A Study on Performance Comparison of Structural Learning Algorithms about Synthetic Pattern of Bayesian Network

Jae-seong Yoo[1], Baek-woon Yu[2] and Rae-hyun Chang[2]

[1]Department of Computer Science and Engineering, Korea University, Republic of Korea
[2]Department of Brain and Cognitive Engineering, Korea University, Republic of Korea

Abstract. Bayesian networks(BN) are graphical models where nodes represent random variables and arrows represent probabilistic dependencies between them. And structure learning refers to a method of finding the structure when only know about variables. Many algorithms have been developed for this. Therefore, the purpose of these algorithms is to find the groundtruth correctly.
The Bayesian network can have many different shapes. In particular, as the number of nodes increases, the number of possible shapes increases exponentially. So should we measure performance for all shapes? It seems impossible to measure performance for all shapes.
However, if we have a somewhat stereotypical shapes, we will be able to do that. At least, we will be able to do these shapes. It is our goal to do this minimal thing.

1 The Basics of Bayesian Network Structure Learning

1.1 Bayesian Network

Bayesian networks(BN) are graphical models where nodes represent random variables and arrows represent probabilistic dependencies between them (Kevin B. K. and Ann E. N., 2010).

A n-dimensional BN is a triple $B = (X, G, \Theta)$ where:

- X is a n-dimensional finite random vector each random variable X_i ranges over by a finite domain D_i. Henceforward, we denote the joint domain by $D = \prod_{i=1}^{n} D_i$
- $G = (N, E)$ is a directed acyclic graph(DAG) with nodes $N = \{X_1, \cdots, X_n\}$ and edges E representing direct dependencies between the variables.
- Θ encodes the parameters $\{\theta_{ijk}\}_{i\in 1,\cdots,n,\, j\in D_{\Pi_{X_i}},\, k\in D_i}$ of the network, where

$$\theta_{ijk} = P_B\left(X_i = x_{ik} \mid \prod_{X_i} = \omega_{ij}\right),$$

Π_{X_i} denotes the set of parents of X_i in G, $D_{\Pi_{X_i}}$ denotes the joint domain of the variables in Π_{X_i}, x_{ik} is the k-th value of X_i and ω_{ij} is the j-th configuration of Π_{X_i}.

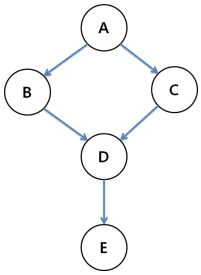

Fig. 1. $P(A, B, C, D, E) = P(A)P(B|A)P(C|A)P(D|B, C)P(E|D)$

A BN defines a unique joint probability distribution over X given by

$$P_B(X_1, \cdots, X_n) = \prod_{i=1}^{n} P_B\left(X_i \mid \prod_{X_i}\right).$$

- A BN encodes the independence assumptions over the component random variables of X.
- An edge (j, i) in E represents a direct dependency of X_i from X_j.
- The set of all Bayesian networks with n variables is denotes by B_n.

1.2 Bayesian Network Structure Learning

The problem of learning a BN given data T consists of finding the BN that best fits the data T. In order to quantify the fitting of a BN a scoring function ϕ is considered.

Learning a Bayesian network is as follows:

Given a data $T = \{y_1, \cdots, y_n\}$ and a scoring function ϕ, the problem of learning a Bayesian network is to find a Bayesian network $B \in B_n$ that maximizes the value $\phi(B, T)$. (Margaritis D., 2003, Marco S., 2010)

1.2.1 Constraint-based algorithms These algorithms learn the network structure by analyzing the probabilistic relations entailed by the Markov property of Bayesian networks with conditional independence tests and then constructing a graph which satisfies the corresponding d-separation statements. The resulting models are often interpreted as causal models even when learned from observational data (Pearl J. 1988).

Grow-Shrink (GS) based on the Grow-Shrink Markov Blanket, the first (and simplest) Markov blanket detection algorithm used in a structure learning algorithm.

Incremental Association (IAMB) based on the Markov blanket detection algorithm of the same name, which is based on a two-phase selection scheme (a forward selection followed by an attempt to remove false positives).

1.2.2 Score-based algorithms The main idea behind score-based learning is to optimize the degree of match between the generated network and the observations. (Benjamin

B. P., 2003) These algorithms assign a score to each candidate Bayesian network and try to maximize it with some heuristic search algorithm. The search problem of identifying a Bayesian network that has a relative posterior probability greater than a given constant is NP-complete. (D.M. Chickering, 1996) Greedy search algorithms (such as hill-climbing or TABU search) are a common choice, but almost any kind of search procedure can be used.

Hill-Climbing (HC) a hill climbing greedy search on the space of the directed graphs. The optimized implementation uses score caching, score decomposability and score equivalence to reduce the number of duplicated tests.

Tabu Search (TABU) a modified hill climbing able to escape local optima by selecting a network that minimally decreases the score function.

1.2.3 Hybrid algorithms Hybrid algorithms are combines the constraint-based and score-based.

Max-Min Hill-Climbing (MHHC) a hybrid algorithm which combines the Max-Min Parents and Children algorithm (to restrict the search space) and the Hill-Climbing algorithm (to find the optimal network structure in the restricted space).

Restricted Maximization (RSMAX2) a more general implementation of the Max-Min Hill-Climbing, which can use any combination of constraint-based and score-based algorithms.

2 Varying topologies and number of nodes

According to Robinson, R. W. (1973), the number of DAGs may be computed by the recurrence relation as shown below.

$$a_n = \sum_{k=1}^{n} (-1)^{k-1} \binom{n}{k} 2^{k(n-k)} a_{n-k}$$

In other words, the number of DAGs on n labeled vertices, for n = 0, 1, 2, 3, (without restrictions on the order in which these numbers appear in a topological ordering of the DAG) is 1, 1, 3, 25, 543, 29281, 3781503, $\cdots$. These exponentially increasing numbers makes difficulties to check performance of each algorithms for structure learning.

Meanwhile, Eitel J. M. L., (2008) proposed a formal DAG model as **Fig. 2.**.

If one node has plurality of parent nodes, then this form called Collapse.

Multiple node bite the tail of the tail, then this form called Line.

If one node has plurality of child node, then this form called Star.

The description of the Pseudo Loop is a little complicated. At first, drew a line form. And next, root node has depended on the very last child node. Then it called Pseudo Loop. Actually loop does not have, it looks like a loop at glance.

A part of the top, one node has plurality of child node like Star form. And the bottom part, one node has plurality of parent node like Collapse form. If it connected, then it called Diamond.

If two nodes has plurality of child node together, then it called Rhombus.

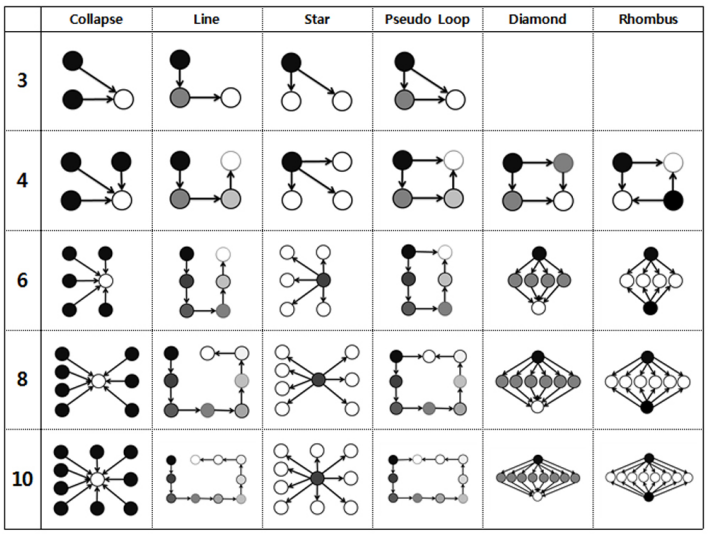

Fig. 2. Bayesian Networks with varying topologies and number of nodes

Although it is difficult to evaluate the performance of the algorithm for all cases of the DAG model according to the number of nodes, but at least when the ground truth equals to **Fig. 2.**, we can compare.

The goal of our team is to compare the performance of the structure learning algorithm based on the above synthetic model and to confirm whether each algorithm learns the pattern of the synthetic model well.

3 The Methodology of Performance Comparison

3.1 The Number of Graphical Errors in Learnt Structure

The comparison methodology used in this paper is similar to the method used in X.-w. Chen *et al.* (2006). The existence of the known network structures allows us to define three important terms which indicate the performance of the algorithm (in terms of the number of graphical errors in the learnt structure).

C (Correct Arcs) Edges present in the original network and in the learnt network structure.

M (Missing Arcs) Edges present in the original network but not in the learnt network structure.

WO (Wrongly Oriented Arcs) Edges present in the learnt network structure, but having opposite orientation when compared with the corresponding edge in the original network structure.

WC (Wrongly Corrected Arcs) Edges not present in the original network but included in the learnt network structure.

		Target Network	Learnt Network	Direction
C	(Correct Arcs)	exist	exist	correct
M	(Missing Arcs)	exist	not exist	
WO	(Wrongly Oriented Arcs)	exist	exist	wrong
WC	(Wrongly Corrected Arcs)	not exist	exist	

3.2 Network Scores

The values of the BDe, the Log-likelihood (LL), the AIC, and the BIC are metrics for the learned networks. (Alexandra M. C., 2009) These measures can offer an idea of the quality of the networks from different points of view. In all four cases, the higher the value of the metric, the better the network. (D. Heckerman *et al.*, 1995, Silvia A. *et al.*, 2004).

BDe $BDe(B,T) = P(B,T) = P(B) \times \prod_{i=1}^{n} \prod_{j=1}^{q_i} \left(\frac{\Gamma(N'_{ij})}{\Gamma(N_{ij}+N'_{ij})} \times \prod_{k=1}^{r_i} \frac{\Gamma(N_{ijk}+N'_{ijk})}{\Gamma(N'_{ijk})}\right)$

$$\phi(B|T) = LL(B|T) - f(N)|B|,$$

$$LL(B|T) = \sum_{i=1}^{n} \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} N_{ijk} \log\left(\frac{N_{ijk}}{N_{ij}}\right).$$

Log-Likelihood(LL) If $f(N) = 0$, we have the **LL** score.
AIC If $f(N) = 1$, we have the **AIC** scoring function:
BIC If $f(N) = \frac{1}{2}\log(N)$, we have the **BIC** score.

---

4 Simulation

In this project, depending on the topology like Fig. 2., after create a set of models of the number of nodes to 3, 4, 6, 8, 10 pieces, and simulate with following restrictions.

- Cardinality was limited to two. It means, all variables are binary data.
- The probability value was set by Uniform distribution with minimum value is 0 and maximum value is 1. This way, we can make a more stochastic interpretation.
- All experiments are repeated 100 times. Because it is to avoid the consequences of coincidence.
- Constraint-based Learning Algorithms often makes undirected arcs. So, this has been excluded from comparison.

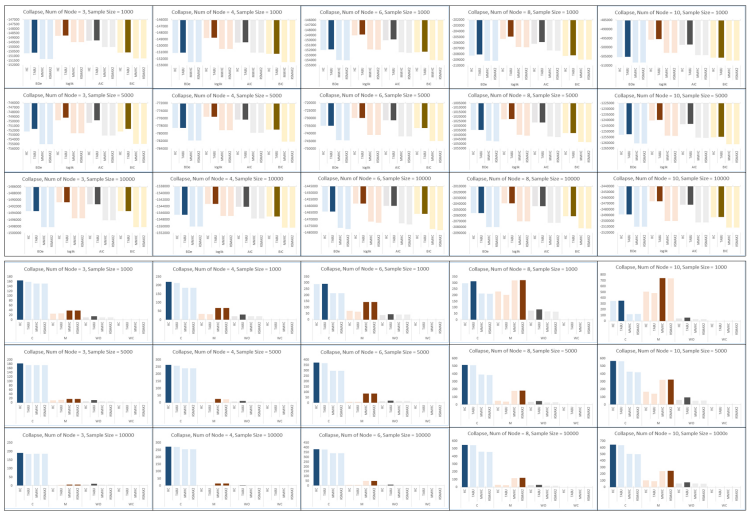

Fig. 3. Comparison of scores(top) and correct arcs(bottom) via Collapse

- According to scores, TABU > HC >>> MMHC = RSMAX2
- According to correct arcs, HC > TABU > MMHC = RSMAX2
- Therefore, HC = TABU >>> MMHC = RSMAX2

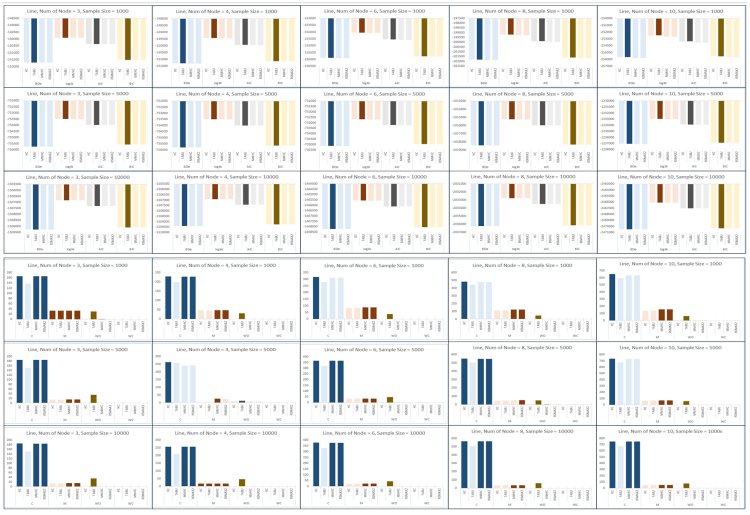

Fig. 4. Comparison of scores(top) and correct arcs(bottom) via Line

- According to scores, TABU > HC = MMHC = RSMAX2
- According to correct arcs, HC = MMHC = RSMAX2 >>> TABU
- Therefore, HC = MMHC = RSMAX2 >>> TABU

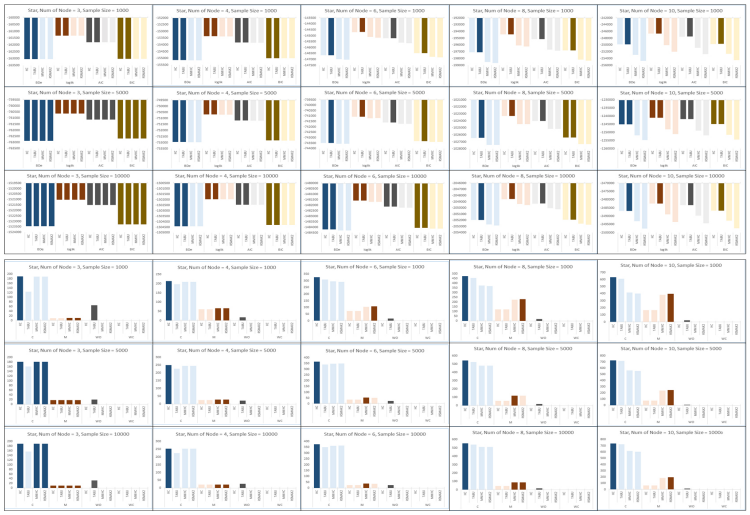

Fig. 5. Comparison of scores(top) and correct arcs(bottom) via Star

– According to scores, HC = TABU > MMHC = RSMAX2
– According to correct arcs, HC > TABU = MMHC = TABU
– Therefore, HC > TABU = MMHC = RSMAX2

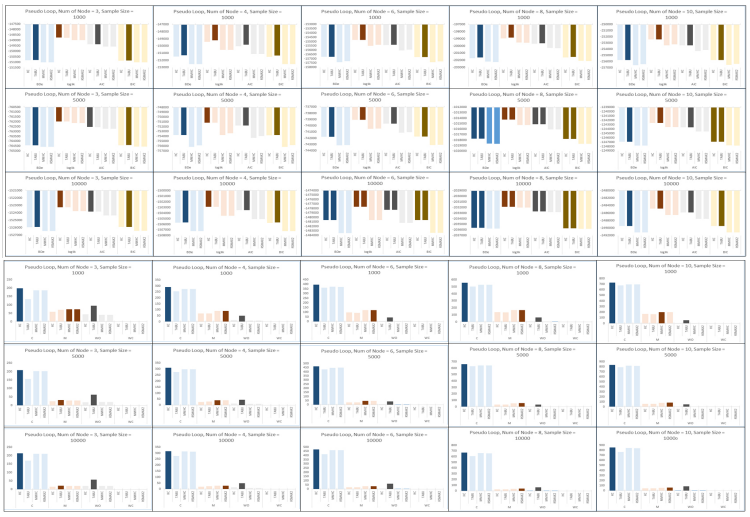

Fig. 6. Comparison of scores(top) and correct arcs(bottom) via Pseudo Loop

– According to scores, TABU > HC > MMHC > RSMAX2
– According to correct arcs, HC > MMHC = RSMAX2 >>> TABU
– Therefore, HC >>> TABU = MMHC = RSMAX2

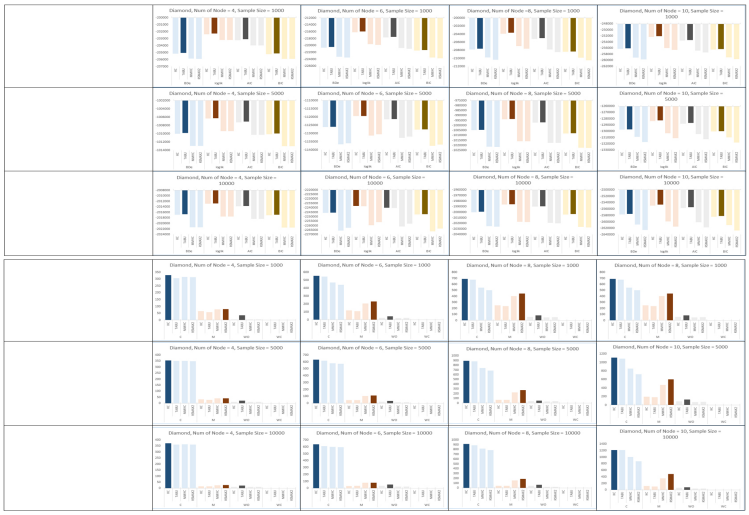

Fig. 7. Comparison of scores(top) and correct arcs(bottom) via Diamond

– According to scores, TABU > HC >>> MMHC > RSMAX2
– According to correct arcs, HC > TABU >>> MMHC = RSMAX2
– Therefore, HC = TABU >>> MMHC > RSMAX2

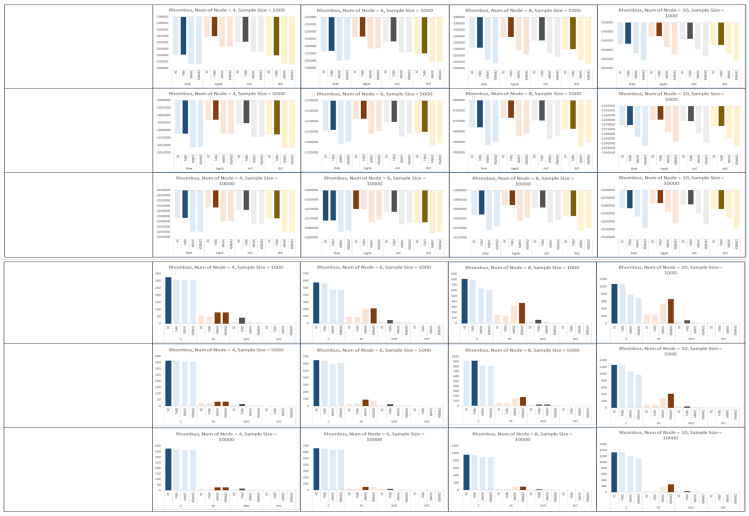

Fig. 8. Comparison of scores(top) and correct arcs(bottom) via Rhombus

– According to scores, TABU > HC >>> MMHC > RSMAX2
– According to correct arcs, HC > TABU >>> MMHC = RSMAX2
– Therefore, HC = TABU >>> MMHC > RSMAX2

5 Result

Results of compare the performance of each algorithm according to topologies like below.

Collapse: HC = TABU >>> MMHC = RSMAX2
Line: HC = MMHC = RSMAX2 >>> TABU
Star: HC > TABU = MMHC = RSMAX2
Pseudo Loop: HC >>> TABU = MMHC = RSMAX2
Diamond: HC = TABU >>> MMHC > RSMAX2
Rhombus: HC = TABU >>> MMHC > RSMAX2

Now, if we have a guess of groundtruth, we can select an algorithm for structure learning easily. Perhaps, this project suggests that Hill-Climbing is a good choice. However, when the many M, WO, WC is fatal, it will be able to try to consider the selection of hybrid algorithm. Hybrid algorithm compared to Score-based algorithm is found to be that draw the arc more conservative. This makes not only C is often less missing arcs, but also WO and WC is drawn very small. It seems to use when WO and WC are fatal. Especially MMHC and RSMAX2 for ”Line” seems to be advantageous

In future study, it can be to increase the number of node topology, to complex topology, to less sample size, or to increase the cardinality.

In this project, the probability when defining the relationship between the probability gave arbitrarily values on U(0, 1). But it is possible to confirm the relationship when given ”sequential” probability in future study.

Test Codes

`https://github.com/JaeseongYoo/PatternRecognition`

Bibliography

1. Alexandra M. C., (2009), Scoring functions for learning Bayesian networks *Inescid Tec. Rep.*
2. Beinlich I., Suermondt H. J., Chavez R. M. and Cooper G. F., (1989), The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks, *Proceedings of the 2nd European Conference on Artificial Intelligence in Medicine*, 247-256.
3. Binder J., Koller D., Russell S. and Kanazawa K., (1997), Adaptive probabilistic networks with hidden variables, *Machine Learning*, Vol. 29, No. 2-3, 213-244.
4. Daly R. and Shen Q., (2007), Methods to accelerate the learning of Bayesian network structures, *Proceedings of the 2007 UK Workshop on Computational Intelligence, Imperial College, London.*
5. D. M. Chickering, (1996), Learning Bayesian networks is NP-complete, *Learning from Data: Artificial Intelligence and Statistics V, Springer Verlag.*

6. Eitel J. M. L., (2008), An Information-geometric approach to learning Bayesian network topologies from data, *Innovations in Bayesian Networks Studies in Computational Intelligence*, Vol. 156, 187-217.
7. Kevin B. K. and Ann E. N., (2010), *Bayesian Artificial Intelligence, 2nd Edition*, CRC Press.
8. Marco S., (2010), Learning Bayesian networks with the bnlearn R package, *Journal of Statistical Software*, Vol. 35, Issue 3.
9. Margaritis D., (2003), Learning Bayesian network model structure from data, *Ph.D. thesis, School of Computer Science, Carnegie-Mellon University, Pittsburgh, PA*, Available as Technical Report CMU-CS-03-153.
10. Pearl J., (1988), *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann.
11. Robinson, R. W. (1973), Counting labeled acyclic digraphs, *New Directions in the Theory of Graphs, Academic Press*, pp. 239273.
12. X.-w. Chen, G. Anantha, and X. Wang, (2006), An effective structure learning method for constructing gene networks, *Bioinformatics*, Vol. 22, 1367-1374.