International Journal of Approximate Reasoning ••• (••••) •••–•••

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

www.elsevier.com/locate/ijar

Efficient learning of Bayesian networks with bounded tree-width

Siqi Nie [a], Cassio P. de Campos [b], Qiang Ji [a],*

[a] Rensselaer Polytechnic Institute, Troy, NY, USA
[b] Queen's University Belfast, Belfast, UK

ARTICLE INFO

Article history:
Received 16 November 2015
Received in revised form 5 July 2016
Accepted 6 July 2016
Available online xxxx

Keywords:
Bayesian network
Structure learning
Bounded tree-width
Hill climbing

ABSTRACT

Learning Bayesian networks with bounded tree-width has attracted much attention recently, because low tree-width allows exact inference to be performed efficiently. Some existing methods [24,29] tackle the problem by using k-trees to learn the optimal Bayesian network with tree-width up to k. Finding the best k-tree, however, is computationally intractable. In this paper, we propose a sampling method to efficiently find representative k-trees by introducing an informative score function to characterize the quality of a k-tree. To further improve the quality of the k-trees, we propose a probabilistic hill climbing approach that locally refines the sampled k-trees. The proposed algorithm can efficiently learn a quality Bayesian network with tree-width at most k. Experimental results demonstrate that our approach is more computationally efficient than the exact methods with comparable accuracy, and outperforms most existing approximate methods.

1. Introduction

Bayesian networks (BNs) use a directed acyclic graph (DAG) to compactly represent the joint probability distribution for multiple variables. The DAG encodes conditional independencies which reduces the number of parameters. Learning BNs from data has been widely studied for decades. In this paper we present our approach of score-based BN structure learning with some special constraint.

The inference problems in BNs, such as querying the probability of some value of a variable conditioned on a configuration of some other variables (belief updating), or finding a configuration of the variables that maximizes the joint probability (MAP inference), are NP-hard to compute exactly [13] or even approximately [15,30]. Existing exact algorithms have worst-case time complexity exponential in the tree-width of the graph [16,20,23,26]. Therefore, for any application that requires fast inferences, it is important to learn networks with small tree-width. Learning a BN with bounded tree-width has received growing attention recently. Besides guaranteed inference complexity, by imposing a hard constraint on the tree-width of the structure, selecting an over-complicated structure is prevented, thus the chance of over-fitting is reduced. Some empirical results [18] demonstrate that bounding the tree-width of a BN achieves better generalization performance.

Several algorithms have been proposed to learn BNs with bounded tree-width. Elidan and Gould [19] designed an approximate algorithm by combining several heuristics to compute the tree-width and to learn the structure of BNs. Korhonen and Parviainen [24] proposed a dynamic programming based algorithm for learning n-node BNs of tree-width at most k

* Corresponding author.
E-mail addresses: nies@rpi.edu (S. Nie), c.decampos@qub.ac.uk (C.P. de Campos), qji@ecse.rpi.edu (Q. Ji).

http://dx.doi.org/10.1016/j.ijar.2016.07.002
0888-613X/© 2016 Elsevier Inc. All rights reserved.

(which we denote as K&P algorithm in this paper). Their algorithm guarantees to find the optimal structure over n nodes maximizing a given score function subject to the tree-width constraint with complexity $O(3^n n^{k+O(1)})$. In practice, it is quite slow for networks with more than 15 nodes, or tree-width more than 3. Parviainen et al. [31] developed an integer programming approach to solve the problem. It iteratively creates a cutting plane on the current solution to avoid exponentially many constraints. Berg et al. [6] transferred the problem into a weighted maximum satisfiability problem and solved it by weighted MAX-SAT solvers. However, all the exact algorithms work only with small networks and small tree-widths. We introduced an exact algorithm based on mixed integer linear programming (MILP) [29] and approximate methods based on k-tree sampling [29,27] to address this problem. In our latest work [28], we further improved the sampling method using the A* search algorithm. Methods have also been proposed to tackle the problem of learning undirected models with bounded tree-width [2,10,35].

In this work, we present a novel method of score-based BN structure learning with bounded tree-width. We design an approximate approach based on sampling k-trees, which are the maximal graphs of tree-width k. The sampling method is based on a fast bijection between k-trees and Dandelion codes [9]. We design a sampling scheme, called *distance preferable sampling* (DPS), in order to effectively cover the space of k-trees using limited samples, in which we give a larger probability for a sample in the unexplored area of the space, based on the existing samples. Smart rules to explore the sample space are essential, because we can only compute a few best structures respecting sampled k-trees in a reasonable amount of time. To evaluate the sampled k-trees, we design an *informative score* (I-score) function to measure the quality of k-trees based on independence tests and BDeu scores. Different from the method proposed in [29], this work focuses on identifying high quality k-trees, instead of uniformly sampling. For each sampled k-tree (represented by a Dandelion code), we first refine it by employing a hill climbing algorithm (HC) to locally identify a code with the largest I-score. One shortcoming of the HC method is that it ends up with a local optimum. To alleviate this issue, we introduce a probabilistic version of the hill climbing method (PHC) to obtain a k-tree of high quality. Once a k-tree is found, both exact [24] and approximate [29] methods are implemented to find the BN as a subgraph of the k-tree.

This paper is structured as follows. We first introduce some definitions and notations for BNs and tree-width in Section 2. Then we discuss the proposed sampling method for learning BNs with bounded tree-width in Section 3. Experimental results are given in Section 4. Finally we conclude the paper in Section 5.

2. Preliminaries

2.1. Learning Bayesian networks

A Bayesian network uses a directed acyclic graph (DAG) to represent a set of random variables $X = \{X_i\}_{i=1}^n$ and their conditional (in)dependencies. Arcs of the DAG encode parent–child relations. Denote X_{pa_i} as the parent set of variable X_i. Conditional probability tables $p(x_i|x_{pa_i})$ are given accordingly, where x_i and x_{pa_i} are instantiations of X_i and X_{pa_i}. We consider categorical variables in this work.

Given a fixed structure G and a complete data set $\mathcal{D} = \{x^{(j)}\}_{j=1}^M$ of points assumed sampled independently from a distribution P on $\mathcal{X}$, the numerical parameters θ of a BN with structure G can be efficiently obtained by maximum likelihood estimation (MLE) by finding θ that maximizes the data log-likelihood according to the model:

$$\mathcal{L}(G,\theta) = \sum_{j=1}^{M} \log P_{G,\theta}(x^{(j)}) \,. \tag{1}$$

The structure learning task of BNs is to identify the "best" DAG from data. In this paper we consider the score-based BN structure learning problem, in which a score $s(G)$ is assigned to each DAG G. The commonly used score functions (such as BIC [33] and BDeu [8,14,22]) are decomposable, i.e., the overall score can be written as the sum of local score functions,

$$s(G) = \sum_{i=1}^{n} s_i(X_{pa_i}) \,. \tag{2}$$

For each variable, its score is only related to its parent set. We assume that local scores have been computed in advance and can be retrieved in constant time.

Most score functions penalize model complexity, because increasing the number of parents of a variable never decreases data likelihood, which leads to overfitting and poor generalization. A typical example is the BIC score,

$$s_i(x_{pa_i}) = \mathcal{L}_{i,pa_i} - t_i(x_{pa_i}) \cdot w \,, \tag{3}$$

where the first term is log-likelihood and the second term is a penalty term to avoid overfitting. $t_i(pa_i)$ is the number of free parameters with respect to the configuration of $\{x_i, x_{pa_i}\}$, and $w = \frac{1}{2}\log M$, where M is the number of data samples. Such penalty term generally leads to structures of small in-degree, but even small in-degree graphs can have large tree-width, which is a problem for subsequent probabilistic inferences with the model. An example is the directed square grid, which has tree-width linear on the number of nodes in the diagonal of the grid, but maximum in-degree fixed in two. A more effective way to constrain the sparsity of the structure is to use a bound for the tree-width, a commonly used measure of the complexity of a graph.