Academy of Scientific Research and Technology
27th National Radio Science Conference
Faculty of Electronic Engineering, Menoufia Univ., Menouf, Egypt
16-18 March 2010

Performance Comparison of the Structure Learning Bayesian Network Algorithms Using Gene Expression Data

Fadhl M. Al-Akwaa[1,2], Nahed H. Solouma[2], Yasser M. Kadah[2]

[1]Biomedical Eng. Dept., Univ. of Science & Technology, Sana'a, Yemen (E-mail f_alakwa@k-space.org)
[2] Biomedical Engineering Department, Cairo University, Giza, Egypt

Abstract

Understanding gene interactions in complex living systems can be seen as the ultimate goal of the systems biology revolution. Hence, to fully understand disease ontology and to reduce the cost of drug development, gene regulatory networks (GRN) have to be constructed. During the last decade, many GRN inference algorithms like 'Bayesian network' that are based on genome-wide data have been developed to unravel the complexity of gene regulation. Recently, many of structure learning algorithms were used to learn Bayesian network that have shown promise in gene regulatory network reconstruction. In this paper we apply different structure learning algorithms on actual microarray data to obtain a better understanding of their relative strengths and weaknesses on the system biology community and we evaluate their outputs from different perspectives.

1. Introduction

As basic building blocks of life, genes, as well as their products (proteins), do not work independently. Instead, in order for a cell to function properly, they interact with each other and form a complicated network. Gene networks represent the relationship between sets of genes that coordinate to achieve different tasks.

With the advent of high-throughput microarray technologies, mRNA expression levels of tens of thousands of genes can now be measured simultaneously. Construction of gene networks from these experimental data will greatly facilitate cellular functional dissection at the molecular level.

a variety of computational methods have been considered for reconstructing gene networks from observational gene expression data including, for example, linear models [1] and Boolean network models [2] , Bayesian network (BN) [3]. Bayesian network methods have shown promise in gene regulatory network reconstruction for the following reasons: (1) the sound probabilistic semantics allows BNs to deal with the noises that are inherent in experimental measurements; (2) BNs can handle missing data and permit the incomplete knowledge about the biological system and (3) BNs are capable of integrating prior biological knowledge into the system.

Generally, a BN is a graphical representation of the dependence structure between multiple interacting quantities. This graphical representation is more commonly called a directed acyclic graph (DAG) as shown in Figure 1. The nodes or the vertices of the DAG represent the random variables in the network while the edges connecting the vertices represent the causal influence of one node on the other.

BN-based gene network inference requires the learning of the BN structure, which is an optimization problem in the space of the DAGs. Many structure learning methods have been proposed in the literature, and it is important to understand their relative merits and shortcomings. Although there are a great many algorithms for learning Bayesian networks from data, they can be categorized as either conditional independence (CI) test-based methods or scoring-based methods. The CI-based methods analyze the dependence and independence relationships among variables via CI tests and construct the networks that characterize these relationships. The scoring-based methods consist of two components: (1) a scoring function that assesses how well a network fits the data and (2) a search method to find

networks with high scores. Acid [4], conducted a comparative study of different structure learning algorithms on data from an emergency medical service.
In the present paper we apply currently available Structure learning algorithms on actual microarray data to obtain a better understanding of their relative strengths and weaknesses on the system biology community and we have carried out a series of experiments to evaluate their behavior from different perspectives.

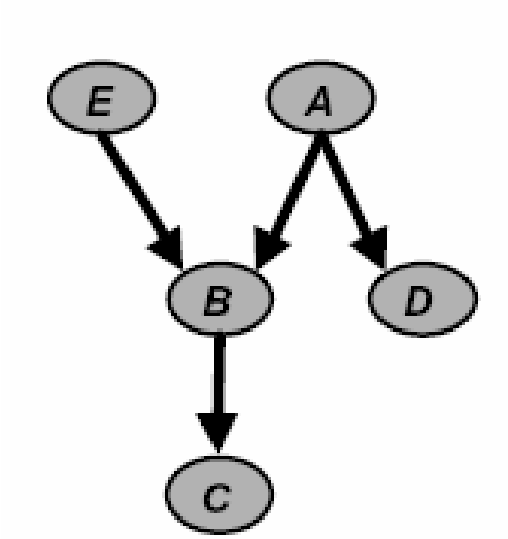

Fig. 1. An example of a simple Bayesian network structure modified from**[using].** This network structure implies several conditional statements: I(A;E),I(B;D|A,E),I(C;A,D,E|B),I(D;B,C,E|A),I(E;A,D). The network structure also implies that the joint distribution has the product form P(A,B,C,D,E)=P(A)P(B|A,E)P(C|B)P(D|A)P(E)

2. Methodology

2.1. The learning Algorithms

We used several algorithms for learning the structure of a Bayesian network from the data set. The selected algorithms are driven by different principles and/or metrics, so the resulting models may differ in their result and the relationships they extract. The structure learning algorithms were used in this comparison are: K2 algorithm [5], Markov Chain Monte Carlo (MCMC)[6], Bayesian Network Power Constructor (BNPC)[7] and Greedy Search in the Markov Equivalent Space (GSMES)[8]. An overview of these algorithms is presented in [7].

2.1.1 K2 algorithm

The K2 Algorithm [5] is a greedy search algorithm that learns the network structure of the BN from the data presented to it. It attempts to select the network structure that maximizes the network's posterior probability given the experimental data. The K2 algorithm reduces this computational complexity by requiring a prior ordering of nodes as an input, from which the network structure will be constructed. The ordering is such that if node X_i comes prior to node X_j in the ordering, then node X_j cannot be a parent of node X_i. In other words, the potential parent set of node X_i can include only those nodes that precede it in the input ordering.

2.1.2 MCMC

Markov chain Monte Carlo (MCMC) methods, are a class of algorithms for sampling from probability distributions based on constructing a Markov chain that has the desired distribution as its equilibrium distribution. The state of the chain after a large number of steps is then used as a sample from the desired distribution. The quality of the sample improves as a function of the number of steps. We can use a Markov Chain Monte Carlo (MCMC) algorithm called Metropolis-Hastings (MH) to search the space of all DAGs[6].

2.1.3 BNPC

The BN Power Constructor (BNPC), uses independence tests and mutual information [7]. This algorithm has a three-phase operation: drafting, thickening, and thinning. In the first phase, the algorithm computes mutual information of each pair of nodes as a measure of closeness, and creates a draft based on this information. In the second phase, the algorithm adds arcs when the pairs of nodes are not conditionally independent on a certain conditioning set. In the third phase, each arc is examined using conditional independence tests and will be removed if the two nodes of the arc are conditionally independent.

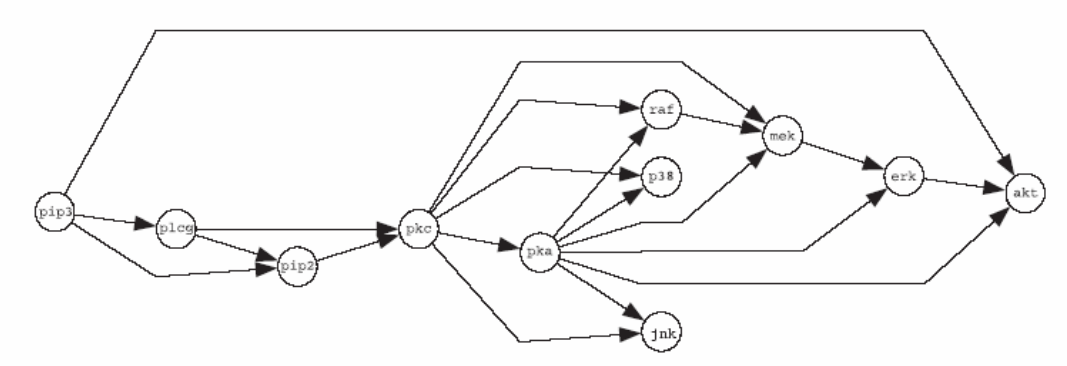

Fig. 2. Raf signaling pathway. The graph shows the currently accepted signaling network, taken from [10].

2.1.4 GSMES

Recent works have shown the interest of searching in the Markov equivalent space . It has proved that a greedy search in this space (with an equivalent score) is more likely to converge than in the DAGs space [8]. This method works in two steps. First, it starts with an empty graph and adds arcs until the score cannot be improved, and then it tries to suppress some irrelevant arcs.

2.2. The Data set

The structure learning algorithms was tested with synthetic data samples randomly generated from Raf signaling network, depicted in Figure 2. The random generation of data samples was done to ensure the robustness of the algorithms. We used the sampling function which was implemented in Bayesnet Toolbox[9]. Raf network includes 11 nodes and 20 arcs. Raf is a critical signalling protein involved in regulating cellular proliferation in human immune system cells. The deregulation of the Raf pathway can lead to carcinogenesis, and the pathway has therefore been extensively studied in the literature [10].

2.3. The Comparison Methodology

The comparison methodology used in this paper is similar with the method was used in [11]. The existence of the known network structures allows us to define three important terms, which indicate the performance of the algorithm (in terms of the number of graphical errors in the learnt structure).

- Correct edges(C): Edges present in the original network and in the learnt network structure.
- Missing edges (M): Edges present in the original network but not in the learnt network structure.
- Wrongly oriented edges (WO): Edges present in the learnt network structure, but having opposite orientation when compared with the corresponding edge in the original network structure.
- Wrongly connected edges (WC): Edges not present in the original network but included in the learnt network structure.

3. Results and Discussion

The simulations of these structure learning algorithms in our comparative evaluation study were carried out with the Bayesnet Toolbox[9] and Structure Learning Package [12]. The tests are carried on an

Intel Core Due 1.8 GHz CPU and 1 GB RAM. Table 1 shows the parameters for each candidate learning algorithms. Table 2 &3 show the performances of the algorithms for the Raf networks with 1000 and 100 data samples generated respectively. Table 2 & 3 report the mean results (the results averaged over 100 trial runs).

Table 2 and 3 show that these algorithms differ significantly in their predictability power and how could use larger data set improve algorithms performance except for BNPC and GSMES which is against our expectation. We attempt to contact the corresponding authors to explain these results. Also the low performance of the small data set promotes the importance of solving the dimensionality reduction of the gene reverse engineering algorithms where the numbers of experiments are minimal.

For the k2 algorithm we present the results obtained with the correct order (of which we have the knowledge, since the network structure is known), order known from Maximum Weight Spanning Tree (MWST) [13] and with the random order. The results for K2 with correct order are the optimal results one can get. K2 algorithms outperform the learning algorithms. For its result with known order about 17 over 20 edges were covered perfectly. Also its result with random order outperforms the tested algorithms. Moreover the results of k2 algorithm getting order from MWST directed the authors to develop a new algorithm to get network order.

GSMES is the only method which has wrong orientation edges.

Table 1
The parameters for each structure learning algorithms
See [10] for more details about them.

Learning Algorithm	**Parameters setting**
K2 (known order)	
K2(order from MWST)	max_fan_in = 2
K2 (random order)	
MCMC	Nsamples=100*11; burnin=5*11
GSMES	_____
BNPC	epsilon=0.05

Table2
Comparative performance for Raf network with 1000 data samples generated randomly 100 times.

Learning Algorithm	**C**	**M**	**WO**	**WC**
K2 (known order)	17.12	2.88	0	0.16
K2(order from MWST)	12.49	7.51	0	7.35
K2 (random order)	8.43	11.57	0	10.86
MCMC	5.86	**14.14**	0	13.84
GSMES	9.82	**10.18**	1.72	10.31
BNPC	2.35	17.65	0	5.08

Table 3
Comparative performance for Raf network with 100 data samples generated randomly 100 times.

Learning Algorithm	**C**	**M**	**WO**	**WC**
K2 (known order)	12.82	7.18	0	2.82
K2(order from MWST)	8.81	11.19	0	6.29
K2 (random order)	5.76	14.24	0	9.51
MCMC	3.98	16	0	12.19
GSMES	9.18	10.82	1.51	8.91
BNPC	1.97	18.03	0	2.1

4. Conclusion

In this paper we aim to compare the structure learning algorithms performance on a gene expression data. We see how the data set size could alter their performance. Also we show the importance of developing the correct network order algorithms. For simulated data was used here, the true structure of the regulatory network is known; this allows us, in principle, to faithfully evaluate the prediction results. However, the sampling approach used for data-generation is a simplification of real molecular biological processes, and this might lead to systematic deviations and a biased evaluation. We can overcome this using real laboratory data.

Acknowledgments

The authors thank Kevin P. Murphy, Philippe Leray's and Dirk Husmeier for the Bayesnet Toolbox, Structure Learning Package and DBmcmc software package respectively. Fadhl Al-Akwaa was supported by The Yemeni Ministry of Higher Education and Scientific Research, Yemen-Sana'a.

References

[1] P. D'Haeseleer, X. Wen, S. Fuhrman, and R. Somogyi, "Linear modeling of mRNA expression levels during CNS development and injury.," *Pac Symp Biocomput,* vol. 41–52, 1999.

[2] I. Shmulevich, E. R. Dougherty, S. Kim, and W. Zhang, "Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks," *Bioinformatics,* vol. 18, pp. 261-274, February 1 2002.

[3] N. Friedman, M. Linial, I. Nachman, and D. Pe'er, "Using Bayesian Networks to Analyze Expression Data," *Journal of Computational Biology,* vol. 7, pp. 601-620, 2000.

[4] S. Acid, L. M. d. Campos, J. M. Fernández-Luna, S. Rodríguez, J. M. Rodríguez, and J. L. Salcedo, "A comparison of learning algorithms for Bayesian networks: a case study based on data from an emergency medical service " *Artif. Intell. Res,* vol. 18, pp.,445–490, 2003.

[5] G. F. Cooper and E. Herskovits, "A Bayesian method for the induction of probabilistic networks from data. ," *Mach. Learn,* vol. 9, pp. 309–347, 1992.

[6] K. P. Murphy, "Active learning of causal bayes net structure.," 2001b.

[7] J. Cheng, D. Bell, and W. Liu, "Learning Bayesian networks from data: an information-theory based approach," *Artif. Intell. Res,* vol. 137, pp. 43-90, 2002.

[8] D. Chickering, "Optimal structure identification with greedy search.," *J. Mach. Learn. Res,* vol. 3, pp. 507-554, 2002.

[9] K. P. Murphy, "http://www.cs.ubc.ca/~murphyk/Software/BNT," 1999.

[10] K. Sachs, O. Perez, D. Pe'er, D. A. Lauffenburger, and G. P. Nolan, "Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data," *Science,* vol. 308, pp. 523-529, April 22 2005.

[11] X.-w. Chen, G. Anantha, and X. Wang, "An effective structure learning method for constructing gene networks," *Bioinformatics,* vol. 22, pp. 1367-1374, June 1 2006.

[12] P. LeRay, "http://banquiseasi.insa-rouen.fr/projects/bnt-slp/," 2003.

[13]C. Chow and C. Liu, "Approximating discrete probability distributions with dependence trees," *IEEE Transactions on Information Theory,* vol. 14, pp. 462–467, 1968.