Computational Statistics and Data Analysis 53 (2009) 865–876

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Learning Bayesian networks for discrete data

Faming Liang [a,*], Jian Zhang [b]

[a] Department of Statistics, Texas A&M University, College Station, TX 77843-3143, USA
[b] Department of Mathematics, University of York, York, YO10 5DD, UK

ARTICLE INFO

Article history:
Received 6 July 2008
Received in revised form 5 October 2008
Accepted 7 October 2008
Available online 17 October 2008

ABSTRACT

Bayesian networks have received much attention in the recent literature. In this article, we propose an approach to learn Bayesian networks using the stochastic approximation Monte Carlo (SAMC) algorithm. Our approach has two nice features. Firstly, it possesses the self-adjusting mechanism and thus avoids essentially the local-trap problem suffered by conventional MCMC simulation-based approaches in learning Bayesian networks. Secondly, it falls into the class of dynamic importance sampling algorithms; the network features can be inferred by dynamically weighted averaging the samples generated in the learning process, and the resulting estimates can have much lower variation than the single model-based estimates. The numerical results indicate that our approach can mix much faster over the space of Bayesian networks than the conventional MCMC simulation-based approaches.

1. Introduction

The use of graphs to represent statistical models has been one focus of research in recent years. In particular, researchers have directed interest in Bayesian networks and applications of such models to biological data, see e.g., Friedman et al. (2000) and Ellis and Wong (2008). The Bayesian network, as illustrated by Fig. 1, is a directed acyclic graph (DAG) in which the nodes represent the variables in the domain and the edges correspond to direct probabilistic dependencies between them. As indicated by many applications, the Bayesian network is a powerful knowledge representation and reasoning tool under conditions of uncertainty that is typical of real-life applications.

Many approaches have been developed for learning of Bayesian networks in the literature. These approaches can be roughly grouped into three categories: the conditional independence test-based approaches, the optimization-based approaches, and the MCMC simulation-based approaches.

The approaches in the first category perform a qualitative study of dependence relationships between the nodes, and generate a network that represents most of the relationships. The approaches described in Spirtes et al. (1993), Wermuth and Lauritzen (1983) and de Campos and Huete (2000) belong to this category. The networks constructed by these approaches are usually asymptotically correct, but as pointed out by Cooper and Herskovits (1992) that the conditional independence tests with large condition-sets may be unreliable unless the volume of data is enormous. We note that due to limited research resources, the sample size of the biological data is often small, e.g., the gene expression data studied in Friedman et al. (2000) and the real examples studied in this paper.

The approaches in the second category attempt to find a network that optimizes a selected scoring function, which evaluates the fitness of each feasible network to the data. The scoring functions can be formulated based on different principles, such as entropy (Herskovits and Cooper, 1990), the minimum description length (Lam and Bacchus, 1994), and

* Corresponding author. Tel.: +1 979 845 8885; fax: +1 979 845 3144.
E-mail addresses: fliang@stat.tamu.edu (F. Liang), jz538@york.ac.uk (J. Zhang).

0167-9473/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.csda.2008.10.007

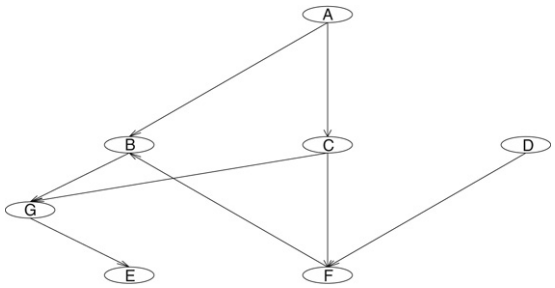

Fig. 1. An example of Bayesian networks.

Bayesian scores (Cooper and Herskovits, 1992; Heckerman et al., 1995). The optimization procedures employed are usually heuristic, such as tabu search (Bouckaert, 1995) and evolutionary computation (de Campos and Huete, 2000; Neil and Korb, 1999). Unfortunately, the task of finding a network structure that optimizes the scoring function is known to be a NP-hard problem (Chickering, 1996). Hence, the optimization process often stops at a local optimal structure.

The approaches in the third category work by simulating a Markov chain over the space of feasible network structures with the stationary distribution being the posterior distribution of the network. The work belonging to this category include Madigan and Raftery (1994), Madigan and York (1995), and Giudici and Green (1999), among others. In these works, the simulation is done using the Metropolis–Hastings (MH) algorithm, and the network features are inferred by averaging over a large number of networks simulated from the posterior distribution. Averaging over different networks can significantly reduce the variation of estimation suffered by the single network-based inference procedure. Although the approaches seem attractive, they can only work well for the problems with a very small number of variables. This is because the energy landscape of the Bayesian network can be quite rugged, with a multitude of local energy minima being separated by high energy barriers, especially when the network size is large. Here, the energy function refers to the negative log-posterior distribution function of the Bayesian network. As known by many researchers, the MH algorithm is prone to get trapped in a local energy minimum indefinitely in simulations from a system for which the energy landscape is rugged. To alleviate this difficulty, Friedman and Koller (2003) introduce a two-stage algorithm: use the MH algorithm to sample a temporal order of the nodes, and then sample a network structure compatible with the given node order. As discussed in Friedman and Koller (2003), for any Bayesian networks, there exists a temporal order of the nodes such that for any two nodes X and Y, if there is an edge from X and Y, then X must be preceding to Y in the order. For example, for the network shown in Fig. 1, a temporal order compatible with the network is ACDFBGE. The two-stage algorithm improves the mixing over the space of network structures, however, the structures sampled by it does not follow the correct posterior distribution, because the temporal order does not induce a partition of the space of network structures. A network may be compatible with more than one order. For example, the network shown in Fig. 1 is compatible with both the orders ACDFBGE and ADCFBGE.

In this article, we propose to learn Bayesian networks using the stochastic approximation Monte Carlo (SAMC) algorithm (Liang et al., 2007). A remarkable feature of the SAMC algorithm is that it possesses the self-adjusting mechanism and is thus less likely trapped by local energy minima. This is very important for learning of Bayesian networks. In addition, SAMC belongs to the class of dynamic weighting algorithms (Wong and Liang, 1997; Liu et al., 2001; Liang, 2002), and the samples generated in the learning process can be used to infer the network features via a dynamically weighted estimator. Like Bayesian model averaging estimators, the dynamically weighted estimator can have much lower variation than the single model-based estimator.

The remainder of this article is organized as follows. In Section 2, we give the formulation of Bayesian networks. In Section 3, we first give a brief review of the SAMC algorithm and then describe its implementation for Bayesian networks. In Section 4, we present the numerical results on a simulated example and two real biological data example. In Section 5, we conclude the paper with a brief discussion.

2. Bayesian networks

A Bayesian network model can be defined as a pair $B = (\mathcal{G}, \boldsymbol{\rho})$, where $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a directed acyclic graph that represents the structure of the network, $\mathcal{V}$ denotes the set of nodes, $\mathcal{E}$ denotes the set of edges, and $\boldsymbol{\rho}$ is a vector of conditional probabilities as described below. For a node $V \in \mathcal{V}$, a parent of V is a node from which there is a directed link to V. The set of parents of V is denoted by $pa(V)$. In this article, we study only the discrete case where V is a categorical variable taking values in a finite set $\{v_1, \ldots, v_{r_i}\}$. There are $q_i = \prod_{V_j \in pa(V_i)} r_j$ possible values for the joint state of the parents of V_i. Each element of $\boldsymbol{\rho}$ represents a conditional probability. For example, ρ_{ijk} is the probability of variable V_i in state j conditioned on that $pa(V_i)$ is in state k. Naturally, $\boldsymbol{\rho}$ is restricted by the constraints $\rho_{ijk} \geq 0$ and $\sum_{j=1}^{r_i} \rho_{ijk} = 1$. The joint distribution of the variables $\boldsymbol{V} = \{V_1, \ldots, V_d\}$ can be specified by the decomposition

$$P(\boldsymbol{V}) = \prod_{i=1}^{d} P\left(V_i | pa(V_i)\right). \tag{1}$$