International Journal of Approximate Reasoning 55 (2014) 1252–1268

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

www.elsevier.com/locate/ijar

Bayesian network approach to multinomial parameter learning using data and expert judgments

Yun Zhou [a,b,*], Norman Fenton [a], Martin Neil [a]

[a] Risk and Information Management (RIM) Research Group, Queen Mary University of London, United Kingdom
[b] Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, PR China

ARTICLE INFO

Article history:
Received 8 May 2013
Received in revised form 17 February 2014
Accepted 26 February 2014
Available online 5 March 2014

Keywords:
Bayesian networks
Multinomial parameter learning
Expert judgments

ABSTRACT

One of the hardest challenges in building a realistic Bayesian Network (BN) model is to construct the node probability tables (NPTs). Even with a fixed predefined model structure and very large amounts of relevant data, machine learning methods do not consistently achieve great accuracy compared to the ground truth when learning the NPT entries (parameters). Hence, it is widely believed that incorporating expert judgments can improve the learning process. We present a multinomial parameter learning method, which can easily incorporate both expert judgments and data during the parameter learning process. This method uses an auxiliary BN model to learn the parameters of a given BN. The auxiliary BN contains continuous variables and the parameter estimation amounts to updating these variables using an iterative discretization technique. The expert judgments are provided in the form of constraints on parameters divided into two categories: linear inequality constraints and approximate equality constraints. The method is evaluated with experiments based on a number of well-known sample BN models (such as *Asia*, *Alarm* and *Hailfinder*) as well as a real-world software defects prediction BN model. Empirically, the new method achieves much greater learning accuracy (compared to both state-of-the-art machine learning techniques and directly competing methods) with much less data. For example, in the software defects BN for a sample size of 20 (which would be considered difficult to collect in practice) when a small number of real expert constraints are provided, our method achieves a level of accuracy in parameter estimation that can only be matched by other methods with much larger sample sizes (320 samples required for the standard machine learning method, and 105 for the directly competing method with constraints).

1. Introduction

Bayesian Networks (BNs) [1,2] are the result of a marriage between graph theory and probability theory, which enable us to model probabilistic and causal relationships for many types of decision-support problems. A BN consists of a directed acyclic graph (DAG) that represents the dependencies among related nodes (variables), together with a set of local probability distributions attached to each node (called a node probability table – NPT – in this paper) that quantify the strengths of these dependencies. BNs have been successfully applied to many real-world problems [3]. However, building realistic and

* Corresponding author at: Risk and Information Management (RIM) Research Group, Queen Mary University of London, United Kingdom.
E-mail addresses: yun.zhou@qmul.ac.uk (Y. Zhou), n.fenton@qmul.ac.uk (N. Fenton), m.neil@qmul.ac.uk (M. Neil).

http://dx.doi.org/10.1016/j.ijar.2014.02.008
0888-613X/

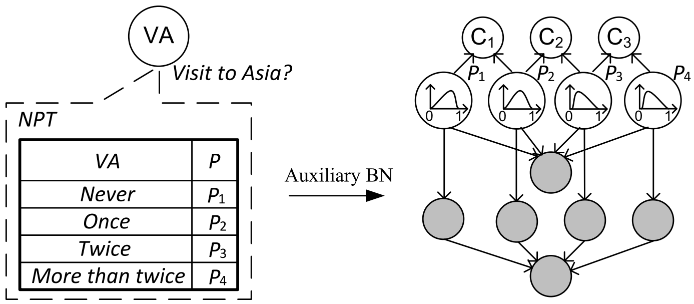

Fig. 1. The overview of parameter learning with constraints and data. The constraints are: C1: $P_1 > P_2$, C2: $P_2 > P_3$ and C3: $P_3 \approx P_4$. The gray color nodes represent part of the MPL model.

accurate BNs (which means building both the DAG and all the NPTs) remains a major challenge. For the purpose of this paper, we assume the DAG model is already determined, and we focus purely on the challenge of building accurate NPTs.

In the absence of any relevant data NPTs have to be constructed from expert judgment alone. Research on this method focuses on the questions of design, bias elimination, judgments elicitation, judgments fusion, etc. (see [4,5] for more details). At the other extreme NPTs can be constructed from data alone, whereby a raw dataset is provided in advance, and statistical based approaches are applied to automatically learn each NPT entry. In this paper we focus on learning NPTs for nodes with a finite set of discrete states. For a node with r_i states and no parents, its NPT is a single column whose r_i cells corresponding to the prior probabilities of the r_i states. Hence, each NPT entry can be viewed as a parameter representing a probability value of a discrete distribution. For a node with parents, the NPT will have q_i columns corresponding to each of the q_i instantiations of the parent node states. Hence, such an NPT will have q_i different r_i-value parameter probability distributions to define or learn. Given sufficient data, these parameters can be learnt, for example using the relative frequencies of the observations [6]. However, many real-world applications have very limited relevant data samples, and in these situations the performance of pure data-driven methods is poor [7]; indeed pure data-driven methods can result in poor results even when there are large datasets [8]. In such situations incorporating expert judgment improves the learning accuracy [9,10].

It is the combination of (limited) data and expert judgment that we focus on in this paper. A key problem is that it is known to be difficult to get experts with domain knowledge to provide explicit (and accurate) probability values. Recent research has shown that experts feel more comfortable providing qualitative judgments and that these are more robust than their numerical assessments [11,12]. In particular, parameter constraints provided by experts can be integrated with existing data samples to improve the learning accuracy. Niculescu [13] and de Campos [14] introduced a constrained convex optimization formulation to tackle this problem. Liao [15] regarded the constraints as penalty functions, and applied the gradient-descent algorithm to search the optimal solution. Chang [16,17] employed constraints and Monte Carlo sampling technology to reconstruct the hyperparameters of *Dirichlet* priors. Corani [18] proposed the learning method for Credal networks, which encodes range constraints of parameters. Khan [19] developed an augmented Bayesian network to refine a bipartite diagnostic BN with constraints elicited from expert's diagnostic sequence. However, Khan's method is restricted to special types of BNs (two-level diagnostic BNs). Most of these methods are based on seeking the global maximum estimation over reduced search spaces.

A major difference between the approach we propose in this paper and previous work is in the way to integrate constraints. We incorporate constraints in a separate, auxiliary BN, which is based on the *multinomial parameter learning* (MPL) model. Our method can easily make use of both the data samples and extended forms of expert judgment in any target BN; unlike Khan's method, our method is applicable to any BN. For demonstration and validation purposes, our experiments (in Section 4) are based on a number of well-known and widely available BN models such as *Asia*, *Alarm* and *Hailfinder*, together with a real-world software defects prediction model.

To illustrate the core idea of our method, consider the simple example of a BN node (without parents) *VA* ("Visit to Asia?") in Fig. 1. This node has four states, namely "Never", "Once", "Twice" and "More than twice"[1] and hence its NPT can be regarded as having four associated parameters P_1, P_2, P_3 and P_4, where each is a probability value of the probability distribution of node *VA*. Whereas an expert may find it too difficult to provide exact prior probabilities for these parameters (for a person entering a chest clinic) they may well be able to provide constraints such as: "$P_1 > P_2$", "$P_2 > P_3$" and "$P_3 \approx P_4$". These constraints look simple, but are very important for parameter learning with small data samples.

Fig. 1 gives an overview of how our method estimates the four parameters with data and constraints (technically we only need to estimate 3 of the parameters since the 4 parameters sum to 1). Firstly, for the NPT column of the target node (dashed callout in Fig. 1), our method generates its auxiliary BN, where each parameter is modeled as a separate continuous node (on scale 0 to 1) and each constraint is modeled as a binary node. The other nodes correspond to data observations for the parameters – the details, along with how to build the auxiliary BN model – are provided in Section 3. It turns out that

[1] In the Asia BN the node "Visit to Asia" actually only has two states. We are using 4 states here simply to illustrate how the method works in general.