

A Computational Approach to Reconstructing Gene Regulatory Networks

Xutao Deng and Hesham Ali
Department of Computer Science
College of Information Science and Technology
University of Nebraska at Omaha
Omaha, NE 68182-0116
xdeng@mail.unomaha.edu

Abstract

With the rapid accumulation of gene expression data in publicly accessible databases, computational study of gene regulation has become an obtainable goal. Intrinsic to this task will be data mining tools for inferring knowledge from biological data. In this project, we have developed a new data mining technique in which we adapt the connectivity of a recurrent neural network model by indexing regulatory elements and including nonlinear interaction terms. The new technique reduces the number of parameters by $O(n)$, therefore increasing the chance of recovering the underlying regulatory network. In order to fit the model from data, we have developed a genetic fitting algorithm with $O(n)$ time complexity and that adapts the connectivity during the fitting process until a satisfactory fit is obtained. We have implemented this fitting algorithm and applied it to two data sets: Rat Central Nervous System development (CNS) data with 112 genes, and Yeast whole genome data with 2467 genes. With multiple runs of the fitting algorithm, we were able to efficiently generate a statistical pattern of the model parameters from the data. Because of its adaptive features, this method will be especially useful for reconstructing coarse-grained gene regulatory network from large scale or genome scale gene expression data sets.

1. Introduction

Recent technology advancement has made large-scale gene expression surveys a reality. Along with genome sequence data, massive gene expression data sets have made biology a data-rich subject. These data sets provide an opportunity to directly view the activity of hundreds of genes in parallel. However, manual analysis of these huge data sets is often not practical. Development of computational methods and data mining tools for knowledge inference from gene expression data base is the only way to face this challenge.

Current widely used methods to facilitate analysis of gene expression data sets are clustering, classification, and visualization tools [1]. These methods are used to group genes based on the similarity of expression patterns. If two genes are clustered together in this way, then they may share a common functional role. But if they have distinct expression patterns, how they are related? It is obvious that a simple clustering analysis cannot answer this question.

In order to answer this kind of query, we need to construct a gene regulatory network. The knowledge of gene regulatory network and its interactions will further the understanding of important biological processes such as disease, cell cycle, and development. Drawing regulatory network information from time series expression data sets is a reverse engineering problem. A common approach to solving this problem has its basis in mathematical modeling; we adopt this approach here. We first construct a mathematical model which simulates the real gene regulatory system with some simplification. Then we apply fitting algorithms to search for the best model parameters that will let the model behave closest to the data. The result parameters are then used to construct the regulatory network.

2. Adaptive model

Many modeling frameworks, including Boolean networks [2], linear additive networks [3], neural networks [4], have been applied in reverse engineering regulatory networks. Here we develop a new model based on recurrent neural networks.

$$\frac{dv_i}{dt} = m_i S(w_{pi}v_p + w_{qi}v_q + b_i) - r_i v_i \quad (1)$$

where the variables and function are defined as:

$S(\cdot)$: Sigmoid function

v_i : mRNA concentration for the i^{th} gene

t : time

m_i : maximum allowed concentration for i^{th} gene

p, q : indices of regulating genes

w_{pi}, w_{qi} : regulation weight from regulating genes
 b_i : bias concentration for i^{th} gene
 r_i : decay constant for i^{th} gene

For better resemblance to a real biological system, the model is designed to be value-continuous, time-continuous, and value-constrained by a sigmoid function. Because gene regulatory networks are very sparse networks with most connections being zero [3], it is not necessary to design a fully connected model. By indexing regulating genes instead of including all genes in the model, we avoid a fully connected model which is the case in a recurrent neural network. The advantage of doing this is two-folded: computation load is significantly reduced and, more importantly, the number of parameters is reduced by $O(n)$. As we know, search space grows exponentially with number of parameters. Reducing the number of parameters improves the chances of pinning down the right regulatory network.

3. Fitting algorithm

The basic idea of the fitting process is iteratively incrementing connectivity of the basic model (equation 1) until the data is satisfied. Inside each iteration, a Genetic Algorithm (GA) is used to search for optimal parameters for the model.

```

begin with basic model;
while (true) do
{
  GA(m Generations);
  if ( fitness is sufficient)
    then exit loop;
  else
    connectivity := connectivity + 1 ;
}
output model;
end;
  
```

For large scale systems, network connectivity k can be viewed as a constant in contrast to the number of genes n . Thus, the iteration does not add complexity to the fitting algorithm. We use the Runge-Kutta method to derive a numerical solution for each differential equation. Because of the simplicity of the model, it takes $O(n)$ time to derive a total n equations in the model. Therefore, $O(n)$ is also the complexity of a GA for a fixed number of generations. The overall time complexity is thus $O(n)$.

4. Application to Rat CNS data

Rat CNS data was obtained from [5]. The data set contains 112 genes with 9 time points. Following the procedure from [5], we first clustered the 112 genes into 5

groups, each a distinct expression pattern. We then applied our method to the reduced data set. With 300 runs, we generated a statistical pattern of the parameter distribution (not shown here). The reconstructed network is shown in Fig 1. The results of application to Yeast whole genome data set are not shown here.

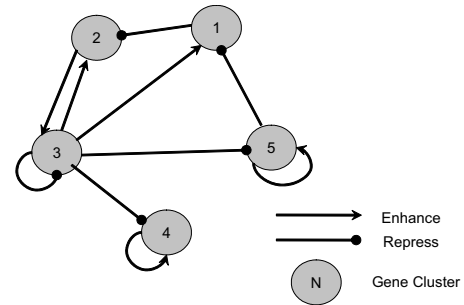


Figure 1. Reconstructed Rat CNS network

5. Conclusion

We have developed a modeling technique to efficiently reconstruct gene regulatory networks from time series expression data sets. This method features an adaptive neural network model and a linear fitting algorithm which fits the model from gene expression data sets. We have applied this method to Rat CNS data set and Yeast whole-genome data set and efficiently generated predictive gene regulatory networks for each system. The validity of this method needs to be examined from further experimental results.

6. References

- [1] Eisen M.B., Spellman, P.T., Brown P.O., and Botstein D., "Cluster analysis and display of genome-wide expression patterns", *Proc Natl Acad Sci, U S A*, 1998 Dec 8; 95(25): 14863-8.
- [2] Kauffman S.A., *The Origins of Order: Self-Organization and Selection in Evolution*, Oxford University Press, New York, 1993.
- [3] D'haeseleer, P., Liang, S., and Somogyi, R., "Genetic network inference: from co-expression clustering to reverse engineering" *Bioinformatics*, 2000, 16(8):707-26.
- [4] Wahde M., and Hertz J., "Modeling genetic regulatory dynamics in neural development", *Journal of Computational Biology*, 2001, 8:429-442.
- [5] Wen X., Fuhrman S., Michaels G.S., Carr D.B., Smith S, Barker J.L., and Somogyi R., "Large-scale temporal gene expression mapping of central nervous system development", *Proc Natl Acad Sci, U S A*, 1998 Jan 6;95(1):334-9.