

Detecting Gene Regulation Relations from Microarray Time Series Data

N. Malhis

Department of computer science
Kent State University
Kent, OH 44240 USA
Phone: (330) 389-0963
Fax: (330) 672-7824
Email: nmalhis@cs.kent.edu

A. Ruttan

Department of computer science
Kent State University
Kent, OH 44240 USA
Phone: (330) 672-9066
Fax: (330) 672-7824
Email: ruttan@cs.kent.edu

Abstract - *Microarrays are important tools in the quest to map the gene regulation networks of cells. A common use of microarrays result in time series pairs that indicates how the output of one gene affects another. Substantial efforts have been made towards identifying pairs of microarray time series that indicate that one gene is a regulator for another. However, most approaches make assumptions about the behavior of the regulation relation time series pairs and then attempt to identify microarray time series pairs with similar profiles. Such approaches are only partially successful and frequently cannot identify microarray times series pairs that are known to be regulation pairs. In this work, we present a new machine learning approach utilizing a set of Hidden Markov Model, HMMs, for scoring temporal relationships between gene expressions that uses a training set of known regulation relation that avoids the need to assume a specific regulation profile.*

Keywords: Bioinformatics, Gene regulatory Networks, Datamining, Machine learning, Hidden Markov Model.

1 Introduction

Recent attempts to map the gene regulation networks of different species of plants and animals using microarrays has made it imperative to develop data mining tools to find evidence of regulation relationships in the resulting data. Microarray data can be thought of as an array of n sequences, one for each gene, of the expression levels (i.e. amounts) of a protein produced by a gene. Each row in the array represents one time series of the m expression levels values of one gene. Each column of the array gives all n expression levels for the genes at a specific time. A regulation relation is relationship between genes that is indicated by a change of expression levels. Genes are called activators if they activate the production of proteins of other genes, called targets. Genes are repressors if they suppress the production of target genes. In microarray data, a target gene's expression level increases—but the increase may not be instantaneous—in

response to an increase of its activator's expression level, and decreases in response to its activator's expression level decreases. The repressor has an opposite effect: target gene's expression level decreases in response to the increase of expression level for its repressor, and increases in response to the decreases of its repressors expression level. In most cases, regulation relations are relationships that are switched on and off by different types of chemical and biological signals. We refer to the behavior of the pair when the relationship is switched on as a temporal episode profile or just a temporal profile.

2 Background

Many different approaches have been used for detecting regulating relations in microarray time series data, each of them uses a different technique for extracting the regulation information from the data. The authors of [01] focused on modeling Bayesian networks based on the conditional dependency between expression levels of different genes. Two genes represent a regulation relation if there was an edge from the regulator to the target in the Bayesian network. The authors of [02] use Pearson's correlation coefficient between two gene profiles. In [03] the correlation over dominant spectrum component is used.

In [04]-[06] different assumptions were made about the nature of a temporal episode profile of an activator-target pair and a repressor target pair. Pairs of rows from the microarray matrix were then tested for adherence to those assumptions. While in [07]-[09] similar assumptions were made for triplets of an activator, a repressor and a target (*ART*), and in a similar way, triplets of rows from the microarray matrix are tested for adherence to those assumptions. However, in [04]-[09] no attempt was made to take into account the transient natures of the temporal profiles.

New approaches use heterogeneous sources of data simultaneously to produced improved predictions. In [10] microarray data and a set of binding sites patterns (or motif

matrix) is used as input for a machine learning approach based on an alternative decision tree ADT (a decision tree with boosting), in which a set of previously known regulation relations is used for training. While [11] uses expression time series data, a motif matrix, and information on the direct physical interaction between a regulator and the upstream regions of its target, a ChiP-chip matrix, and correlation between gene time series to score the time series data.

3 Problem description

In this paper we use a machine learning approach that utilizes a set of hidden Markov Models (HMMs) to detect temporal regulation relations in microarray time series data. Unlike methods presented in [04]-[09], no assumed profiles are used. Instead, a set of known regulation relationships is used to train a collection of HMMs. Later, these trained HMMs are to be used to score each pair of the gene expression sequences. We consider a set of n genes $G = \{g_0, g_1, g_2, \dots, g_{n-1}\}$, and a set M of m microarray experiments taken for those n genes at fixed intervals, where for each t in T , the set of time points of M , and for any two integers $k, l \in [1, m-1]$, $t_k - t_{k-1} = t_l - t_{l-1}$. In other words, we consider sequences of experiments where the time interval between observations is always the same. Each microarray experiment measures the expression levels of all n genes simultaneously. The result is a microarray time series array $D = \{d_{ij}\}$ with $i \in [0, n-1]$ and $j \in [0, m-1]$, and each entry d_{ij} represents the expression level of gene g_i at time t_j .

We now establish some notation to describe our algorithm. Let R be the set of all $(n * n - n)$ distinct ordered pairs of G . An *expression relation time sequence* is the sequence generated by combining two different genes time sequences into a sequence of ordered pairs. That is, each entry of the new sequences is an ordered pair values taken from corresponding entries of the two original gene time sequences. An *expression relation array* Z is the array of all possible expression relation time sequences that can be generated from a microarray expression data D .

Example 1: Suppose G is a set of $n = 3$ genes over $m = 5$ time points, and the microarray data D looks like.

D		M				
		T_0	T_1	t_2	t_3	T_4
G	g_0	0.1	0.11	0.78	0.81	0.5
	g_1	0.3	0.51	0.01	0.9	0.79
	g_2	0.88	0.76	0.48	0.13	0.06

The expression relation array Z that is generated from this data is:

Z		M				
		t_0	t_1	t_2	t_3	t_4
R	g_0, g_1	0.1, 0.3	0.11, 0.51	0.78, 0.01	0.81, 0.9	0.5, 0.79
	g_0, g_2	0.1, 0.88	0.11, 0.76	0.78, 0.48	0.81, 0.13	0.5, 0.06
	g_1, g_0	0.3, 0.1	0.51, 0.11	0.01, 0.78	0.9, 0.81	0.79, 0.5
	g_1, g_2	0.3, 0.88	0.51, 0.76	0.01, 0.48	0.9, 0.13	0.79, 0.06
	g_2, g_0	0.88, 0.1	0.76, 0.11	0.48, 0.78	0.13, 0.81	0.06, 0.5
	g_2, g_1	0.88, 0.3	0.76, 0.51	0.48, 0.1	0.13, 0.9	0.06, 0.79

4 Problem Definition

The basic problem we consider in this paper is: *Given expression relation arrays Z_A and Z_R that are known from outside sources, e.g. the biological literature, to contain respectively activator and repressor expression relation time sequences, but for which the precise temporal profiles are not known, and another expression array Z_T , where Z_T contains data from genes pairs not found in Z_A or Z_R , characterize the temporal profiles in such a manner so as to permit the identification of expressions times sequences in Z_T that contain temporal profiles similar to the ones in Z_A or Z_R .*

Since microarray data in general contains high level of noise, and for many cases it also under sampled, one needs to considers ways to minimize the error that is caused by the noise and amplified by the under sampling of the data. Additionally, since we are considering expression relation time sequences constructed from biological data, it is possible that the sets Z_A and Z_R are contaminated by temporal episodes that are not related to the regulation mechanisms known to be expressed in that data. Provided that the extraneous temporal episodes occur simultaneously in Z_A and Z_R can we avoid identifying expression time sequences in Z_T that contain only extraneous temporal episodes?

In the remainder of this paper, we describe an algorithm that shows significant promise of identifying sequences with activator or repressor temporal profiles in noisy under sampled data and that reduces the likelihood of identifying sequences that only contain extraneous data.

5 Input Data Analysis

Our microarray data is taken from two sets that are publicly available:

- Four separate microarray time sequence array $D[4]$ for *saccharomyces cerevisiae* (baker yeast) cell cycle time expression data [12]. These time sequences are: CDC15, 24 time points, Alpha, 18 time points, CDC28, 17 time points, and Elu Elutriation. 14 time points. Those 4 Microarray time series are publicly available at <http://genome-www.stanford.edu/cellcycle/>. This data is used to form a set Z_T , discussed below, of data with unknown regulation profiles.

2. A set of 1018 known regulation relations was extracted from the yeast protein database www.ypd.com. 728 pairs are known activators-target pairs that are used to form a set Z_A . 290 are repressor-target pairs that make up a set Z_R .

On counting the number of local maxima and local minima (a local maxima is considered at a time point t when its value is higher than the values of its both neighbors at point $(t-1)$ and $(t+1)$, and local minima is when its value is lower than its neighbors) in all four of these time series, it was found that the percentage of local maxima plus minima points was from 57% to 71% of the total number of points in the four data sets (only those time point that have two neighbors are considered). This very high percentage of local extreme indicates to us that this data is under sampled, at least for our purposes.

Table 1: Counting local extremes in the input datasets.

Dataset	Total	Local Maxima	Local Minima
Cdc15	0.71	0.35	0.35
Alpha	0.66	0.33	0.33
cdc28	0.57	0.28	0.28
Elu	0.57	0.27	0.29

Also as there can be unaccounted-for biological activity occurring in yeast cells, any such activity that affects the expression level data will appear as severe background noise. Such background activity together with random measurement errors yields a high level of apparently random activity that causes microarray data to appear to be quite noisy.

6 Model Description

In the section, we describe an adaptive HMM method that builds on the properties of both of HMMs and their training algorithms. All HMM training algorithms (in particular Baum-Welch) are iterative algorithms that attempt to maximize the probability of the training sequence given the HMM [13], but in most cases HMM training algorithms lead only to local maxima. In most problems of interest, the optimization surface is very complex and has a large number of local maxima [13]. Therefore, the final outcome of training is usually a local maxima that depends strongly on the initial state of the model. A given starting state will find one local maxima of the optimization surface, and thereby select one feature of the training data. A different starting state will like find a different local maxima, and in turn select a different feature of the training data. This suggests that *a number of HHMs, each started with a different random initial state and trained on with the same data, will in the aggregate produce a good description of the training data*. One can aggregate the results of many HHMs either by averaging the scores or taking the maximum of all the HHMs scores.

If reasoning above is accurate, it should be the case that *the procedure described above, when applied to two distinct training sets that contain a common pattern, should yield two distinct sets of HHMs both of which, when aggregated separately, give high scores to any sequence containing that pattern*. This property can usefully be applied to minimize the likelihood of selecting extraneous temporal episodes whenever one has two sets of data that are known to contain different temporal patterns, but may both contain the same extraneous temporal patterns.

For instance in the case of expression relation data Z_A and Z_R discussed above, this observation could be used by making the reasonable assumption that activator-target patterns are significantly different than repressor-target patterns. Then, if one produces two distinct sets of random-initial-states HHMs, one is trained on Z_A and one on Z_R , then the difference of the scores on the two aggregations (the score from Z_A 's set of HHMs – the score from Z_R 's set of HHMs) applied to arbitrary sequence will be

1. Large and positive if the arbitrary sequences contain activator-target patterns.
2. Close to zero, either positive or negative if it contains extraneous patterns or no patterns similar to activator-target or repressor-target patterns.
3. Significantly negative if it contains repressor-target patterns.

Extraneous patterns score close to zero since subtracting the two scores will cancel their contribution from both Z_A 's and Z_R 's HMM set scores.

Formally the idea is to train two sets of HHMs, an $H_{activation}$ set containing N_A HHMs and an $H_{repression}$ set containing N_R HHMs using the training sets data Z_A and Z_R derived from the known activator-target and repressor-target data, described above, using a different random initial state for each HMM. If N_A and N_R are large enough, almost all of the frequent occurring patterns in the training sets will be selected by one or more HHMs. Then, for each expression relation time sequence S in the data Z_T , we calculate the probability its being generated by each of the HHMs in the $H_{activation}$ (as in a standard HMM application), and aggregate the results by averaging these probabilities. This average is called $A_{activation}$. The average $A_{repression}$ is calculated in the same way from $H_{repression}$. The score for sequence S is the computed as $S_{score} = A_{activation} - A_{repression}$.

7 The Algorithm

The complete algorithm can be divided in three major parts, data preparation, training HHMs, and scoring data.

7.1 Data preprocessing:

1. Normalization: each expression time sequence in D is normalized to the range $[0 \dots 1]$.
2. Discretization: each value d_{ij} is discretized into three possible values:
 - It is low (*labeled L*) if it was in the range $[0 \dots (0.5 - \alpha)]$.
 - It is in the midrange (*labeled A*) if it was in the range $[(0.5 - \alpha) \dots (0.5 + \alpha)]$.
 - It is high (*labeled H*) if it was in the range $[(0.5 + \alpha) \dots 1]$.

Here the threshold α is set to be large enough to reflect the uncertainty that is due to measurement noise and small enough to prevent the real value from been masked.

3. Generating the expression relation array: for each possible pair of genes, expression relation time sequence is generated from the discretized data as an ordered pair (Activator, Target) or (Regulator, Target), and the arrays Z_A and Z_R are constructed.

Example 2: From Example 1, if α is set at 0.15, then the discretized microarray data D will look like this:

D		M				
		t_0	t_1	t_2	t_3	t_4
G	g_0	L	L	H	H	A
	g_1	L	A	L	H	H
	g_2	H	H	A	L	L

And the discretized expression relation array Z will be:

Z		M				
		t_0	t_1	t_2	t_3	t_4
R	g_0, g_1	L, L	L, A	H, L	H, H	A, H
	g_0, g_2	L, H	L, H	H, A	H, L	A, L
	g_1, g_0	L, L	A, L	L, H	H, H	H, A
	g_1, g_2	L, H	A, H	L, A	H, L	H, L
	g_2, g_0	H, L	H, L	A, H	L, H	L, A
	g_2, g_1	H, A	H, A	A, L	L, H	L, H

7.2 HMMs Details:

The nine pairs (L,L), (L,A), (L,H) ... (H,H) along with two special break symbol d_s and d_e (described below) form the 11 observation symbols used in our HMMs. There are 18 state in our HMMs. That number was chosen to be roughly comparable to the size of microarray time series sequences in our data.

A set with N_A HMMs $H_{\text{activation}}$ is trained using Z_A and a set with N_R HMMs $H_{\text{repression}}$ is trained using Z_R . The

training process is the same for all HMMs. First a training sequence is constructed from appropriate training set by forming the individual sequences in the set into one long sequence. As a regulation relation pattern cannot extend from one data sequence to another we insert two break symbol d_s and d_e between each sequence to reflect this fact. Likewise, a missing data value will result in an ill-defined pattern, so we broke it into two sequences by inserting the two break symbols at the missing data point in order to avoid identifying an erroneous pattern. Given that both of the resulting subsequences are longer than or equal to a predefined SeqMin value they are used, otherwise, they are discarded.

Following [13], we produce a randomly generated left-to-right HMM, such that the first state is a special state S_f that can produce only one observation symbol d_s and this symbol can only be generated from that state. The last state is also a special state S_l that can produce only one observation symbol d_e and this symbol can only be generated from this S_l state. Once a Markov chain leaves either state S_x , that state S_x can not be revisited at a later time. Finally, Baum-Welch is used repeatedly with the long training sequence to update the HMM.

8 Experiment and Results

Since the training data set is limited—only to 728 activation relation and 290 repression relations—to test the effectiveness of our algorithm, we perform a k-fold cross validation experiment with $k=10$. A sample Z_T of 3000 expression relation sequences, built from randomly select expression data with unknown regulation profiles, is used to test the overall classification effectiveness of our method by comparing the scores on Z_T with the test scores calculated using the 10-fold cross validation procedure.

Since the final score is the difference of a sequence's score on each of HMMs sets, the scores take on both positive and negative values in an interval [low, high]. Since the scores can be scaled arbitrarily, the significant aspect of a score is its position in that interval rather than its numerical value. In particular, sequences with the smallest scores are most likely to have repressor-target patterns while those with the highest scores are more likely to have activator-target patterns. To avoid possibly confusing references to the actual scores, we map the interval [low, high] to $[0, 99]$. With that scaling sequences near 0 are predicted to contain repressor-target patterns while those near 99 are expect to contain activator-target patterns.

The results of cross validation and random relation tests are shown in Figure 1. The results of these tests preliminarily validate the heuristic reasoning presented above. One sees in Figure 1 that scores of the activation relations are clearly higher than those of the repression

relations. Sequences with scores in the lower range (in this case from 1 to 25) are most likely repressors-target pairs. Sequences with higher scores (in this case from 53 to 99) are more likely to be activator-target pairs, while relations with scores in the midrange could represent either.

Considering the scores of the sequences from Z_T in Figure 1, we conclude that sequences with low scores are more likely to be repressor-target relations, while those with higher scores are more likely to be activator-target relations. This is more clearly illustrated in Figure 2,

Figure 1: Results of cross validation and random relation test. All scores were redistributed to the range [0.. 99], and a window size three equal weight was used for smoothing.

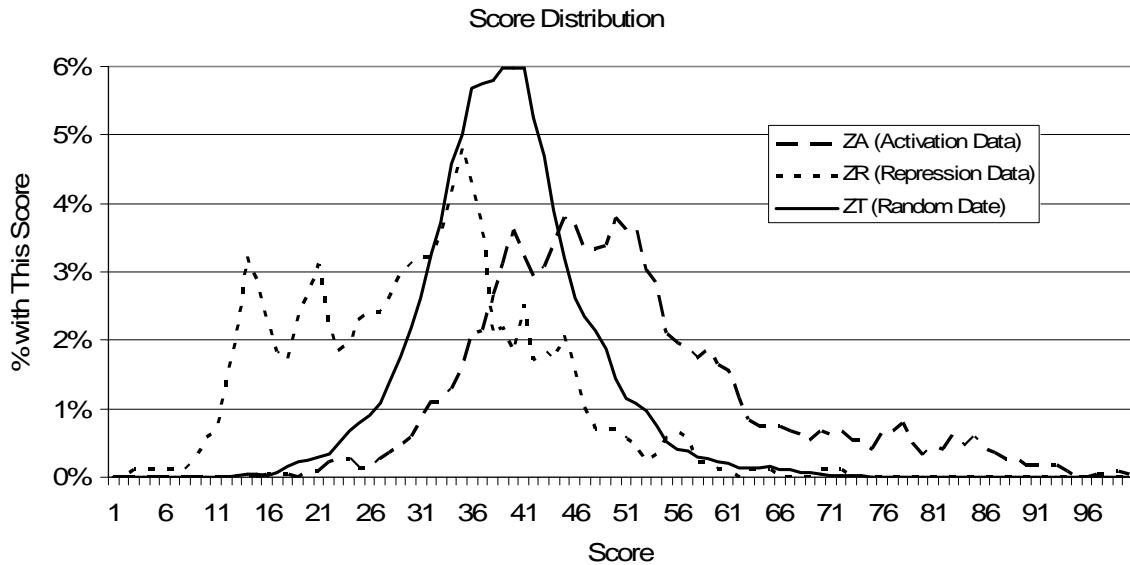
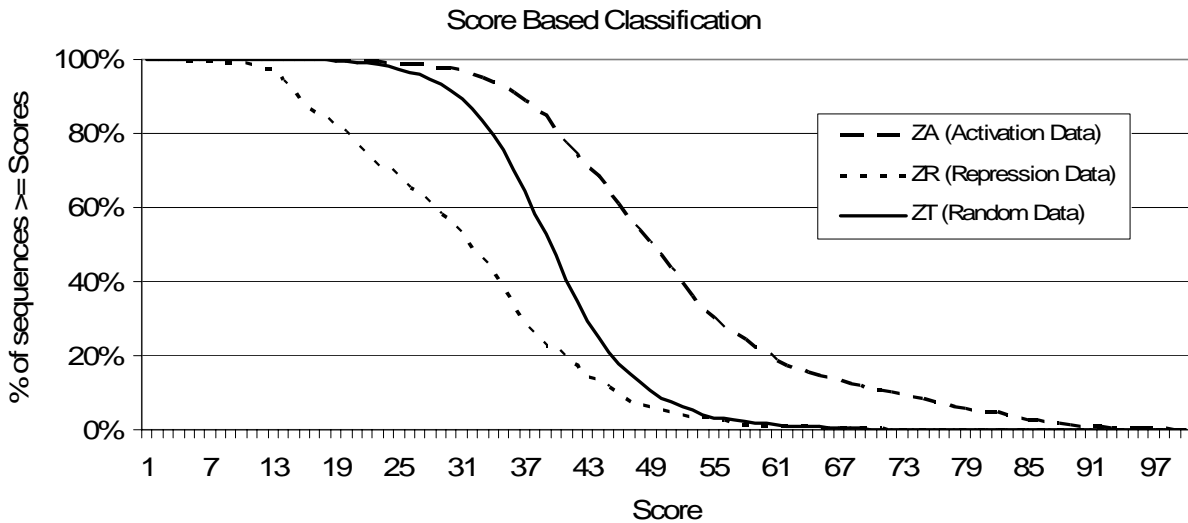


Figure 2: The points on this graph represents a pair (x, y) where y is the percentage of that scores from Figure 1 greater or equal to x.



9 Conclusions

We have presented in this paper a new approach based on HMMs for detecting temporal regulation relations in microarray time series data. Our algorithm appears to be able to accurately predict that high or low scoring sequences are likely to contain activator-target patterns or repressor target patterns respectively at least in the limited situation described above. A detailed theoretical analysis of this new approach is fourth coming.

We encouraged by the performance of this approach and we think that it can be extended in several different directions. In particular, the method developed herein can be combined, using for instance a support vector machine approach, with information obtained from other type of biological data such as a motif matrix and a ChIP-chip matrix to generate a final more accurate regulation prediction. A paper describing this approach is in preparation.

10 References

- [1] Nir Friedman and Michal Linial and Itzhak Nachman and Dana Pe'er, "Using Bayesian networks to analyze expression data", Proceedings of the Fourth Annual International Conference on Computational Molecular Biology, 2000 (RECOMB2000), pp. 127-135.
- [2] Horimoto, K., Aburatani, S., Kuhara, S., and Toh, H., "ASIAN - automatic system for inferring a network from gene expression profiles", *Cell Mol. Biol.*, 5:192-207, 2001.
- [3] Yeung, L.K., Kong, R., Liew, A.W.-C., Szeto, L.K., Yan, H. and Yang, M. "Measuring Correlation between Microarray Time-series Data using Dominant Spectrum Component". In Proc. Second Asia-Pacific Bioinformatics Conference (APBC2004), 2004, Dunedin, New Zealand. CRPIT, 29. Chen, Y.-P. P., Ed. ACS. 309-314.
- [4] Malhis, N., and Ruttan, A., "Predicting Gene Regulatory Networks from Micro Array Time Series Data by Elimination", Proceedings of the Second Annual BioTechnology and BioInformatics Symposium BIOT-05, October 20-21, 2005, Colorado Spring, Colorado, USA, pp. 37-42.
- [5] Hongquan Xu, Peiru Wu, C.F. Jeff Wu, Carl Tidwell and Yixin Wang, "A Smooth Response Surface Algorithm for Constructing Gene Regulatory Network", *Physiol Genomics*. 2002 Oct 2;11(1):11-20.
- [6] Woolf PJ, and Wang Y. "A fuzzy logic approach to analyzing gene expression data". *Physiol Genomics* 3: 9-15, 2000.
- [7] Kwon, A., Hoos, H. & Ng, R. "Inference of transcriptional regulation relationships from gene expression data", *Bioinformatics* 19, 2003, pp. 905-912.
- [8] Filkov, V., Skiena, S. & Zhi, J. "Analysis techniques for microarray time-series data", *J. Comput. Biol.* 9, 2002, pp. 317-330.
- [9] Chen, T., Filkov, V., Skiena, S. "Identifying Gene Regulatory Networks from Experimental Data", RECOMB 1999.
- [10] Manuel Middendorf, Anshul Kundaje, Chris Wiggins, Yoav Freund, and Christina Leslie "Predicting genetic regulatory response using classification", *Bioinformatics* 2004 20: i232-i240
- [11] T. D. Bie, P. Mousieus, K. Engelen, B. D. Moor, N. Cristianini, and K. Marchal, "Discovering Transcriptional Modules From Motif, ChIP-CHIP and Microarray data", The Pacific Symposium on Biocomputing (PSB) 2005.
- [12] Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown, P.O., Botstein, D., and Futcher, B. 1998. Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell* 9, 3273-3297.
- [13] L.R. Rabiner. "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", In Proceedings of the IEEE, 77(2):267--296, February 1989.