) = \left(\frac{1}{4}\right)^{\ell} \prod_{b=A}^{T} \left(\frac{\ell}{\sum_{i=1}^{\ell} f_{ib}^{z}} \right)^{\sum_{i=1}^{\ell} f_{ib}^{z}} \qquad (6)$$

Any nodes which receive less than a reasonably low complexity score (0.15 is used in this study), are discounted from being treated as a possible functional motif. It should be noted that the use of a complexity filter does not interfere with SOMBRERO's ability to find regulatory motifs that contain sequential runs of a single nucleotide. For example, the *hb* motif in *Drosophila* contains 6 sequential 'T's, but yet the *hb* motif identified by SOMBRERO is not filtered by the complexity measure (see the "Performance in *Drosophila* regulatory regions" section below).

Finally, before presenting the output to the user, a re-sampling step is carried out; where every string in the input sequences that receives a higher similarity score at a given node than that of the node's lowest scoring hard-clustered string is also counted as an occurrence of the motif.

RESULTS AND DISCUSSION

Performance in artificial sequences

In real biological motif detection we often do not know (a) how many (if any) occurrences of a motif exist in each sequence, (b) how many independent motifs there are in the sequences, or (c) the length of such motifs. Our approach can handle such ambiguities automatically. The SOM algorithm allows the simultaneous recognition of multiple motifs through the process of training multiple nodes. There are also no assumptions made by our method as to the number of instances of a motif per sequence, so cases where a motif does not occur in some input data sequences, but often in the others, can be handled. SOMBRERO also allows the iteration of SOM training through various motif lengths and tabulates the results, allowing motifs of unknown length to be found.

We summarize here a demonstration of SOMBRERO's performance when finding motifs in artificial sequence sets that contain multiple motifs. The test sets used in this study each consist of 10 data sets, and each data set contains 10 sequences (generated from a third order Markov model of *E. coli* intergenic regions). Each sequence harbors a random number (0 to 3) of occurrences of each of the motifs gcn4, gal4 and mat1 (generated from TRANSFAC weight matrices). Three such test sets are generated, the three test sets being distinguishable by the total sequence length of each component data set (4500bp, 8000bp or 12500bp).

SOMBRERO is trained using three separate SOM sizes for each test set. Each SOM is trained for 100 cycles and checking both sequence orientations, and as we assume that SOMBRERO has no prior knowledge of the lengths of each motif, training is allowed to repeat over all even lengths from 8 to 18 (inclusive). The model that is used to generate the sequences is also used as SOMBRERO's background model.

Table 1 summarizes the average performance of each SOM size (in terms of the best match to each motif from the top 20 SOMBRERO results) in each test set. As Table 1 demonstrates, the optimum SOM performance is achieved by keeping a ratio in the order of ten input dataset base pairs for every node SOM. In other words, the SOM size should be 'scaled-up' for bigger input data sets. This is necessary because a small SOM trained on a large dataset leads to overcrowding at the nodes, and thus performance (in particular specificity) is heavily reduced. Conversely, a large SOM trained on a small dataset leads to the nodes becoming too specialized, explaining the

poorer performance rates in such cases. The 10:1 heuristic ratio is used to define suitable SOM sizes for the remainder of this study.

Table 1 also demonstrates that SOMBRERO can effectively find multiple motifs of unknown length in a dataset where the number of instances of each motif varies across the sequences. This is an indicator of SOMBRERO's performance in real genomic data.

Performance in *S. cerevisiae* promoter regions

In this section, we compare the performance of SOMBRERO with two popular motif identification programs, MEME and AlignACE, on real biological data sets with experimentally verified motif locations. The test set used is a collection of 10 yeast genomic sequence sets taken from the *Promoter Database of Saccharomyces cerevisiae* (SCPD; http://cgsigma.cshl.org/jian/), the selection of sequence sets based on suitable sequence length and also on there being at least a total of four motif instances in each set. Each sequence set consists of between 3 and 19 promoter sequences, where each sequence is at least 500 bp long and contains at least one instance of a particular motif as specified by the name of the data set. Each sequence may also potentially contain other motifs and signals, which makes the motif identification task more difficult than in the artificial sequence problem.

SOMBRERO is run with a SOM size chosen to roughly keep the 10:1 ratio described earlier. In each case the SOM is trained for 100 cycles, and each SOM is trained on multiple ℓ-mer lengths across a window of at least 6 bp (for example, in the case of the abf1 sequence set, the SOM is trained on ℓ-mers from length 10 to length 16 inclusive). The background model used is a third order Markov model taken from all yeast intergenic regions. Both sequence orientations are checked by each SOM.

MEME is run using the following command: "meme $infile -dna -mod tcm -revcomp -nmotifs 10 -minw $min -maxw $max", where $min and $max are replaced by the same values used by SOMBRERO. This command allows MEME to search both strands for up to 10 motifs, each of which can occur zero or more times in each sequence. These settings are as close as possible to those of SOMBRERO. It may seem that the fair comparison would allow MEME to generate the same number of motifs as there are nodes on the SOM. However, MEME masks predictions as it progresses to prevent from finding a motif multiple times. Therefore, the value of the '-nmotifs' parameter has no effect on the order of the motifs presented in the results.

Since we are only interested here in finding the top 10 results from each program, we do not need to have MEME find more than 10 motifs.

AlignACE is run online (http://atlas.med.harvard.edu/cgi-bin/alignace.pl) with default arguments, which are nearly identical to those of MEME and SOMBRERO, except the online version of AlignACE does not take a minimum or maximum motif length. Because of this, AlignACE is afforded the slight advantage of being given the optimum motif length for finding each respective motif.

Figure 3 compares each of the known motifs in the data set to those found by SOMBRERO. Table 2 shows the results of the comparative study, showing false negative (FN) and false positive (FP) rates, given with respect to the best matching pattern found in the top 10 results returned by each program. In terms of FN rates, SOMBRERO performs better than or equal to the other methods in 9 of the 10 cases. In those four cases where SOMBRERO and AlignACE have equal FN rates, SOMBRERO has the best overall performance (i.e. lower FP rate) in two data sets. A higher FP rate in some cases seems to be the price paid for the improvement in FN rates. SOMBRERO predicts the known sites more completely, at the expense of introducing a few more false predictions and possibly finding new uncharacterized sites. False positives are defined here as predicted occurrences of a motif that do not appear in the relevant annotation (in this case taken from the SCPD). However, many so-called false positive sites may in fact bind the transcription factor *in vivo*, and therefore the FP estimates presented here may not be especially accurate or useful.

It should be noted that by the very nature of differing input parameter formats and differences in the underlying algorithms, no comparison between programs can be completely fair. Every program will have certain advantages over others, and therefore the MEME and AlignACE results demonstrated here should serve only as a frame of reference.

Performance in *Drosophila* regulatory regions

Berman, *et al.* (Berman *et al.*, 2002) describe a set of 19 regulatory regions of 9 *Drosophila* genes that harbor binding sites for the transcription factors *Bicoid* (*bcd*), *Caudal* (*cad*), *Hunchback* (*hb*), *Knirps* (*kni*) and *Krüppel* (*Kr*). The total sequence length of this dataset is 22,535bp, making it useful for evaluating the performance of motif-finders in large datasets. Indeed, the dataset has been previously used to test the accuracy of the LOGOS motif-finder (Xing *et al.*, 2003). Figure 4 shows the PWMs

postulated by Berman, *et al.* for the five motifs, based on biologically identified binding sites.

SOMBRERO is run on the Berman, *et al.* dataset using a SOM size of 50x25 nodes trained for 100 cycles, and trained for all ℓ-mer lengths between 8 and 20bp. In this example, a third order Markov model of all *Drosophila* intergenic regions is used as the background model. MEME and AlignACE are also run as before, except this time using commands that allow each method to return up to twenty motif predictions. The top 20 predictions from each of SOMBRERO, MEME and AlignACE are compared to the known binding sites in the Berman, *et al.* dataset, and the best matching predictions for each of the five known motifs are shown in terms of FN and FP in Table 3.

No method satisfactorily finds the *kni* binding sites, but of the other four motifs, SOMBRERO gives the best performance in three cases (*bcd*, *cad* and *hb*). Eight of the motifs found by SOMBRERO are shown in Figure 5. Motifs 1 to 5 in Figure 5 correspond to the best matching motifs found by SOMBRERO for the known binding sites of *bcd*, *cad*, *hb*, *kni* and *Kr*, respectively.

Motifs 6, 7 and 8 are additional putative motifs predicted by SOMBRERO. However, motif 8 is very similar to the TRANSFAC recorded PWM for the binding sites of the *Tramtrack* (*ttk*) protein. *Tramtrack* has been shown to be a repressor of many of the genes in the Berman, *et al.* dataset (Brown and Wu, 1993), so it is quite likely that SOMBRERO is correctly predicting the occurrences of *ttk* binding sites that play a functional role in the repression of these genes. As *Tramtrack* is not one of the transcription factors whose binding sites are annotated in the Berman, *et al.* dataset, the identification of *ttk* binding sites demonstrates SOMBRERO's potential for finding novel regulatory motifs in real genomic data.

CONCLUSION

This paper explores a new approach to motif finding based on a self-organizing map of position weight matrices, and a software implementation named SOMBRERO is evaluated. It should be noted that our approach is quite different to probabilistic motif-finders in several respects. A crucial distinction between the SOMBRERO algorithm and a motif-finder based on EM, for example, is that EM probabilistically weights each subsequence as being generated by a model, while SOMBRERO makes

hard associations via a clustering procedure. Therefore, SOMBRERO is more similar to a k-means clustering than to the EM algorithm. However, a comparison between SOMBRERO and a similar k-means based algorithm (using the same number of cluster centres as SOM nodes) shows that SOMBRERO performs significantly better (see Supplementary Table 1). The results suggest that the neighbourhood update function (Eq. 3), the distinguishing factor between the SOM and the k-means algorithm, plays an important role in optimizing motif models. In fact, the generality introduced into the motif models by the neighbourhood update function in the early stages of training may explain the improved motif detection over EM-based methods.

Another key distinction between the SOMBRERO algorithm and EM is that the SOM allows all motifs in a dataset to be simultaneously characterized. For the purposes of finding regulatory elements, the motif features of the input sequences can be subsequently ranked in terms of over-representation in relation to a background model. Our approach is thus useful for finding multiple regulatory elements in a dataset, as no limit on the number of motifs to find is employed. Simultaneously characterizing all features in an input dataset may also help to separate weak motif signals from large or noisy datasets. Indeed, this may explain SOMBRERO's improved performance in finding known instances of binding sites over MEME and AlignACE in the large Berman, *et al. Drosophila* dataset. Neither MEME nor AlignACE allows the simultaneous characterization of multiple motifs, relying instead on an iterative process of single motif detection and subsequent screening of that motif's occurrences.

We have demonstrated the improved performance of SOMBRERO over other popular methods in real motif identification problems. However, such improvements in performance come with a computational time cost; our algorithm has a running time of $O(L(MN) + (MN))$ when an M x N SOM is applied to a data set of total length L. Therefore, the application of SOMBRERO to larger data sets (where larger SOMs are also necessary) is currently computationally costly. Nevertheless, the SOM algorithm outlined in the Methods section is highly parallelizable, and future work will include the implementation of a parallelized version of SOMBRERO.

Recent research has shown that motif finders are more effective at finding weak motifs in larger data sets when certain information that is external to the sequence data is incorporated into the method's probabilistic models. Examples of this approach have recently included the use of information regarding the tendency of

binding sites to cluster together in eukaryotic promoter regions (Berman *et al.*, 2002), and also the apparent constraint imposed by the structure of the binding proteins on the conservation pattern of some motifs; the so-called "shape bias" (Xing *et al.*, 2003). Future improvements to the SOM-based motif finder could incorporate similar probabilistic models in the algorithm, thus allowing the application of the method to larger promoter sequence sets, especially those yielded by eukaryotic gene expression experiments.

ACKNOWLEDGEMENTS

The authors wish to thank Pavel Tomancak and Ben Berman for providing the *Drosophila* datasets and for their help in interpreting the SOMBRERO results. The anonymous reviewers are also thanked for their valuable comments. S.M. thanks the Irish Research Council for Science, Engineering and Technology, and the NUI, Galway - University of California EAP program for supporting this work.

REFERENCES

Abe, T., Kanaya, S., Kinouchi, M., Ichiba, Y., Kozuki, T., and Ikemura, T. (2003) Informatics for unveiling hidden genome signatures. *Genome Res*, **13,** 693-702.

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

Berman, B.P., Nibu, Y., Pfeiffer, B.D., Tomancak, P., Celniker, S.E., Levine, M., Rubin, G.M., and Eisen, M.B. (2002) Exploiting transcription factor binding site clustering to identify cis-regulatory modules involved in pattern formation in the Drosophila genome. *Proc Natl Acad Sci U S A*, **99,** 757-762.

Brown, J.L. and Wu, C. (1993) Repression of Drosophila pair-rule segmentation genes by ectopic expression of tramtrack. *Development*, **117,** 45-58.

Bussemaker, H.J., Li, H., and Siggia, E.D. (2000) Building a dictionary for genomes: identification of presumptive regulatory sites by statistical analysis. *Proc Natl Acad Sci U S A*, **97,** 10096-10100.

Gupta, M. and Liu, J.S. (2003) Discovery of conserved sequence patterns using a stochastic dictionary model. *Journal of the American Statistical Association*, **98,** 55-66.

Hughes, J.D., Estep, P.W., Tavazoie, S., and Church, G.M. (2000) Computational identification of cis-regulatory elements associated with groups of functionally related genes in Saccharomyces cerevisiae. *J Mol Biol*, **296,** 1205-1214.

Kanaya, S., Kinouchi, M., Abe, T., Kudo, Y., Yamada, Y., Nishi, T., Mori, H., and Ikemura, T. (2001) Analysis of codon usage diversity of bacterial genes with a self-organizing map (SOM): characterization of horizontally transferred genes with emphasis on the E. coli O157 genome. *Gene*, **276,** 89-99.

Kohonen, T. (1995) *Self-Organizing Maps*. Springer-Verlag, Berlin.

Kohonen, T. and Somervuo, P. (2002) How to make large self-organizing maps for nonvectorial data. *Neural Netw*, **15,** 945-952.

Liu, X., Brutlag, D.L., and Liu, J.S. (2001) BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. *Pac Symp Biocomput***,** 127-138.

Mahony, S., McInerney, J.O., Smith, T.J., and Golden, A. (2004) Gene prediction using the Self-Organizing Map: automatic generation of multiple gene models. *BMC Bioinformatics*, **5,** 23.

Pevzner, P.A. and Sze, S.H. (2000) Combinatorial approaches to finding subtle signals in DNA sequences. *Proc Int Conf Intell Syst Mol Biol*, **8,** 269-278.

Rigoutsos, I. and Floratos, A. (1998) Combinatorial pattern discovery in biological sequences: The TEIRESIAS algorithm. *Bioinformatics*, **14,** 55-67.

Sinha, S. and Tompa, M. (2002) Discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic Acids Res*, **30,** 5549-5560.
Wan, H., Li, L., Federhen, S., and Wootton, J.C. (2003) Discovering simple regions in biological sequences associated with scoring schemes. *J Comput Biol*, **10,** 171-185.
Wang, H.C., Badger, J., Kearney, P., and Li, M. (2001) Analysis of codon usage patterns of bacterial genomes using the self-organizing map. *Mol Biol Evol*, **18,** 792-800.
Xing, E.P., Wu, W., Jordan, M.I., and Karp, R.M. (2003) LOGOS: a modular Bayesian model for de novo motif detection. *Proc IEEE-CSB***,** 266-276.
Yang, Z.R. and Chou, K.C. (2003) Mining biological data using self-organizing map. *J Chem Inf Comput Sci*, **43,** 1748-1753.

FIGURES

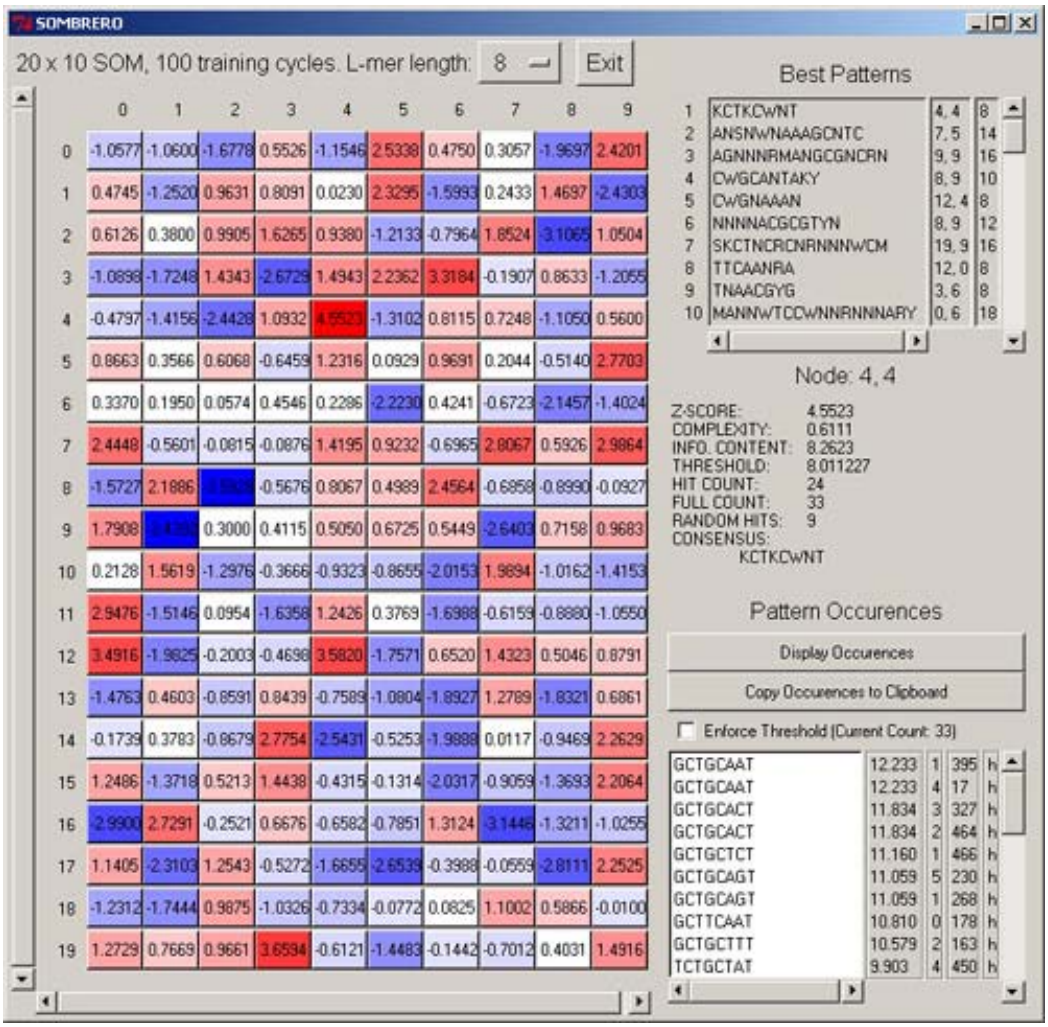

Figure 1. The SOMBRERO results viewer. In this example, SOMBRERO has been trained on genomic sequences from *S. cerevisiae* that contain binding sites for *mcb*. Separate SOMs were trained for each even sub-sequence length from 8 to 18, and each SOM is accessible from the results viewer. The SOM shown here is a 20x10 node SOM trained using length 8 subsequences. In the software display, the SOM nodes are color-coded according to the z-score of the motif contained in the node. A list of the most significant motifs across all trained SOMs is displayed in the top right hand corner of the results viewer. Information can be displayed for any motif, including a display of each instance of the motif on the input sequences.

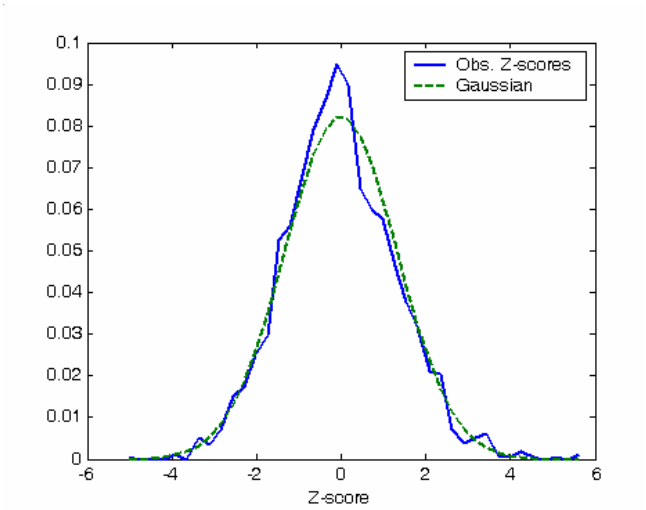

Figure 2. An empirical distribution of motif z-scores computed from random data. A 60x30 SOM was trained on a data set of 20 purely random sequences of 1000bp each. The distribution of the z-scores for the nodes of the SOM is plot alongside a Gaussian distribution with the same mean and variance. This demonstrates that the z-scores can be properly viewed as Gaussian random variables when discerning significant nodes from insignificant ones.

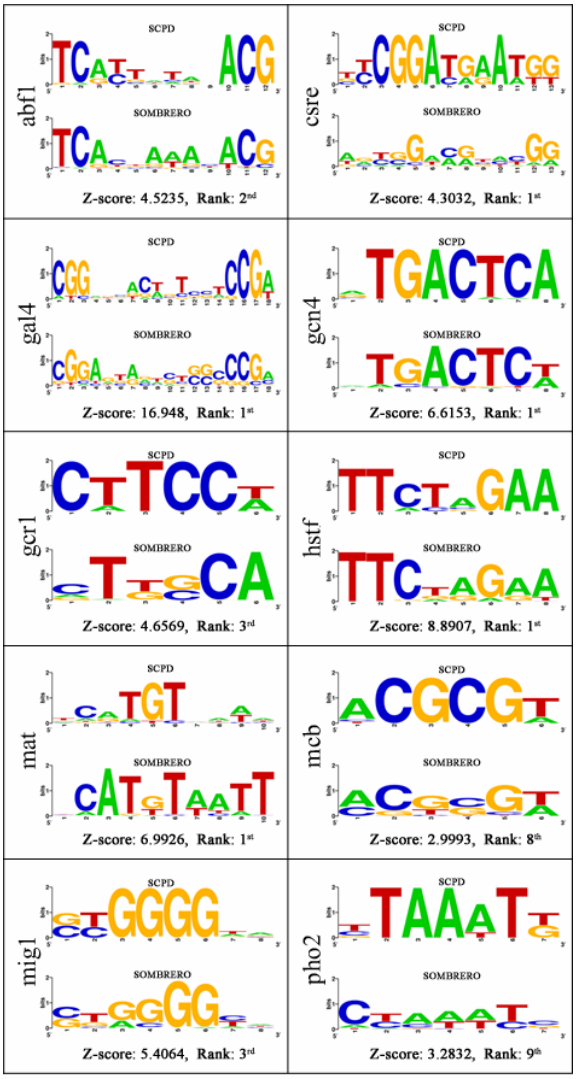

Figure 3. A comparison of known logos to predicted logos of yeast enhancers. The known logos were constructed using *Promoter Database of Saccharomyces cerevisiae* (SCPD) probability matrices and the predicted logos were produced using the motifs found by SOMBRERO. Shown under the SOMBRERO motifs are the z-scores and the rank given to that motif by SOMBRERO.

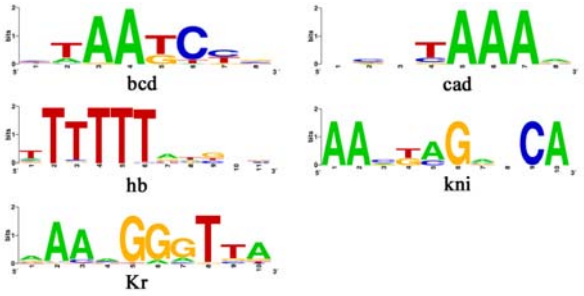

Figure 4. Motif patterns derived by Berman, *et al.* (Berman *et al.*, 2002) from multi-alignments of biologically identified protein binding elements.

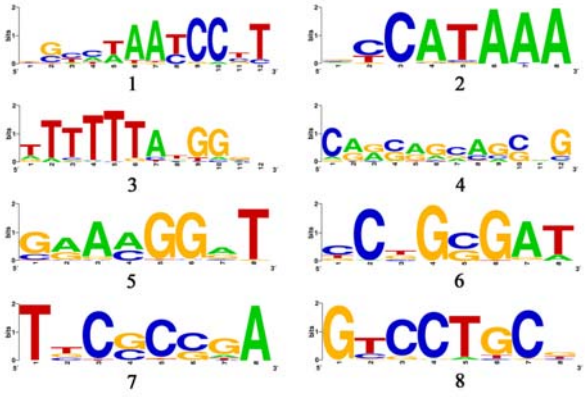

Figure 5. Eight of the top twenty motifs found by SOMBRERO in the Berman, *et al.* (Berman *et al.*, 2002) dataset of 19 *Drosophila* regulatory sequences.

TABLES

		30x15 SOM			40x20 SOM			50x25 SOM		
		FN	FP	Perf.	FN	FP	Perf.	FN	FP	Perf.
Length 12500bp datasets	***gcn4***	0.541	0.684	0.230	0.558	0.607	0.263	0.472	0.458	0.365
	gal4	0.098	0.322	0.631	0.087	0.254	0.696	0.173	0.311	0.603
	mat1	0.264	0.610	0.342	0.363	0.554	0.356	0.329	0.446	0.436
			Avg. Perf.	0.401		*Avg. Perf.*	0.438		*Avg. Perf.*	**0.468**
Length 8000bp datasets	***gcn4***	0.388	0.522	0.367	0.339	0.396	0.461	0.346	0.381	0.466
	gal4	0.082	0.277	0.680	0.090	0.346	0.614	0.223	0.322	0.568
	mat1	0.248	0.468	0.453	0.177	0.298	0.610	0.154	0.322	0.604
			Avg. Perf.	0.500		*Avg. Perf.*	**0.562**		*Avg. Perf.*	0.546
Length 4500bp datasets	***gcn4***	0.254	0.488	0.436	0.261	0.337	0.537	0.458	0.377	0.408
	gal4	0.180	0.270	0.629	0.173	0.266	0.636	0.153	0.208	0.693
	mat1	0.048	0.188	0.780	0.252	0.267	0.588	0.181	0.160	0.708
			Avg. Perf.	**0.615**		*Avg. Perf.*	0.587		*Avg. Perf.*	0.603

Table 1. The simultaneous discovery of three different motifs in simulated datasets. Average false negative (FN), false positive (FP) and performance coefficients (Perf.) are shown for each SOM size in each of the three test sets. The performance coefficient is defined as $|K \cap P| / |K \cup P|$, where K is the set of known motif sites and P is the set of predicted motif sites (Pevzner and Sze, 2000). The best average performance (Avg. Perf.) across all three motifs in each test set is highlighted in **bold**. This table demonstrates that the best average performance can be achieved using the heuristic 10:1 input base pairs to SOM nodes ratio.

				SOMBRERO				MEME			AlignACE		
Name	seq	sites	bp	SOM	NP	FN	FP	NP	FN	FP	NP	FN	FP
abf1	19	20	8600	40x20	25	**0.450**	0.560	11	0.550	0.182	16	0.500	0.375
csre	4	4	2550	20x10	11	**0.250**	0.727	6	0.500	0.667	17	**0.250**	0.824
gal4	4	14	3100	20x10	17	**0.071**	0.235	12	0.286	0.167	12	0.214	0.083
gcn4	9	25	4500	30x15	14	**0.600**	0.286	10	0.920	0.800	18	**0.600**	0.444
gcr1	6	9	3350	30x15	29	**0.222**	0.690	9	0.444	0.444	16	0.333	0.625
hstf	6	9	3400	30x15	21	**0.111**	0.571	24	0.333	0.750	18	**0.111**	0.556
mat	7	13	3500	30x15	12	0.308	0.250	15	**0.154**	0.267	9	0.308	0.000
mcb	6	12	3150	20x10	31	**0.083**	0.645	12	0.250	0.250	12	**0.083**	0.083
mig1	9	10	4500	30x15	25	**0.200**	0.680	0	1.000	1.000	11	0.900	0.909
pho2	3	6	2350	20x10	33	**0.500**	0.909	0	1.000	1.000	0	1.000	1.000

Table 2. Comparison of motif detectors on 10 yeast promoter sequence datasets. For each method, the number of predictions (NP), false negative (FN), and false positive (FP) rates are shown. The best FN rate in each dataset is highlighted in **bold**.

		SOMBRERO		MEME		AlignACE	
	sites	FN	FP	FN	FP	FN	FP
bcd	23	**0.57**	0.80	0.87	0.93	0.78	0.83
cad	63	**0.43**	0.46	0.75	0.43	0.78	0.67
hb	119	**0.35**	0.40	0.82	0.21	0.77	0.37
kni	24	**0.76**	0.94	0.88	0.82	0.88	0.93
Kr	61	0.61	0.59	0.64	0.46	**0.52**	0.25

Table 3. Comparison of motif detectors on 19 Drosophila regulatory sequences that contain instances of 5 regulatory binding sites. The best FN rate for each motif is highlighted in **bold**.