1. Introduction

Historically, biological experiments have concentrated on the transcriptional regulation of only a few genes, thus limiting inference of genetic networks to those occurring locally within the cell. With the advancement of DNA hybridization microarrays, however, biologists now have the ability to measure the abundance of thousands of mRNA transcripts simultaneously, providing a genome-wide perspective of the state of genetic transcripts. The immense size of the resulting data sets poses a significant challenge to researchers who wish to mine the gene expression data in search of genetic regulatory networks. Thus, algorithms developed to analyze these data sets must effectively handle thousands of expression profiles, and extract important information for relating and contrasting genes under differing conditions.

Several methods have been developed to construct genetic networks from gene expression data. These include Boolean networks [19,16], Bayesian networks [13,16,40,8], weight matrices [33], differential equations models [7,11], causal inference [37], and fuzzy logic models [36]. The primary difficulty in constructing networks from microarray data is the sparseness of samples (usually under a hundred) relative to the great number of genes (in the tens of thousands). Weight matrices and differential equations models involve the estimation of numerous network parameters, which is difficult with the relative scarcity of samples. Boolean networks and Bayesian networks simplify the data structure by converting continuous gene expression data into discrete data. However, this requires placing the expression data into one of two states, “on” or “off”, and this simplification ignores valuable information. Fuzzy logic represents a compromise between discrete and continuous models. Observations are categorized, but can maintain partial membership in more than one category. The hope is to allow a simplified rule structure, similar to Boolean networks, while retaining more of the information in the original, continuous data.

The analysis of gene expression microarrays plays an important role in elucidating the functionality of genes, including the discovery of genetic interactions that regulate gene expression. Several methods for modeling such gene regulatory networks exist, including a variety of continuous and discrete models. Methods based on fuzzy logic provide an interesting alternative. However, the guidelines for modeling gene expression with fuzzy logic are fairly open, and the need arises to investigate how adjustments in the modeling scheme will affect the results. In this work, we modify an existing fuzzy logic algorithm to involve an arbitrary number of classification states, and investigate the limiting behavior as the number of states tends to infinity. We also propose a probabilistic model as an alternative to the fuzzy logic model. We investigate the behavior of both models using yeast cell-cycle data and the simulated data of Werhli et al. [A.V. Werhli, M. Grzegorczyk, D. Husmeier, Comparative evaluation of reverse engineering gene regulatory networks with relevance networks, graphical Gaussian models, and Bayesian networks, Bioinformatics 22 (2006) 2523–2531]. We found that altering the number of classification states in both the fuzzy logic and probability models can influence which networks are predicted by both models. As the number of states tends to infinity, the predictions made by both models converge to those of a regression model. Models with a small to moderate number of classification states produced better results from a biological standpoint, compared to models with higher numbers of states. In simulated data, models with differing numbers of classification states produced similar overall results. Thus, increasing the complexity of the models has no apparent benefit, and models with smaller numbers of classification states are therefore preferred based on their ease of linguistic interpretation. The software used in this paper is freely available for non-commercial use at http://louisville.edu/~g0broc01.

The analysis of gene expression microarrays plays an important role in elucidating the functionality of genes, including the discovery of genetic interactions that regulate gene expression. Several methods for modeling such gene regulatory networks exist, including a variety of continuous and discrete models. Methods based on fuzzy logic provide an interesting alternative. However, the guidelines for modeling gene expression with fuzzy logic are fairly open, and the need arises to investigate how adjustments in the modeling scheme will affect the results. In this work, we modify an existing fuzzy logic algorithm to involve an arbitrary number of classification states, and investigate the limiting behavior as the number of states tends to infinity. We also propose a probabilistic model as an alternative to the fuzzy logic model. We investigate the behavior of both models using yeast cell-cycle data and the simulated data of Werhli et al. [A.V. Werhli, M. Grzegorczyk, D. Husmeier, Comparative evaluation of reverse engineering gene regulatory networks with relevance networks, graphical Gaussian models, and Bayesian networks, Bioinformatics 22 (2006) 2523–2531]. We found that altering the number of classification states in both the fuzzy logic and probability models can influence which networks are predicted by both models. As the number of states tends to infinity, the predictions made by both models converge to those of a regression model. Models with a small to moderate number of classification states produced better results from a biological standpoint, compared to models with higher numbers of states. In simulated data, models with differing numbers of classification states produced similar overall results. Thus, increasing the complexity of the models has no apparent benefit, and models with smaller numbers of classification states are therefore preferred based on their ease of linguistic interpretation. The software used in this paper is freely available for non-commercial use at http://louisville.edu/~g0broc01.

The analysis of gene expression microarrays plays an important role in elucidating the functionality of genes, including the discovery of genetic interactions that regulate gene expression. Several methods for modeling such gene regulatory networks exist, including a variety of continuous and discrete models. Methods based on fuzzy logic provide an interesting alternative. However, the guidelines for modeling gene expression with fuzzy logic are fairly open, and the need arises to investigate how adjustments in the modeling scheme will affect the results. In this work, we modify an existing fuzzy logic algorithm to involve an arbitrary number of classification states, and investigate the limiting behavior as the number of states tends to infinity. We also propose a probabilistic model as an alternative to the fuzzy logic model. We investigate the behavior of both models using yeast cell-cycle data and the simulated data of Werhli et al. [A.V. Werhli, M. Grzegorczyk, D. Husmeier, Comparative evaluation of reverse engineering gene regulatory networks with relevance networks, graphical Gaussian models, and Bayesian networks, Bioinformatics 22 (2006) 2523–2531]. We found that altering the number of classification states in both the fuzzy logic and probability models can influence which networks are predicted by both models. As the number of states tends to infinity, the predictions made by both models converge to those of a regression model. Models with a small to moderate number of classification states produced better results from a biological standpoint, compared to models with higher numbers of states. In simulated data, models with differing numbers of classification states produced similar overall results. Thus, increasing the complexity of the models has no apparent benefit, and models with smaller numbers of classification states are therefore preferred based on their ease of linguistic interpretation. The software used in this paper is freely available for non-commercial use at http://louisville.edu/~g0broc01.

The analysis of gene expression microarrays plays an important role in elucidating the functionality of genes, including the discovery of genetic interactions that regulate gene expression. Several methods for modeling such gene regulatory networks exist, including a variety of continuous and discrete models. Methods based on fuzzy logic provide an interesting alternative. However, the guidelines for modeling gene expression with fuzzy logic are fairly open, and the need arises to investigate how adjustments in the modeling scheme will affect the results. In this work, we modify an existing fuzzy logic algorithm to involve an arbitrary number of classification states, and investigate the limiting behavior as the number of states tends to infinity. We also propose a probabilistic model as an alternative to the fuzzy logic model. We investigate the behavior of both models using yeast cell-cycle data and the simulated data of Werhli et al. [A.V. Werhli, M. Grzegorczyk, D. Husmeier, Comparative evaluation of reverse engineering gene regulatory networks with relevance networks, graphical Gaussian models, and Bayesian networks, Bioinformatics 22 (2006) 2523–2531]. We found that altering the number of classification states in both the fuzzy logic and probability models can influence which networks are predicted by both models. As the number of states tends to infinity, the predictions made by both models converge to those of a regression model. Models with a small to moderate number of classification states produced better results from a biological standpoint, compared to models with higher numbers of states. In simulated data, models with differing numbers of classification states produced similar overall results. Thus, increasing the complexity of the models has no apparent benefit, and models with smaller numbers of classification states are therefore preferred based on their ease of linguistic interpretation. The software used in this paper is freely available for non-commercial use at http://louisville.edu/~g0broc01.
In particular, Woolf and Wang [36] developed a fuzzy logic algorithm which searches microarray data sets for regulatory triplets consisting of an activator, repressor, and target gene. Gene expression levels are first classified into low, medium, and high states to varying degrees based on a set of membership functions. Gene pairs are then paired into an activator and repressor gene pair, and this gene pair determines a predicted target gene expression profile based on a set of heuristic rules. The predicted target profile is compared with the expression profile of the remaining genes. The networks are then ranked based on a residual score between the predicted and actual target expression profiles and a variance score which indicates the fluctuation of the activator and repressor gene pair. Networks with a low residual score and a low variance score have the best fit with the regulatory model.

While fuzzy logic provides a simplified structure to build intuitive models based on natural language, the guidelines for constructing these models are open. Thus, building fuzzy logic models is a highly subjective process. For example, Woolf and Wang use three states (low, medium, and high) to classify gene expression data. However, it is entirely reasonable to classify the expression data into five states (low, medium–low, medium, medium–high, and high). It is important to determine whether these slight changes in the model parameters will significantly alter the results.

In this paper, we extend the fuzzy logic technique of Woolf and Wang [36] to include an arbitrary odd number of states. In addition, we develop an alternative probabilistic method for the problem based on the work of Laviolette et al. [17]. Both models are evaluated using publicly available yeast (Saccharomyces cerevisiae) cell-cycle Affymetrix GeneChip microarray data from Cho et al. [9] and simulated data. We also prove that the fuzzy logic and probability models tend to a limiting regression model as the number of states in the models tends to infinity, and demonstrate this with the yeast cell-cycle data set.

2. Methods

2.1. Fuzzy logic model

For a full description of the fuzzy logic model, the reader is referred to [36]. Briefly, the algorithm begins with gene expression values measured over time and/or treatment conditions, standardized to be in the interval [0, 1]. The standardized expression values are classified into some number of states, with partial membership allowed in more than one state. Woolf and Wang [36] used three states (low, medium, and high) to classify the expression data. We extend their model to an arbitrary number of odd states, \( N \). The membership level that an expression value has in each state is determined. This fuzzy target value can then be transformed back into a predicted value between 0 and 1 via the process of de-fuzzification. The method used for our fuzzy system is a zero-order Sugeno model [28], which uses a singleton output function that assigns a single value to each of the \( N \) fuzzy states in the model. The value for a particular fuzzy state is equal to its center of mass, obtained by interpreting the fuzzy membership function as a density function. To transform the fuzzified values back into numbers between 0 and 1, we take a weighted average of this output function, with the weights corresponding to the membership level that the fuzzy output has in each fuzzy state. Fig. 2 illustrates the process, for an activator with membership levels 0.6 low, 0.4 medium, and 0.0 high and a repressor with membership levels 0.2 low, 0.8 medium, and 0.0 high. The resulting predicted target value is 0.389.

2.2. Probability model

While fuzzy logic has an intuitive appeal, important issues still exist concerning its theoretical foundations and how to specify the fuzzy logic model (see [2]) for a statistician’s perspective on issues
concerning fuzzy logic modeling). Recently, researchers have developed probabilistic alternatives to the fuzzy logic controller [17] and critically evaluated the performance of fuzzy logic applied in traditional statistical settings like regression [22] and forecasting [27]. These probability models replicated the performance of the fuzzy logic model while providing a more natural interpretation using the foundations of probability theory. Thus, we developed a probability model following the work of [17] as an alternative to the fuzzy logic model.

Let $X_R$, $X_A$, and $X_T$ be random variables for the expression level of the repressor, activator, and target genes, respectively, at a fixed time point/treatment condition. Let $x_R$, $x_A$, and $x_T$ be the observed values of $X_R$, $X_A$, and $X_T$. Instead of letting the expression values have 'partial' membership in each state, as in the fuzzy logic model, a probability framework is imposed on them and is used to calculate conditional probabilities that they would be classified into the different states, given their observed values.

Let $Y_R$, $Y_A$, and $Y_T$ represent the random variables for the states in which the expression levels of the activator, repressor, and target genes are classified. The probabilities $\mathbb{P}\{Y_R \in k | X_R = x_R\}$ and $\mathbb{P}\{Y_A \in k | X_A = x_A\}$, where $k \in \{1, \ldots, N\}$, are determined by interpreting the membership functions from the fuzzy logic model as conditional probability functions (see Fig. 1). Each triangular function associated with a state in the model gives the probability that an expression level would be classified into that state, given its observed value. As an example, in a three state model, an observed expression level of $\frac{1}{2}$ has a probability of $\frac{1}{2}$ to be classified in the low state and a probability of $\frac{1}{2}$ to be classified in the medium state.

Once $\mathbb{P}\{Y_R \in i | X_R = x_R\}$ and $\mathbb{P}\{Y_A \in j | X_A = x_A\}$ are found for all states $i$, $j \in \{1, \ldots, N\}$, the next step is to determine $\mathbb{P}\{Y_T \in k | X_R = x_R, X_A = x_A\}$ for all states $k \in \{1, \ldots, N\}$. This is accomplished using a decision matrix analogous to the fuzzy logic decision matrix. Let the rows of the decision matrix represent the states for the repressor gene expression level and the columns represent the states for the activator gene expression level. If $\Gamma$ represents the set of cells $(i,j)$ in the decision matrix which imply a target level of $k$, then $\mathbb{P}\{Y_T \in k | X_R = x_R, X_A = x_A\}$ is determined by

$$\mathbb{P}\{Y_T \in k | X_R = x_R, X_A = x_A\} = \frac{\sum_{(i,j) \in \Gamma} \mathbb{P}\{Y_R \in i | X_R = x_R\} \mathbb{P}\{Y_A \in j | X_A = x_A\}}{\sum_{(i,j) \in \Gamma} \mathbb{P}\{Y_R \in i | X_R = x_R\} \mathbb{P}\{Y_A \in j | X_A = x_A\}} \equiv p_{\Gamma} \quad (1)$$

Note that Eq. (1) assumes independence between the activator and repressor genes. Although this is a simplifying assumption for the gene regulatory data considered here (because, for example, the physiological stimuli and/or biological pathways which regulate the activator and repressor genes might be related), the important point to emphasize is that the assumption is explicitly stated. This allows one to explore methods for relaxing this condition when mechanisms for the violation of independence can be modeled. In contrast, in the fuzzy logic model, the intersection of two fuzzy events is resolved by taking the minimum membership level of the events, and thus it is difficult to determine what assumptions are implied concerning the interaction between the activator and repressor genes. Because of the challenge of adequately modeling the deviation from independence for gene interaction data, we focus instead on examining the robustness of our inference of regulatory networks under this simplifying assumption.

The predicted target expression level, $\hat{x}_T$, is found by taking the expectation of the probability density function for $X_T$. This density,
\( f_k \), is a weighted average of density functions associated with each state in the model, \( f_i \) through \( f_k \) (these density functions are membership functions re-scaled to integrate to one). The weight for each \( f_k \) is \( P_{kc} \), and so the predicted target expression level is

\[
X_t = E(X_t) = p_{1c}c_1 + p_{2c}c_2 + \cdots + p_{Nc}c_N
\]  

(2)

where \( c_k \) represents the center of mass for density \( f_k \).

### 2.3. Ranking networks

For each activator/repressor gene pair, both models generate a predicted target gene expression value at each time point. This predicted profile is then compared with the actual profiles of the remaining genes, to determine whether there are any close fits with the regulatory network model. For the fuzzy logic and probability models, each network is evaluated based on a residual and a variance score, analogous to the scoring mechanism used by Woolf and Wang [36]. A low residual score indicates accurate prediction of the target gene’s expression profile by the activator and repressor gene pair. A low variance score indicates that the activator and repressor pair ‘cover’ the decision matrix well. In other words, the gene pair exists in a variety of configurations throughout the time course. This is important, because activator and repressor gene pairs which predict well in a variety of situations should be given greater weight. For the probability model, an alternative to the variance score would be to use the entropy of the resulting decision matrix. However, we opted instead to use the same scoring mechanism for both the fuzzy logic and probability models. An overall score is obtained by multiplying the residual and variance scores together, and networks with lower overall scores rank higher and exhibit a better fit with the regulatory model.

### 2.4. Implementation

We implemented both models with programs written in the C language. For the fuzzy logic and probability models, the programs search through all possible \( G(G−1)(G−2) \) unique activator/repressor/target combinations, where \( G \) is the total number of genes in the data set. Although the algorithm does not scale particularly well with the number of genes, in practice it takes under 6 h to analyze \( \sim 2000 \) genes using a desktop PC. Further, this run time can be significantly reduced using clustering as a pre-processing step [23]. The residual and variance scores are calculated for each gene triplet, and the triplets are ranked on the basis of their composite score. Both programs return a ranked list of networks where the number of networks the program should return is specified.

### 2.5. Evaluation

#### 2.5.1. Yeast cell-cycle data

The models presented in this paper were tested on Affymetrix GeneChip microarray data obtained for the yeast cell-cycle. This data set, originally described by Cho et al. [9], was analyzed by Woolf and Wang [36] with their fuzzy logic model and consists of expression measurements taken at 10-minutes intervals throughout two complete cell-cycles in yeast.

Instead of using all 6220 genes, the data set was first screened to eliminate genes which (a) did not have a maximum expression level above the noise threshold and (b) did not exhibit significant fluctuation during the cell-cycle. Therefore all genes with a maximum expression level of 30 or less (the noise level determined for the yeast data set) were eliminated. In addition, all genes with less than a three fold difference in expression between the minimum and maximum values over the time course were discarded. This screening procedure was the same one used by Woolf and Wang in their analysis [36]. After the screening procedure, the expression levels for each gene were standardized to the interval [0, 1]. The final data set consisted of 2032 gene expression values at 17 time points covering roughly two cell-cycles.

We note that several features of this data (e.g., values sampled over time) are not specifically modeled by our method, though models which do account more specifically for time have been proposed [40]. We chose not to model these features because our goal was to develop an easy-to-understand technique to screen large data sets for potentially interesting regulatory interactions that has broad applicability coupled with reasonable performance. It is of course possible that the performance of our model for this data set could be improved by taking advantage of data-specific features, such as sampling at regular time intervals. We return to this point in the Discussion.

#### 2.5.2. Raf signaling pathway

We also evaluated our models on previously generated data concerning the Raf signaling pathway, taken from Werhli et al. [34]. Raf is a critical signaling protein involved in regulating cellular proliferation of cells in the human immune system, and deregulation of the Raf pathway can lead to carcinogenesis [34]. Fig. 3 shows the currently accepted signaling network, taken from Sachs et al. [24]. The data from Werhli et al. [34] consist of measurements on 11 variables (proteins) of four different fundamental types: observation versus interventional observations, and real versus synthetic observations. They used two different approaches for generating simulated data, (1) sampling from a linear-Gaussian distribution and (2) a more sophisticated approach employed by the software package Netbuilder [38,39]. Further, in addition to the correctly specified network topology, Werhli et al. [34] simulated data from the same topology but with 4 edges removed, resulting in the addition of four “V-structure” networks (coregulation of a variable, or node, by two regulators). In each scenario, 100 observations were generated for each of the 11 proteins, and the process was repeated five times to generate five independent data sets of each type. We present results from a representative subset of all the data sets, namely, the observed and interventional real cytometry data, observed and interventional Gaussian data, and

![Fig. 3. The Raf signaling pathway. Nodes represent proteins, and edges indicate the direction of signal transduction.](image-url)
observed data generated by Netbuilder with a noise level \( \sigma = 0.03 \), under both the correct and V-structure topology.

2.5.2.1. Evaluation measures. We evaluated the performance of the fuzzy logic and probability models on the Raf signaling pathway data using the same measures as in [34], adapted for our generated ranked list of network triplets. The activator/target and repressor/target from each triplet define a ranking of the edges, and can be used to define a receiver operator characteristic (ROC) curve of the relative number of true positive (TP) versus the relative number of false positive (FP) edges. The area underneath the ROC curve (AUC) was used to score the methods, with larger scores indicating an overall better performance. A second score was used to capture the performance of each method at low FP values. This score recorded the number of TP edges discovered when the number of FP edges was five. Each score was recorded under two approaches, the directed evaluation approach (DGE) and the undirected evaluation approach (UGE). The first considered the directionality of the edge when scoring a correctly specified edge, and the second considered only the edge itself.

2.5.2.2. Comparison methods. Werhli et al. [34] evaluated three different methods for reconstructing genetic networks: relevance networks, graphical Gaussian models, and Bayesian networks. We briefly review these methods here, for more detailed information see [34] and the references cited below.

Relevance networks (RNs) [4,5] are a straightforward approach based on pairwise association scores between all pairs of nodes. The association score can be based on the mutual information or the Pearson correlation between the signals associated with each node. The approach has the advantage of being computationally simple, but may have difficulty in distinguishing between direct and indirect interactions.

Graphical Gaussian models (GGMs) [25,26] are inferred from the matrix of partial correlation coefficients between each node. The partial correlation coefficients are calculated from the empirical covariance matrix, which for high-dimensional genomic data should be computed stably using a regularization approach based on shrinkage. [26]. Edges between nodes in the graph are determined by large partial correlation coefficients, and significantly small values are removed from the graph.

Bayesian networks (BNs) consist of a graphical structure combined with a family of conditional probability distributions that together define the joint distribution over the set of nodes. Model structures \( M \) are sampled from a posterior distribution \( P(M|D) \propto P(D|M)P(M) \), where \( P(D|M) \) is the likelihood of the data given the model structure \( M \) and depends on the model parameters. Sampling from the posterior distribution is achieved via Markov Chain Monte Carlo (MCMC) methods, using an efficient proposal algorithm based on node orders [12].

3. Results

3.1. Yeast cell-cycle data

We ran both the fuzzy logic and probability models with various states on the screened and standardized yeast cell-cycle data. To compare the different models and the different number of states, the top 10,000 scoring three-member networks were collected from each analysis. To obtain some intuition about how the models work, it useful to examine the predictions made by the models for a specific example, as done by Wolff and Wang [36]. A top scoring network from the 7-state fuzzy logic model list consists of SWI1 (activator), CDC47 (repressor), and SPC98 (target). The profiles for each gene are displayed in Fig. 4. The top figure displays the profiles for SWI1 and CDC47, while the bottom figure displays the profile for SPC98 and the predicted target profile. SWI1 is present in high levels during the S phase of the yeast cell-cycle (around 40 and 120 min) while CDC47 is present in high levels during the mitotic phase (around 80 and 160 min). Since the two profiles are quite distinct, an interaction between these two genes would not be revealed through clustering algorithms using common distance measures. The bottom figure displays the close prediction of the target gene SPC98 by the SWI1/ CDC47 gene pair. Due to this accurate prediction, this network fits the regulatory model well, resulting in a high ranking for the network. Similarly accurate results are also obtained using the probability model (results not shown).

One question of interest was whether the networks returned by the fuzzy logic and probability models changed as the number of states was altered. Fig. 5 displays the overlap in the top 10,000 scoring networks returned by adjacent state fuzzy logic and probability models. The dashed and dotted lines give the proportion of the results from the fuzzy logic and probability models comparison of the results from the fuzzy logic and probability models as the number of states increases. See text for fuller description.
networks returned which are the same for consecutive state (for example, between the 3- and 5-state) fuzzy logic and probability models, respectively. For the models with lower states, the percentage of networks shared by two adjacent models is fairly low. Thus, initial altering of the number of states significantly changes the networks that are returned by the models. However, for the higher state models, there is considerable overlap between consecutive state models. It is known that fuzzy logic models can approximate any real continuous function on a compact set to an arbitrary degree [31,3], so convergence to some limiting model is expected. In the appendix, we show that both models converge to a specific regression model. The dashed-dotted line gives the proportion of overlap in the networks returned between equivalent state fuzzy logic and probability models, illustrating the similarities between the two models.

One way to get an overall sense of whether the models are returning biologically plausible results, and which models may be performing better, is to look at the number of networks returned that involve genes biologically regulated during the yeast cell-cycle. Cho et al. [9] found 416 yeast genes that were cell-cycle regulated. Of these, 346 were present in our cell-cycle data set after 15

<table>
<thead>
<tr>
<th>Number of states modeled</th>
<th>Fuzzy logic model</th>
<th>Probability model</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1.37</td>
<td>1.24</td>
</tr>
<tr>
<td>5</td>
<td>0.68</td>
<td>0.60</td>
</tr>
<tr>
<td>7</td>
<td>1.22</td>
<td>1.30</td>
</tr>
<tr>
<td>9</td>
<td>0.90</td>
<td>0.96</td>
</tr>
<tr>
<td>13</td>
<td>0.75</td>
<td>0.73</td>
</tr>
<tr>
<td>19</td>
<td>0.77</td>
<td>0.76</td>
</tr>
<tr>
<td>25</td>
<td>0.62</td>
<td>0.61</td>
</tr>
<tr>
<td>31</td>
<td>0.47</td>
<td>0.44</td>
</tr>
<tr>
<td>39</td>
<td>0.32</td>
<td>0.34</td>
</tr>
<tr>
<td>43</td>
<td>0.44</td>
<td>0.44</td>
</tr>
<tr>
<td>47</td>
<td>0.42</td>
<td>0.42</td>
</tr>
<tr>
<td>51</td>
<td>0.46</td>
<td>0.44</td>
</tr>
</tbody>
</table>

Percent of the top 10,000 scoring networks with transcription factors in the activator and repressor positions.

Table 3

There may be several reasons why transcription factors do not appear in the top scoring networks more often. First, many of the transcription factors were screened from the data set prior to analysis. Perhaps screened transcription factors would appear in a higher percentage of high ranking networks. Second, the level of mRNA transcript present may not be indicative of the transcription factor’s activity. For instance, a regulatory protein could be present at a fairly constant level within the cell. The protein could be deactivated, for example via phosphorylation, when necessary. The activity of the regulatory gene is thus determined by activation and deactivation of the gene’s protein within the cell, not by how much mRNA transcript is being produced.

3.2. Raf signaling pathway

We evaluated the fuzzy logic and probability models on data generated from the Raf signaling pathway [34]. We tested models with 5, 15, and 45 states, to determine if there is an effect of varying the states on the success of the models. Table 4 gives the true positive (TP) counts for the fuzzy logic model on six data sets: Gaussian observed and interventional, real observed and interventional, and Netbuilder original and V-structure topology. The probability models had similar results, and so were not included. Also factors in the repressor position. Curiously, there is a drop in the number of transcription factors detected from 3 to 5 states, which then recovers at 7 states. Both models failed to detect many transcription factors in the activator position.

<table>
<thead>
<tr>
<th>Measure</th>
<th>Model</th>
<th>Gaussian</th>
<th>Gaussian</th>
<th>Netbuilder</th>
<th>Netbuilder</th>
<th>Real</th>
<th>Real</th>
</tr>
</thead>
<tbody>
<tr>
<td>DGE</td>
<td>BN</td>
<td>18.4</td>
<td>4.9</td>
<td>4.1</td>
<td>7.7</td>
<td>6.9</td>
<td>3.3</td>
</tr>
<tr>
<td></td>
<td>GMM</td>
<td>5.2</td>
<td>4.7</td>
<td>4.7</td>
<td>5.5</td>
<td>4.1</td>
<td>5.1</td>
</tr>
<tr>
<td></td>
<td>RN</td>
<td>1.8</td>
<td>3.8</td>
<td>5.1</td>
<td>5.0</td>
<td>1.7</td>
<td>5.1</td>
</tr>
<tr>
<td></td>
<td>FL</td>
<td>3.0</td>
<td>3.8</td>
<td>3.0</td>
<td>5.6</td>
<td>1.4</td>
<td>3.0</td>
</tr>
<tr>
<td></td>
<td>FL</td>
<td>3.6</td>
<td>2.8</td>
<td>2.8</td>
<td>6.0</td>
<td>0.6</td>
<td>3.0</td>
</tr>
<tr>
<td></td>
<td>FL</td>
<td>3.4</td>
<td>2.8</td>
<td>3.2</td>
<td>3.0</td>
<td>1.6</td>
<td>2.2</td>
</tr>
<tr>
<td>UGE</td>
<td>BN</td>
<td>18.5</td>
<td>15.8</td>
<td>15.5</td>
<td>14.2</td>
<td>11.1</td>
<td>9.5</td>
</tr>
<tr>
<td></td>
<td>GGM</td>
<td>13.2</td>
<td>14.8</td>
<td>14.8</td>
<td>13.2</td>
<td>9.6</td>
<td>9.6</td>
</tr>
<tr>
<td></td>
<td>RN</td>
<td>6.5</td>
<td>8.1</td>
<td>16.6</td>
<td>13.6</td>
<td>7.1</td>
<td>9.3</td>
</tr>
<tr>
<td></td>
<td>FL</td>
<td>6.6</td>
<td>8.0</td>
<td>4.8</td>
<td>8.0</td>
<td>6.2</td>
<td>6.6</td>
</tr>
<tr>
<td></td>
<td>FL</td>
<td>8.0</td>
<td>7.4</td>
<td>4.8</td>
<td>7.4</td>
<td>4.6</td>
<td>6.2</td>
</tr>
<tr>
<td></td>
<td>FL</td>
<td>7.2</td>
<td>7.8</td>
<td>4.2</td>
<td>6.0</td>
<td>4.4</td>
<td>5.8</td>
</tr>
</tbody>
</table>

Table 4

True positive edges found with no more than five false positive edges, using data generated from the Raf signaling pathway.
in the table are the TP counts for the methods evaluated in [34]. For determining the directed edges (DGE score), the fuzzy logic (FL) model has performance similar to RNs and GGMs. BNs are clearly superior for the Gaussian interventional data, but outperform the FL model by a smaller margin on the remaining data sets. When considering undirected edges (UGE score), the FL model has similar performance to RNs on the real and Gaussian data, but clearly performs worse on the Netbuilder data. GGMs and BNs outperform the FL model in all data sets for the UGE score. However, it should be noted that the intended purpose of the FL model as a screening tool for large (1000s of genes) genomic data sets is different from that of BNs and GGMs, which are primarily useful for refined modeling of smaller gene sets (100s of genes). The poor performance on the Netbuilder simulations (UGE score) for the FL model can perhaps be explained by the non-linear nature of those simulations, which the essentially linear FL model has difficulty capturing. This can be addressed by using adaptive FL models [15,10], although this increases the complexity in two aspects: computation and model interpretation. This point is returned to the Discussion.

Table 5 gives the AUC values for the ROC curves determined by the relative number of TP versus FP counts. The substantive conclusions regarding the performance of the FL model are the same as from Table 4. One last note concerns the relative performance of the various state FL models. From both Tables 4 and 5, it is apparent that increasing the number of states does not necessarily improve the performance of the FL model. Hence, the lower state

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>DGE</td>
<td>BN</td>
<td>0.980</td>
<td>0.782</td>
<td>0.821</td>
<td>0.875</td>
<td>0.697</td>
<td>0.623</td>
</tr>
<tr>
<td></td>
<td>GGM</td>
<td>0.749</td>
<td>0.797</td>
<td>0.798</td>
<td>0.835</td>
<td>0.666</td>
<td>0.644</td>
</tr>
<tr>
<td></td>
<td>RN</td>
<td>0.663</td>
<td>0.641</td>
<td>0.824</td>
<td>0.845</td>
<td>0.553</td>
<td>0.631</td>
</tr>
<tr>
<td></td>
<td>FL 5</td>
<td>0.709</td>
<td>0.614</td>
<td>0.641</td>
<td>0.759</td>
<td>0.555</td>
<td>0.578</td>
</tr>
<tr>
<td></td>
<td>FL 15</td>
<td>0.703</td>
<td>0.608</td>
<td>0.598</td>
<td>0.742</td>
<td>0.491</td>
<td>0.569</td>
</tr>
<tr>
<td></td>
<td>FL 45</td>
<td>0.699</td>
<td>0.623</td>
<td>0.644</td>
<td>0.615</td>
<td>0.465</td>
<td>0.530</td>
</tr>
<tr>
<td>UGE</td>
<td>BN</td>
<td>0.966</td>
<td>0.885</td>
<td>0.905</td>
<td>0.933</td>
<td>0.791</td>
<td>0.690</td>
</tr>
<tr>
<td></td>
<td>GGM</td>
<td>0.820</td>
<td>0.881</td>
<td>0.883</td>
<td>0.904</td>
<td>0.713</td>
<td>0.685</td>
</tr>
<tr>
<td></td>
<td>RN</td>
<td>0.710</td>
<td>0.681</td>
<td>0.916</td>
<td>0.915</td>
<td>0.569</td>
<td>0.668</td>
</tr>
<tr>
<td></td>
<td>FL 5</td>
<td>0.694</td>
<td>0.656</td>
<td>0.535</td>
<td>0.708</td>
<td>0.571</td>
<td>0.639</td>
</tr>
<tr>
<td></td>
<td>FL 15</td>
<td>0.675</td>
<td>0.639</td>
<td>0.536</td>
<td>0.685</td>
<td>0.477</td>
<td>0.625</td>
</tr>
<tr>
<td></td>
<td>FL 45</td>
<td>0.686</td>
<td>0.652</td>
<td>0.549</td>
<td>0.561</td>
<td>0.475</td>
<td>0.622</td>
</tr>
</tbody>
</table>

Fig. 6. Convergence of the fuzzy logic model illustration of the convergence of the fuzzy logic model. As the number of states increases, the prediction surface of the models closely resemble a regression model where $x_t = -1/2 + 1/2(x_A - x_R)$. 
functions in the fuzzy logic algorithm might help to overcome a cube, Epanechnikov) and altering the spacing of the membership may be less appropriate for use as a screening tool of genomic data. The simple zero-order Sugeno method we have implemented, and these inference systems are more computationally intensive than defuzzifying the fuzzy target value into a crisp number. However, Tsukamoto [29] systems, which provide alternative means of the next could be used (an example of such a model is given in [34]). This paper demonstrates an effect on the biological significance of the results found in the yeast cell-cycle microarray data and data generated from the Raf signaling pathway. The fuzzy logic and probability models we implemented are somewhat deficient at discovering networks which contain non-linear effects among the regulatory genes. However, both models can be modified to detect more complicated networks. In the probability model, the activator and repressor gene are assumed to be independent (this assumption is used in Eq. (1)). However, relaxing this assumption simply requires the use of an appropriate discrete bivariate distribution to model the relationship between these genes, although this may make the calculations in Eq. (1) considerably more complicated. For the fuzzy logic model, consulting with experts in the particular field of study could provide knowledge about gene interactions in linguistic form, which could then be translated into an appropriate fuzzy logic model. Alternatively, use of adaptive fuzzy logic models as in [15,10] can be used to automatically tune certain aspects of the model (rule weighting, hedging of fuzzy regions corresponding to the term set), which are learned from the data.

Increasing the complexity of the models needs to be weighed against the usefulness of these models as exploratory tools, however. If the purpose is to screen microarray data for potential simple gene regulatory networks, then we have shown that both models provide results that can potentially be useful in guiding subsequent experimental approaches. Ideally, the searches need to be quick, and building more complicated models may compromise this goal.

It is worth noting that our methods allow the analysis of a large number of genes in a relatively quick manner (just a few hours on a desktop PC), while the methods we have used for comparison are typically confined to smaller data sets. This rapid evaluation does not necessarily sacrifice performance. For the Raf pathway data, our model performs nearly as well as RNs and just below the others. For the real data, for which the models incorporated by all of the methods are necessarily approximations to the true nature of gene interactions, our methods perform similarly to the others. This hints at a degree of robustness to model misspecification in our models.

3.3. Convergence of fuzzy logic and probability models

Both the increasing percentage of overlap in the top ranking networks as the number of states increases and the previous work in fuzzy logic theory [31,32] indicate that both the fuzzy logic and probability models will converge to a limiting model as the number of states increases. This behavior is illustrated in Fig. 6 for the fuzzy logic model that we have implemented. The probability model closely resembles the fuzzy logic model, except with smoother edges along the prediction surface. It is readily seen that both models tend to a limiting regression model

\[
\hat{x}_T = \frac{1}{2} \sum_{i=1}^{2} (x_A - x_R)
\]  

where \(x_A, x_R, \) and \(x_T\) all lie in the interval [0, 1]. The proof of this convergence is given in Appendix A for the fuzzy logic model. The proof for the probability model is very similar.

4. Discussion

In this work, the fuzzy logic model of Woolf and Wang [36] was expanded and a new probability model was developed for detecting gene regulatory networks. The performance of both models was compared using yeast cell-cycle microarray data and data generated from the Raf signaling pathway. The fuzzy logic and probability models produced similar results for both data sets.

Both models performed similar to RNs on the Gaussian and real data generated from the Raf pathway, but had suboptimal performance on the Netbuilder simulated data. Altering the number of states in both models had little effect on the accuracy of the networks returned for the Raf pathway data, but appeared to have an effect on the biological significance of the results found in the yeast cell-cycle data. The low to middle state models (roughly 15 states and below) appeared to find more networks of biological interest compared to the higher state models. Since these models have the additional advantage of intuitive linguistic interpretation, using a smaller number of classification states should be favored over a larger number of states.

Because our goal was to develop a quick and general screening tool to mine gene regulatory networks from large expression data sets, our methods used both a simple model for gene interactions and a straightforward class of membership functions. More detailed modeling of either component could lead to increased performance, but at the potential cost of being more narrowly applicable. For example, when data are sampled over a time course, as are the yeast cell-cycle data, a model that specifically incorporates correlation of expression values from one point to the next could be used (an example of such a model is given in [40]).

Another extension of our fuzzy logic model includes use of alternative inference systems such as the Mamdani [21,20] and Tsukamoto [29] systems, which provide alternative means of defuzzifying the fuzzy target value into a crisp number. However, these inference systems are more computationally intensive than the simple zero-order Sugeno method we have implemented, and may be less appropriate for use as a screening tool of genomic data.

Lastly, use of different membership functions (e.g. Gaussian, tri-cube, Epanechnikov) and altering the spacing of the membership functions in the fuzzy logic algorithm might help to overcome a possible shortcoming in the triangular membership function of Fig. 1 which prevents an observation from belonging to more than two classes. We note that this discretization may explain the poor behavior seen in the biological results for the higher state models, as these models treat the expression value as a discrete value approaching a point. Indeed, this is exactly what happens in the limiting model (3), as the number of states tends to infinity.

The simulated Raf pathway data also indicates that the fuzzy logic and probability models we implemented are somewhat deficient at discovering networks which contain non-linear effects among the regulatory genes. However, both models can be modified to detect more complicated networks. In the probability model, the activator and repressor gene are assumed to be independent (this assumption is used in Eq. (1)). However, relaxing this assumption simply requires the use of an appropriate discrete bivariate distribution to model the relationship between these genes, although this may make the calculations in Eq. (1) considerably more complicated. For the fuzzy logic model, consulting with experts in the particular field of study could provide knowledge about gene interactions in linguistic form, which could then be translated into an appropriate fuzzy logic model. Alternatively, use of adaptive fuzzy logic models as in [15,10] can be used to automatically tune certain aspects of the model (rule weighting, hedging of fuzzy regions corresponding to the term set), which are learned from the data.

5. Conclusion

As noted in the introduction, the literature for modeling genetic networks is growing very rapidly [35,30,14]. However, there is still a general lack of research concerning how these methods relate to each other (but see Werhli et al. [34]). This paper demonstrates that the fuzzy logic and probability models converge to a regression surface, drawing a significant link among these methods. Two promising avenues for future research are thus also identified by this study. The first is the incorporation of more realistic models for gene interactions, which will necessarily increase the complexity of the inference procedure. The second is to examine the feasibility of combining the concepts from several networking models. For example, the conditional probability functions from the probability model might be useful in the framework of a Bayesian network.

Overall, the fuzzy logic and probability models appear to produce biologically reasonable results. However, all such methods will suffer from inherent difficulties in discovering regulatory networks with microarray data. First, any interactions pulled from the microarray data are not necessarily causative in nature. Quite possibly, the associations among genes within a network are indirect, perhaps through involvement in similar biological pathways.
Further experimental approaches will be needed to determine the validity of the genetic interactions suggested by the network models. Second, gene expression data may not indicate a gene’s level of biological activity, and the network models are unable to detect valid interactions involving these genes using microarray data. In spite of these considerable challenges, we remain optimistic that approaches such as those proposed here will provide a useful exploratory tool to mine large microarray data sets for potential genetic regulatory interactions. This information can then be used to focus future experiments, saving the researcher precious time and resources.

Acknowledgements

The authors thank four anonymous reviewers for comments on an earlier version of this manuscript, and the guest editors for their support during the submission and review process. GNB thanks the National Center for Genome Resources (NCGR) for supporting him when the majority of this research was conducted. The work was supported by the National Science Foundation grant 0078307.

Appendix A

Using the triangular, equally spaced membership functions displayed in Fig. 1, each expression level will be classified into at most two states with nonzero membership values. Let $k_A$ and $k_R$ represent the smallest states with positive membership values for the activator and repressor gene expression levels, $x_A$ and $x_R$. The membership levels that $x_A$ and $x_R$ have in $k_A$ and $k_A + 1$ and $k_R$ and $k_R + 1$, respectively, are

$$
\phi_{k_A}(x_A) = k_A - (N - 1)x_A \\
\phi_{k_R}(x_R) = k_R - (N - 1)x_R \\
\phi_{k_A+1}(x_A) = (N - 1)x_A - (k_A - 1) \\
\phi_{k_R+1}(x_R) = (N - 1)x_R - (k_R - 1)
$$

where $\phi_k(x)$ is the fuzzy membership level that $x$ has in state $k$ and $N$ is the total number of states in the model.

In fuzzy logic, the minimum is used to resolve the intersection of two events, so there are at most four cells in the decision matrix which have nonzero membership values associated with them. These nonzero membership levels are given by

$$
\phi_{k_Ak_A} = \min(\phi_{k_A}, \phi_{k_A}) \\
\phi_{k_A(k_A+1)} = \min(\phi_{k_A}, \phi_{k_A+1}) \\
\phi_{k_A+1k_A} = \min(\phi_{k_A+1}, \phi_{k_A}) \\
\phi_{k_A+1(k_A+1)} = \min(\phi_{k_A+1}, \phi_{k_A+1})
$$

In the limit, as the number of states tends to infinity, these four nonzero cells will all lie above or below the diagonal of the decision matrix. Consider the case where the nonzero cells all lie below the diagonal and $k_R - k_A$ is even (the proofs in the other cases are analogous). When $k_R - k_A$ is even then the predicted target expression level $x_T$ has positive membership in two states, $(N + 1 - (k_R - k_A))$ and $N - (k_R - k_A)$, with membership levels given by

$$
\phi_{N-k_R-k_A}(x_T) = \max \left\{ \phi_{k_Rk_A}, \phi_{k_R(k_A+1)} \right\} \\
\phi_{N-k_R-k_A}(x_T) = \phi_{k_Rk_A+1}
$$

The maximum is taken to evaluate the union of a set of events. The predicted target expression value, $x_T$, is then

$$
x_T = \frac{c_1 - 1}{c_2} \left( \max \left\{ \phi_{k_Rk_A}, \phi_{k_R(k_A+1)}, \phi_{k_Rk_A+1} \right\} \right) + \frac{c_1 - 2}{c_2} \left( \phi_{k_Rk_A+1} \right)
$$

where $c_1 = 2(N + 1 - (k_R - k_A))$ and $c_2 = (N - 1)(\max(\phi_{k_R}, \phi_{k_Rk_A+1}, \phi_{k_R(k_A+1)} + \phi_{k_Rk_A+1}))$. To evaluate this expression further, we have to assume an ordering on the terms $\phi_{k_A}(\phi_{k_A+1}), \phi_{k_R}(\phi_{k_R+1},$ of which there are eight possible. Consider the case where $\phi_{k_R} < \phi_{k_A} < \phi_{k_A+1} < \phi_{k_R+1}$. The other seven cases are similar. Substituting values for $\phi_{k_A+1}$ and $\phi_{k_R}$, the above expression for $x_T$ reduces to

$$
x_T = \frac{1}{N - 1} \left( N + 1 - k_R - k_A \right)
$$

which converges to the regression model $x_T = \frac{1}{N - 1} (x_A - x_R)$ as $N \to \infty$.

References


