Postprint: Bayesian Network Structure Learning Based on Causal Effects

Authors: tranquility, Teng Yue, Yang Jiaoyun, Li Lian

Date: 2018-09-12T00:00:00+00:00

Abstract

Learning Bayesian network structure from data is an NP-hard problem, and improving the accuracy of network structure learning algorithms represents a major research challenge. Based on Judea Pearl' s causal theory, we propose a Bayesian network structure learning method that enhances the accuracy of existing algorithms. This method employs improved Pearl causal effects and BDe scoring to learn the priority ordering of network nodes, utilizes the K2 algorithm to learn the initial network structure, and corrects the learning results through BDe score reverse adjustment, mutual information, and BDe score-based edge deletion. Experiments were conducted on standard Bayesian network datasets ASIA and ALARM. Across 20 experimental groups with sample sizes ranging from 2000 to 20000, the learning accuracy improved by an average of 16% compared to the MMHC algorithm, while the standard deviation of accuracy decreased by an average of 17% relative to the MMHC algorithm. Experimental results demonstrate that the causal effect-based method exhibits superior performance compared to the MMHC algorithm.

Full Text

Preamble

Bayesian Network Structure Learning Method Based on Causal Effect

An Ning, Teng Yue, Yang Jiaoyun†, Li Lian

(National Smart Eldercare International S&T Cooperation Base, School of Computer & Information, Hefei University of Technology, Hefei 230009, China)

Abstract: Learning Bayesian networks from data is an NP-hard problem. Improving the accuracy of network structure learning algorithms remains a critical research challenge. Based on Judea Pearl' s causal theory, this paper proposes a Bayesian network structure learning method that enhances the accuracy of

existing algorithms. The method utilizes an improved Pearl causal effect and BDe scoring to learn node ordering, applies the K2 algorithm to learn an initial network, and refines the result through BDe score-based reverse adjustment, mutual information, and BDe score-guided edge deletion. Experiments on standard Bayesian network datasets ASIA and ALARM demonstrate that across 20 experimental groups with sample sizes ranging from 2,000 to 20,000, the learning accuracy improves by an average of 16% compared to the MMHC algorithm, while the standard deviation of accuracy decreases by an average of 17%. The results indicate that the causal effect-based method achieves superior performance compared to the MMHC algorithm.

Keywords: Bayesian network; Alzheimer' s disease; K2 algorithm; causal effect; BDe scoring; mutual information

0 Introduction

Bayesian networks are essential tools for representing uncertain knowledge, representing joint probability distributions over variables as directed acyclic graphs (DAGs). A Bayesian network consists of two components: a directed acyclic graph and conditional probability tables. The DAG qualitatively represents independence relationships among variables, while the conditional probability tables quantitatively represent the degree of dependency between variables. Due to their graphical visualization capabilities, Bayesian networks are widely applied in biomedicine, prediction, classification, causal inference, visual recognition, and information retrieval.

Construction methods for Bayesian networks primarily fall into two categories: expert knowledge-based construction and data-driven construction. Building Bayesian networks through expert knowledge is extremely tedious and error-prone, whereas learning them from data is an NP-hard problem. Consequently, developing efficient and high-quality methods for learning Bayesian networks from data has become a major research focus. Over recent decades, numerous structure learning algorithms have emerged, including those based on conditional independence tests, scoring-based search algorithms, and hybrid methods combining conditional independence testing with scoring search.

Scoring-based methods comprise two components: (a) scoring functions that measure the goodness-of-fit between the network and data, such as the Bayesian Information Criterion (BIC) score, which uses log-likelihood to measure fit under the i.i.d. assumption; the BDe score, which assumes a Dirichlet prior distribution over structures; the CH score (K2 scoring function), which assumes a uniform prior distribution; and the MDL score based on minimum description length; and (b) search methods for finding the highest-scoring network, such as the K2 algorithm requiring node ordering, max-min hill-climbing, particle swarm optimization for better global optima, and hill-climbing algorithms. Since the

search space grows exponentially with the number of nodes, exhaustive search for the highest-scoring network is infeasible.

While conditional independence test-based methods can accurately learn Bayesian network structures, they are only suitable for discrete data and exhibit poor efficiency in high-dimensional settings. Hybrid methods combining conditional independence testing with scoring search achieve higher accuracy, with the MMHC algorithm being the most representative. The MMHC algorithm consists of two phases: the MMPC algorithm in the first stage and a scoring method in the second stage. The MMPC algorithm first identifies the candidate parent-child set CPC(T) for target variable T, then computes the minimum dependency degree Assoc(X,T|CPC(T)) for other variables relative to T, adding variables with maximum association to the candidate set until MaxAssoc(X,T|CPC(T)) becomes zero. In the second stage, MMHC uses three operators—deletion, addition, and reversal—combined with scoring functions to obtain the final network structure.

The K2 algorithm is a greedy search algorithm that significantly reduces the search space but requires a predefined node ordering (where earlier nodes cannot be children of later nodes). The correctness of this ordering directly affects the learning outcome. Recent research on K2 node ordering has yet to meet the demands of big data applications in terms of time complexity and accuracy. The proposed causal effect-based approach draws from Judea Pearl' s causal theory, defines causal effect strength calculations and BDe scoring for empty networks to obtain node ordering, and combines this with the K2 algorithm and mutual information to learn Bayesian networks. Experiments on standard databases demonstrate that this method significantly outperforms existing approaches in accuracy and standard deviation.

1.1 Bayesian Networks

A Bayesian network is a parameterized directed acyclic graph, denoted as $\langle \mathcal{G}, \Theta \rangle$, where $\mathcal{G}$ represents the directed acyclic graph (as shown in [Figure 1: see original paper]) and Θ represents the conditional probability tables of parent nodes for child nodes. The graph $\mathcal{G}$ is a tuple $\langle V, E \rangle$, where V is the set of all random variables, $V = \{X_1, X_2, ..., X_n\}$, X_i is the i-th node in $\mathcal{G}$, and E is the set of edges. According to graph theory, the edge set can be represented by matrix A, where $A_{ij} = 1$ if a directed edge exists from X_i to X_j, and $A_{ij} = 0$ otherwise. In this acyclic graph, nodes represent variables, and edges represent dependency relationships between variables. As shown in [Figure 1: see original paper], if an edge $X_i \to X_j$ exists for $i, j \in \{1, 2, ..., n\}$, then X_i is a parent of X_j. Based on the Markov assumption, the joint probability distribution can be expressed as:

$$P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i|\pi(X_i))$$

where $\pi(X_i)$ denotes all parent nodes of X_i.

1.2 Causal Theory

Traditional statistics uses correlation to describe relationships between random variables, but correlation cannot fully capture these relationships. For example, while we can easily establish the association between "wind blowing" and "leaves shaking," and 常识 tells us that wind causes leaves to shake (wind → leaves shaking), reversing this direction would incorrectly suggest that leaves shaking causes wind. Such causal relationships exist within associations. Judea Pearl proposed causal theory to address these issues, formalizing causal strength using average causal effect (ACE). More formally, the ACE between two variables X and Y is described by:

$$ACE = P(Y = 1|do(X = 1)) - P(Y = 1|do(X = 0))$$

where $do(\cdot)$ represents Pearl' s "do-operator." The formal definition of the do-operator is given by:

$$P(y|do(x)) = \sum_{z} P(y|x, z)P(z)$$

where Z is the set of backdoor paths from X to Y.

---

2 Bayesian Network Structure Learning Method

Since the K2 algorithm requires node ordering that only qualitatively describes parent-child relationships, the proposed method first defines node priority to quantitatively characterize these relationships. Section 2.1 defines the node priority calculation method, which sorts the node priority vector in descending order to obtain the node ordering. The second step uses this ordering with the K2 algorithm to initialize the Bayesian network. To improve computational efficiency, the node priority calculation does not consider backdoor paths between nodes. While this yields correct ordering in most cases, some node orders may deviate from the true ordering, resulting in reversed or extra edges in the initial network. Therefore, the third step systematically reverses and deletes edges to find networks with higher BDe scores than the initial network. To avoid local optima when using score-based deletion and reversal, a mutual information-based

edge deletion strategy removes some redundant edges, allowing the score to restart from a new starting point closer to the global optimum. The algorithm terminates when the score no longer improves.

2.1 Node Priority and Node Ordering

Definition 1 (Node Priority). For any nodes $X_i, X_j \in V$, where V is the set of nodes in the Bayesian network and n is the number of nodes, given a criterion S, if there exist N nodes $\{X_1, X_2, ..., X_N\}$ such that criterion S holds, then N is the priority of node X_i.

The node ordering includes two algorithms that compute the node priority vector from improved causal effect and empty-network BDe scoring, respectively. Sorting this vector in descending order yields the node ordering. More formally:

For node set V and node X_i, given a judgment criterion S, if there exist N nodes $\{X_{k_1}, X_{k_2}, ..., X_{k_N}\}$ such that criterion S holds, then N is the priority of node X_i.

It should be noted that if nodes X and Y have identical priority values in the node priority vector, the node ordering between them is determined using the current priority criterion applied only to these two nodes.

2.1.1 Node Priority Algorithm Based on Causal Effect

The causal effect between any two nodes X_i and X_j in the dataset is approximated by:

$$CE_{X_i \to X_j} = \frac{N(X_i = 1, X_j = 1)}{N(X_i = 1)} - \frac{N(X_i = 0, X_j = 1)}{N(X_i = 0)}$$

where $N(\cdot)$ represents the sample count and N denotes the total sample size.

This formulation extends Pearl' s causal effect by considering both $X_j = 1$ and $X_j = 0$ cases, as the original ACE only considered $Y = 1$. Using causal effect as the judgment criterion, the algorithm starts from a node in the network and sequentially computes its causal effect on other nodes. If $CE_{X_i \to X_j} > CE_{X_j \to X_i}$, the priority of X_i increases by one; otherwise, the priority of X_j increases by one. The algorithm traverses all pairs of nodes, performing $\frac{N(N-1)}{2}$ calculations for N nodes. The node priority vector is then sorted in descending order to obtain the node ordering.

2.1.2 Node Priority Method Based on BDe Scoring Function

The BDe scoring function, one of the earliest metrics for evaluating Bayesian network fit, assumes data follows a Dirichlet distribution:

$$score_{BDe}(G|D) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + m_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + m_{ijk})}{\Gamma(\alpha_{ijk})}$$

where $\Gamma(\cdot)$ is the gamma function, m_{ijk} represents the count of samples where node i takes its k-th value and its parents take their j-th configuration, $m_{ij} = \sum_k m_{ijk}$, and α_{ijk} are hyperparameters of the Dirichlet distribution.

Using the BDe function as the node priority criterion, the algorithm starts from a network with only nodes. For all node pairs $\{X_i, X_j\} \subseteq \{X_1, X_2, ..., X_n\}$, it constructs two graphs: G_1 where X_i points to X_j, and G_2 where X_j points to X_i. If $score_{BDe}(G_1|D) > score_{BDe}(G_2|D)$, the priority of X_i increases by one; otherwise, the priority of X_j increases by one. This yields the node priority vector based on BDe scoring.

2.3.1 Edge Deletion Strategy Based on Mutual Information

In most cases, the initialized network approximates the true Bayesian network. However, slight deviations in node ordering may introduce incorrect edges. The scoring adjustment strategy in Section 2.3.2 operates on specific edges under particular conditions, making it prone to local optima. To approach the global optimum and reduce this risk, mutual information is used to delete edges with low correlation (which do not exist in the true network but may have higher scores in the current state), enabling the scoring process to restart from a better position.

Mutual information measures the correlation between variables:

$$I(X;Y) = H(X) - H(X|Y)$$

where $H(X)$ is the entropy of variable X and $H(X|Y)$ is the conditional entropy of X given Y. The entropy is defined as:

$$H(X) = -\sum_{x} P(x) \log P(x)$$

Similarly, the conditional entropy is:

$$H(X|Y) = -\sum_{x,y} P(x,y) \log P(x|y)$$

Entropy describes the uncertainty of information in a variable. Using mutual information instead of conditional independence tests effectively improves computational efficiency. The deletion strategy sorts all pairwise mutual information in descending order. For each target node T, it selects the top h nodes as potential neighbors. For each edge between T and neighbor Y in the initial network, it checks: (a) whether T appears in Y' s top-h mutual information list, and (b) whether Y appears in T' s top-h list. If neither condition holds, the edge between T and Y is deleted. To avoid incorrectly deleting true edges or failing to delete redundant ones, h is set based on network size. In experiments, $h = \lceil \sqrt{n} \rceil$, where n is the number of nodes.

2.3.2 Reverse Adjustment and Edge Deletion Based on BDe Score

After completing Section 2.3.1, most low-correlation edges have been removed, bringing the network score closer to the global optimum. This stage further adjusts reversed edges caused by node ordering errors.

Using BDe score as the adjustment criterion, two operators are applied: edge reversal and edge deletion. For each edge $e_i \in \{e_1, e_2, ..., e_n\}$ in the network, the algorithm sequentially reverses and deletes the edge, recalculating the BDe score. If the score improves, the reversal or deletion is retained. This process continues until the score no longer increases.

---

3 Experimental Results

The causal effect-based method was tested on two standard Bayesian network datasets (ASIA and ALARM) and compared against MMHC, MCMC, hill-climbing, and random K2 algorithms. For each dataset, experiments compared average and standard deviation metrics for correct edges, missing edges, reversed edges, and extra edges across different sample sizes (2,000, 4,000, 6,000, 8,000, 10,000, 12,000, 14,000, 16,000, and 20,000). The results demonstrate that the proposed method outperforms other Bayesian network learning approaches in both correct edge counts and error metrics.

3.1 ASIA Database

The ASIA database is a standard Bayesian network dataset describing a chest diagnosis network, as shown in [Figure 2: see original paper]. Experimental results are presented in and [Figure 3: see original paper]-[Figure 4: see original paper].

The causal effect-based method achieved an average of 7.0 correct edges across 10 experimental groups, compared to 6.3 for MMHC—a 11.1% improvement.

[Figure 3: see original paper] shows correct edge counts across different sample sizes, with MMHC being the strongest comparative algorithm. [Figure 4: see original paper] illustrates the incremental improvement over MMHC (horizontal axis values correspond to [Figure 3: see original paper]). Notably, the proposed method reached the global optimum (all 8 correct edges) in four of ten experiments, while MMHC achieved this only once, indicating superior convergence to the true network structure.

3.2 ALARM Network

The ALARM network, derived from a medical diagnostic monitoring system, comprises 37 nodes and 46 edges, as shown in [Figure 5: see original paper]. Using the BDe score-based node ordering method, the causal effect-based approach achieved an average of 41.2 correct edges across ten datasets, compared to 34.1 for MMHC—representing a 20.8% accuracy improvement. Combined with ASIA results, the average accuracy improvement across 20 experimental groups is 16% over MMHC.

[Figure 6: see original paper] compares correct edge counts across sample sizes, with MMHC outperforming other baselines. [Figure 7: see original paper] shows the incremental improvement over MMHC (horizontal axis values correspond to [Figure 6: see original paper]). The proposed method outperformed MMHC in eight of ten ALARM datasets, with better performance in missing and reversed edges. However, due to the K2 initialization and the edge deletion strategy' s limitation of deleting at most one edge at a time (for efficiency), some redundant edges may remain, potentially causing local optima.

[Figure 8: see original paper] presents standard deviation comparisons across 20 experimental runs. The causal effect-based method reduces the average standard deviation of correct edges by 17% compared to MMHC. The difference is less pronounced in ASIA (8 nodes, 8 edges) where learning difficulty is lower, but more significant in ALARM (46 edges) where the risk of local optima is higher. While random K2 shows the lowest standard deviation, [Figure 9: see original paper] reveals that its accuracy is substantially lower, as it tends to produce nearly fully connected networks with limited practical value when node ordering is significantly incorrect.

[Figure 9: see original paper] compares accuracy across methods, confirming that the causal effect-based approach achieves superior accuracy and stability, outperforming MMHC on ASIA and substantially exceeding MMHC, MCMC, and hill-climbing on the larger ALARM network.

---

4 Conclusion

Bayesian networks are graphical models for describing joint probability distributions over nodes, where arrows can represent causal relationships. This paper proposes node priority algorithms based on Pearl' s causal effect and BDe scoring functions, combined with the K2 algorithm, demonstrating superior performance over MMHC and other algorithms on standard datasets. The causal effect-based method adjusts reversed edges using scoring functions. Current scoring functions assume data follows specific prior distributions, which may not accurately capture network-data fit. Future research could leverage text mining and natural language processing to identify causal directions from keyword co-occurrence sentences in text, providing a promising direction for refining edge directions learned from data.

References

[1] Seixas F L, Zadrozny B, Laks J, et al. A Bayesian network decision model for supporting the diagnosis of dementia, Alzheimer' s disease and mild cognitive impairment [J]. Computers in Biology and Medicine, 2014, 51(7): 140-158.

[2] Ramazzotti D, Graudenzi A, Antoniotti M. Modeling cumulative biological phenomena with Suppes-Bayes causal networks [J/OL]. Evolutionary Bioinformatics Online, 2018, 14. (2016-02-25) [2018-02-26]. http://dx.doi.org/10.1101/041343.

[3] Fenton N, Neil M, Marquez D. Using Bayesian networks to predict software defects and reliability [J]. Journal of Risk & Reliability, 2017, 222(222): 263-292.

[4] 金杉, 崔文, 金志刚. 正态分布的贝叶斯网络火灾数据融合预警研究 [J]. 计算机应用研究, 2016, 33(5): 1473-1476.

[5] Sein M, 傅顺开, 吕天依, 等. 一般贝叶斯网络分类器及其学习算法 [J]. 计算机应用研究, 2016, 33(5): 1327-1334.

[6] He Lianghua, Hyu Die, Wan Meng, et al. Common Bayesian network for classification of EEG-Based multiclass motor imagery BCI [J]. IEEE Trans on Systems Man & Cybernetics Systems, 2017, 46(6): 843-854.

[7] 高瑞, 王双成, 杜瑞杰. 企业运行指标因果分析的动态贝叶斯网络方法 [J]. 计算机应用研究, 2016, 33(5): 1433-1436.

[8] Kaiser J L, Bland C L, Ii D J K. Identifying causal networks linking cancer processes and anti-tumor immunity using Bayesian network inference and metagene constructs [J]. Biotechnol Prog, 2016, 32(2): 470-479.

[9] Klinger T, Rottensteiner F, Heipke C. A dynamic Bayes network for visual pedestrian tracking [C]//ISPRS-International Archives of the Photogrammetry: Remote Sensing and Spatial Information Sciences, 2014, XL-3(3): 145-150.

[10] 徐建民, 唐万生, 陈振亚. 贝叶斯网络在信息检索中的应用 [J]. 河北大学学报: 自然科学版, 2007, 27(1): 93-98.

[11] Chickering D M, Heckerman D, Meek C. Large-sample learning of Bayesian networks is NP-Hard [M]. Brookline: JMLR.org, 2004.

[12] Zhao bo, Wyu Qingchang, Yin Shitang, et al. An improved Bayesian network structure learning algorithm based on the conditional independence test [J]. Journal of Yunnan University of Nationalities, 2011, 20(5).

[13] Geng Zhi, Wang Chi, Zhao Qiang. Decomposition of search for v-structures in DAGs [J]. Journal of Multivariate Analysis, 2005, 96(2): 282-294.

[14] Lam W, Bacchus F. Learning Bayesian belief networks: an approach based on the MDL principle [J]. Computational Intelligence, 1994, 10(3): 269-293.

[15] Cooper G F, Herskovits E. A Bayesian method for the induction of probabilistic networks from data [J]. Machine Learning, 1992, 9(4): 309-347.

[16] Li Guoling, Xing Lining, Zhang Zhongshan, et al. A new Bayesian network structure learning algorithm mechanism based on the decomposability of scoring functions [J]. IEICE Trans on Fundamentals of Electronics Communications & Computer Sciences, 2017, 100(7): 1541-1551.

[17] Chen Xuewen, Anantha G, Lin Xiantong. Improving Bayesian network structure learning with mutual information-based node ordering in the K2 algorithm [J]. IEEE Trans on Knowledge & Data Engineering, 2008, 20(5): 628-640.

[18] 张连文. 贝叶斯网引论 [M]. 北京: 科学出版社, 2006.

[19] Acid S, Campos L M D. Searching for Bayesian network structures in the space of restricted acyclic partially directed graphs [M]. USC: AI Access Foundation, 2003.

[20] Liu Xuqing, Liu Xinsheng. Structure learning of Bayesian networks by continuous particle swarm optimization algorithms [J]. Journal of Statistical Computation & Simulation, 2018, 88(9): 1-29.

[21] Lerner B, Malka R. Investigation of the K2 algorithm in learning bayesian network classifiers [J]. Applied Artificial Intelligence, 2011, 25(1): 74-96.

[22] Cooper G F, Herskovits E. A Bayesian method for the induction of probabilistic networks from data [J]. Machine Learning, 1992, 9(4): 309-347.

[23] Tsamardinos L, Brown L E, Aliferis C F, et al. The max-min hill-climbing Bayesian network structure learning algorithm [J]. Machine Learning, 2006, 65(1): 31-78.

[24] Pearl J, Glymour M, Jewell N P. Causal inference in statistics: a primer [M]. Hoboken: Wiley, 2016: 53-87.

Note: Figure translations are in progress. See original paper for figures.