Available online at www.sciencedirect.com
ScienceDirect
Neurocomputing 71 (2008) 600–610
NEUROCOMPUTING
www.elsevier.com/locate/neucom

A clustering-based approach for inferring recurrent neural networks as gene regulatory networks

Wei-Po Lee[a,*], Kung-Cheng Yang[b]

[a]Department of Information Management, National Sun Yat-sen University, Kaohsiung, Taiwan
[b]Department of Management Information Systems, National Pingtung University of Science and Technology, Pingtung, Taiwan

Available online 1 October 2007

Abstract

Constructing genetic regulatory networks is one of the most important issues in system biology research. Yet, building regulatory models manually is a tedious task, especially when the number of genes involved increases with the complexity of regulation. To automate the procedure of network construction, in this work we establish a clustering-based approach to infer recurrent neural networks as regulatory systems. Our approach also deals with the scalability problem by developing a clustering method with several data analysis techniques. To verify the presented approach, experiments have been conducted and the results show that it can be used to infer gene regulatory networks successfully.

Keywords: Gene regulatory network; Recurrent neural network; Reverse engineering; System modeling

1. Introduction

With the newly invented experimental tools and techniques, it is now possible to simultaneously measure and record information of multiple genetic reactors and their interactions. By analyzing and studying the interactions between genes in a regulatory network, we can now uncover some complex behavior patterns [19]. Gene network modeling provides a methodology to manipulate time series data for building a model that can describe the observed phenotypic behavior of a system to be studied. But it should be noted that gene regulation networks (GRNs) are complex biological systems consisting of many interacting components. In many cases, the detailed molecular mechanism that governs the interactions among system components is not yet fully understood. Generally, it is difficult to model these complex processes mathematically, because the networks involve many genes connected through interlock feedback loops and their description requires a representation general enough to capture the characteristics of experimentally observed responses. Also the model must be robust against noises and can be enforced by some constraints on the connectivity, stability, and redundancy.

Many gene regulation models have been proposed [7,8]. The models can range from very abstract models (involving Boolean values only) to very concrete ones (including fully biochemical interactions with stochastic kinetics). The former approach is mathematically tractable that provides the possibility of examining large systems; but it cannot infer networks with feedback loops. The latter approach is more suitable in simulating the biochemical reality and is more realistic to the experimental biologists. However, due to its computational complexity, this approach can only be applied to very small systems. To reconstruct a gene regulatory network, one can start from an initial model, simulate the system behaviors for a variety of experimental and environmental conditions, and then compare the predictions with the observed gene expression data to give an indication of the adequacy of the model. If the experimental data is considered reliable, but the observed and predicted system behavior does not match the data, the model must be revised. The activities of constructing and revising models of the regulatory network, simulating the behavior of the system, and testing

*Corresponding author.
E-mail address: wplee@mail.nsysu.edu.tw (W.-P. Lee).

0925-2312/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2007.07.023

the resulting predictions are repeated until an adequate model is obtained.

As the above procedure of constructing a network from data takes a considerable large amount of time, an automated procedure is thus advocated. Reverse engineering is a paradigm with great promise for analyzing and constructing GRNs [5,13]. It is an effective way to utilize either literature or laboratory experimental data to determine the underlying network structure of a given model. To reconstruct a network, experimental iterations and priori knowledge are required until sufficient data is available to perform computational inference of the network structure. In the case of GRNs, the experiment involves altering the gene network in some way, observing the outcome, and using mathematics and logic to infer the underlying principles of the network. It can thus be concluded that the key issues of this process lie in selecting network model and fitting network parameters and structures into the available data.

Neural networks have been used in recognizing patterns in a variety of fields including genetics studies [4,18]. They are biological plausible and noise resistance. A special kind of neural networks, namely recurrent neural networks (RNNs), considers the feedback loops and takes internal states into account. It is able to show the system dynamics of the network over time. With such unique characteristics, this kind of network thus offers an ideal model to work as GRNs. The learning algorithms associated with RNNs can also be used to solve the abovementioned data-fitting problem. Therefore, in this work we establish an approach that takes RNNs to represent GRNs and also exploits the network learning algorithms to reconstruct regulatory systems from collected expression data. In order to deal with the scalability problem, a clustering method with some data analysis techniques for feature extraction is applied to develop GRNs hierarchically. To verify the presented approach, three series of experiments have been conducted to demonstrate how it works.

2. Modeling gene regulatory networks

Models for gene regulatory networks can mainly be categorized into two types that use discrete and continuous variables, respectively [7,8]. The first type of GRN models assumes that genes only exist in discrete states. This approximation is usually implemented by Boolean variables in which the gene is in either on or off state. Boolean networks are easy to simulate in a cheaper computational cost, but it has been proven that Boolean networks are not able to capture some system behaviors that can only be observed on continuous models [7]. Also the dynamics of a Boolean network are deterministic and they depend on the initialization of the node states [15]. These unexpected features make the Boolean network model not realistic. Another popular discrete variable model is Bayesian network that explicitly establishes probabilistic relationships between nodes [9,11,16]. Bayesian models have rich statistics and probability semantics, but learning network structure for such models is computationally expensive. In addition, Bayesian models are inherently static. As the directed network graphs are acyclic by definition, there can be no auto-regulation and no time-course regulation. Some methods have been proposed to solve this problem but the computational complexity will be increased significantly.

Unlike the above discrete variables models, the other type of models uses continuous variables. One of the popular continuous variables models is the one based on differential equations that can describe more accurately the system dynamics of a GRN. Many models based on differential equations have been proposed, including the traditional linear ordinary differential equations and the non-linear power law ones [7]. The well-researched model S-system is a kind of power law model. It consists of a particular set of ordinary differential equations in which the component processes are characterized by power law functions. Compared to the discrete variables models, the differential equations models can more accurately represent the underlying physical phenomena due to its continuous variables. In addition, there are many theories of system analysis and of control design on dynamical systems to support this type of models. Especially, with the non-linear feature of ordinary differential equations (such as S-system), steady-state evaluation, control analysis, and sensitivity analysis of a given system can be established mathematically [23]. But it should be noted that the non-linear ordinary differential equations are hard to solve. It is too difficult for the traditional optimization methods to estimate all the large number of parameters involved in a GRN.

The other commonly used continuous variables model is the neural network-based model, among which the RNNs are the most successful ones [4,22]. This model is biologically plausible and noise resistance. It is continuous in time, and uses a transfer function to transfer the inputs into a shape close to the one observed in natural processes. Also its non-linear characteristics provide information about the principles of control and natural interactions of elements of the modeled system. Furthermore, this model can record and consider its internal states in its operating process, and present the dynamical system behavior over time. It is thus a suitable candidate for modeling time series data.

As is analyzed above, different models have been proposed to simulate GRNs, and computational methods have also been developed to reconstruct networks from the expression data correspondingly. More details can be found in the review work (for example, Refs. [7,8]). From these literatures, it can be seen that work in modeling GRNs shared similar ideas in principle. They improved the performance of network model over iterations and have shown the preliminary results to support the idea of reverse engineering. However, depending on the research motivations behind the work, different researchers explored the same topic from different points of view; thus the implementation details of individual work are different.