Chapter 60
An Algorithm for Bayesian Network Structure Learning Based on Simulated Annealing with Adaptive Selection Operator

Ao Lin, Bing Xiao, and Yi Zhu

Abstract In order to solve the problems that the intelligence algorithm falls into the local optimum easily and has a slow convergence in Bayesian networks (BN) structure learning, an algorithm based on adaptive selection operator with simulated annealing is proposed. This chapter conducts the adaptive selection rule in combination with conditional independence tests of BN nodes to guide the generation of neighbor. In order to better compare the adaptive effect, an algorithm based on selection operator with simulated annealing (SOSA) is proposed; at the same time 15 data sets in the three typical networks are accessed as learning samples. The results of the Bayesian Dirichlet (BD) score, Hamming distance (HD), and evolution time of the network after learning show that it has the quicker convergence and it searches the optimal solution more easily compared with simulated annealing (SA) and SOSA.

60.1 Introduction

As a graph model, BNs is an effective tool to deal with the uncertain problems in modeling and analysis, which is widely applied in many domains. Meanwhile, the learning problem of BNs is an important part of BN study. BN learning includes BN structural learning and parameter learning. Structure learning is needed to disclose the qualitative and the quantitative relationship between variables to light at the same time, while the BN structure learning is proved to be NP hard. Therefore, studying the BN structure learning problems is more challengeable and meaningful.

A. Lin (✉) • Y. Zhu
Department of Graduation Management, Air Force Early Warning Academy, Wuhan 430019, China
e-mail: lin_ao4035@163.com

B. Xiao
No. 4 Department, Air Force Early Warning Academy, Wuhan 430019, China

W.E. Wong and T. Zhu (eds.), *Computer Engineering and Networking*, Lecture Notes in Electrical Engineering 277, DOI 10.1007/978-3-319-01766-2_60,

In the investigation of BN structure learning, there are two classes of methods to deal with the structure learning problems [1]: The method of independence analysis and the method of score searching. The former determines whether there is border between the corresponding points or not by examining the conditional independence and dependence between variables, thereby establishing the skeleton of BN structure and orienting the border to get the BN structure. The latter is the method of score searching. For the network search space is of great extent generally, some BN structure learning adopts heuristic greedy algorithm, which tends to lead to the local optimum in the learning outcomes.

Nowadays, some researchers adopt the method of intelligence evolution to avoid the shortage of the heuristic algorithm [2, 3]. The SA algorithm is just the intelligence algorithm which is applied therein firstly, which is proved to be successful [4]. In contrast with the typical methods, it has a persistent evolution. In fact, examining the dependence and the independence between variables can reveal the variables' relational information which is camouflaged in data. Using this information can guide the evolution of the intelligence algorithm, thereby achieving the target of rapid convergence. ASOSA algorithm for the BN structure learning is proposed in this chapter.

60.2 BN Structure Learning

As a pictorial model that represents the joint probability distributions between variables, BNs include two parts: directed acyclic graph (DAG) and BN parameters. BN joint probability distributions can be decomposed as following through the independence relationship between variables contained in BN structure:

$$p(X_1, \cdots, X_n) = \prod_{i=1}^{n} p\left(X_i \middle| \mathbf{Pa}_i, G\right) \tag{60.1}$$

wherein $\mathbf{Pa}_i$ refers to the father node of the variable X_i in BN structure (G). BN structure learning refers to obtaining the network structure which matches the sample data fitting best by analyzing a variety of samples, which is the focus of this chapter.

The method based on conditional independence test is efficient. But in certain cases, the conditional independence test order (the variables' number of the conditional set is the number of the conditional independence test orders) increases exponentially with respect to the number of the variables.

This chapter adopts the mutual information and conditional mutual information in the conditional independence test. $I(X_i; X_j)$ expresses the mutual information between variables X_i and X_j. According to the information theory, $I(X_i; X_j)$ is

$$I(X_i;X_j) = I(X_j;X_i) = \sum_{X_i,X_j} p(X_i,X_j)\log\frac{p(X_i,X_j)}{p(X_i)p(X_j)} \tag{60.2}$$

The mutual information is nonnegative. The more variables X_i and X_j incline to independence, the more $I(X_i; X_j)$ approaches 0.

The method based on score searching is another method of the BN structure learning; this method defines grading function S for each candidate network. Generally, S is posterior probability of the network. When assuming that BNs have equal a priori probability, the comparison of the posterior probability S of different BN structures is the comparison of the structure likelihood $p(\mathrm{D}|G)$. Cooper and Herskovits provided the calculating procedure of the structure likelihood:

$$p(\mathrm{D}|G) = \prod_{i=1}^{n}\prod_{j=1}^{q_i}\frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij}+N_{ij})}\prod_{k=1}^{r_i}\frac{\Gamma(\alpha_{ijk}+N_{ijk})}{\Gamma(\alpha_{ijk})} \tag{60.3}$$

wherein Γ is gamma function, N_{ijk} is the number of cases in the dataset in which the parents of X_i are in state j and X_i itself is in state k, and q_i and r_i are the number of the parents of X_i and X_i in its own state separately. α_{ij} and α_{ijk} are the Dirichlet prior distributions. Equation (60.3) is also the famous Bayesian Dirichlet (BD) score function; the greater the structure score, the better the network structure.

60.3 Adaptive Selection Operator

The traditional SA algorithm of BN structure learning gets the neighbor by randomly selecting add, delete, or reverse the directed edges. This kind of operation is purposeless and ineffective. The zero-order independence information of the BN nodes can characterize the relationship between nodes substantially. So the information can be used to guide the generation of neighbor. The main thought is constructing the initial selection matrix with the nodes' zero-order information in the independence test. With the conducting of the annealing, effect of selection matrix is weakened gradually and selection matrix will not change any more when it meets certain conditions.

The selection matrix can be classified into the dependence selection matrix and the independence selection matrix which are applied to the addition and the deletion of the BN edges, respectively.

The selection matrix can be gained in two steps:

Step 1: Initialization. Set the coefficient of renovation k (k > 1). Calculate the mutual information of the two different random variable $C_{ij}^0 = I(X_i; X_j)$, $i = 1 \cdots m - 1$, and $j = i + 1 \cdots m$; m is the number of the BN nodes. The matrix C^0 is the initial dependence selection matrix; the independence selection matrix and the dependence selection matrix have the opposite

effects. In order to ensure the nonnegativity of the probability, $D_{ij}^0 = -C_{ij}^0 + \max(C_{ij}^0)$, $i = 1 \cdots m - 1, j = i + 1 \cdots m$. D^0 is the initial independence selection matrix.

Step 2: Updating. $p = p + 1$. $C_{ij}^p = (C_{ij}^{p-1})^{1/k}$, $D_{ij}^p = (D_{ij}^{p-1})^{1/k}$, $p \geq 1$ when it meets the conditions; $C_{ij}^p = C_{ij}^{p-1}$, $D_{ij}^p = D_{ij}^{p-1}$, $p \geq 1$ when it does not meet the conditions.

In order to keep the selectivity of the selection matrix, the maximum of the number of updating can be set.

60.4 ASOSA Algorithm

For the SA algorithm of BN structure learning, the initialized network structure is empty. It looks for the better network locally through the operation such as randomly adding, deleting the edge, and converting the edge's direction. Lower the temperature gradually to look for the locally optimal network, until the suspense condition is reached. Adaptive selection operator with simulated annealing (ASOSA) algorithm has the same main body frame as the SA algorithm, but it uses the selection probability of the selection matrix instead of the random selection to operate the edges. The algorithm proposed in this chapter is presented in Algorithm 1.

The marking criterion of the algorithm based on ASOSA adopts the BD score. For SA algorithm seeking for the minimum, the score results should maintain the negative value only. The condition of the step 13 and 14 refers to $\Delta S \leq 0$, and the updating time does not reach the maximal time λ.

In order to compare the effectiveness of the ASOSA algorithm, an algorithm based on selection operator with simulated annealing (SOSA) is designed the selection matrix of which will not change in the annealing process.

60.5 Experimentation

The standard way of assessing the effectiveness of a learning algorithm is to draw samples from a known BN, apply the algorithm on the artificial data, and to compare the learned structure with the original one [3]. This chapter chooses three typical networks in different domains for the experiment: Asia network [5] (8 variables, 8 edges), insurance network [6] (27 variables, 52 edges), and alarm network [7] (37 variables, 49 edges). For the three experimental networks, datasets with 100, 200, 500, 1,000, and 5,000 samples were generated separately by applying Gibbs sampling.

Algorithm 1: ASOSA algorithm

Input: Set of learning data
Output: Bayesian network

1. Set the initial temperature T_0.
2. Set the minimum temperature T_{end}.
3. $\nu = 0.99$;//The descent velocity of temperature.
4. $\beta = 20$;//The maximal time of the outside loop.
5. $\sigma = 20$;//The maximal time of the inside loop.
6. Calculate the mutual information between any two variables to get C^0.
7. $D^0 = -C^0 + \max(C^0)$.
8. $k = 1.1$;//Coefficient of renovation.
9. $\lambda = 15$;//The maximal updating time of the selection matrix.
10. G = empty graph;//The candidate graph is initialized into empty graph.
11. $T = T_0$.
12. **Repeat**
13. **If** the conditions are met, update the selection matrix C and D; **end**.
14. **If** the conditions are not met, do not update the selection matrix; **end**.
15. **For** σ times **do**.
16. Add one edge and reduce one edge to G, according to the selection matrix C and D, and reverse the direction of one of the directed edges in G randomly.
17. Calculate the score difference ΔS coming from the three operations above.
18. **If** $\Delta S > 0$ or $e^{(\Delta S/T)} > rand(0,1)$ **then** apply these actions to G; **end**.
19. **End**.
20. $T = T \times \nu$.
21. **Until** the maximal time of the loop or $T > T_{end}$ is obtained.
22. **Return** G.

60.5.1 Qualitative Analysis of Algorithms

Table 60.1 shows the BD score statistics to the learning results of the 15 experimental datasets from the three experimental networks by three algorithms (SA, SOSA, and ASOSA).

Hamming distance (HD) is an available approach to describe the difference between the network G after learning and the original network G^0. HD is the sum of excessive edge, deleted edge, and reversed edge. Table 60.2 shows the statistic results of the three different algorithms from the three different samples. It can be found from the statistic results of the HD that the performance of ASOSA is optimal and the posterior is SOSA.

Table 60.1 BD score for the learned structure (normalized for correct network score)

		Sample size					
Sample	Method	100	200	500	1,000	5,000	Avg.
Asia	SA	1.008	1.010	1.009	1.004	1.004	1.007
	SOSA	0.994	0.995	0.997	1.000	1.000	0.997
	ASOSA	0.994	0.995	0.997	0.999	1.000	0.997
	TRUE	1.000	1.000	1.000	1.000	1.000	1.000
	Empty	1.271	1.384	1.333	1.336	1.322	1.329
Insurance	SA	0.876	0.932	0.970	0.990	1.003	0.954
	SOSA	0.889	0.916	0.965	0.982	1.000	0.950
	ASOSA	0.886	0.927	0.964	0.982	0.998	0.951
	TRUE	1.000	1.000	1.000	1.000	1.000	1.000
	Empty	1.109	1.251	1.403	1.492	1.588	1.369
Alarm	SA	0.994	1.009	1.006	1.005	1.005	1.004
	SOSA	0.997	1.004	1.002	1.004	1.007	1.003
	ASOSA	0.990	1.003	0.996	0.999	1.000	0.998
	TRUE	1.000	1.000	1.000	1.000	1.000	1.000
	Empty	1.571	1.682	1.768	1.836	1.888	1.749

Table 60.2 HD for the learned structure

		Sample size					
Sample	Method	100	200	500	1,000	5,000	Avg.
Asia	SA	15	15	15	8	13	13.2
	SOSA	5	7	3	0	2	3.4
	ASOSA	5	3	3	2	2	3.0
Insurance	SA	53	43	31	37	22	37.2
	SOSA	49	29	25	24	23	30.0
	ASOSA	40	25	22	24	22	26.6
Alarm	SA	45	34	33	21	26	31.8
	SOSA	33	31	21	16	21	24.4
	ASOSA	25	20	18	13	15	18.2

Since the operation platforms of the algorithms are at variance, the scores should be normalized for the correct network score. As BD score is subtractive, the lower the normalized score, the better the network structure

It can be found from the learning results of the three networks that the normalized score of the SA algorithm is higher than the other two improved algorithms except the 100 samples of the insurance and alarm network learning, and the score of SOSA and ASOSA are superior to SA algorithm. In the learning, the score of ASOSA is no higher than SOSA in each group or on an average

60.5.2 Constringent Analysis of Algorithms

The runtime of algorithm can be the leading indicator to measure the algorithm efficiency, but the final results of the different algorithms are different. For convenience in comparison, take the time consumption of the BD final score of the SA

Table 60.3 Runtime (normalized for SA)

		Sample size					
Sample	Method	100	200	500	1,000	5,000	Avg.
Asia	SA	1.30	1.15	1.26	1.00	1.15	1.30
	SOSA	0.22	0.26	0.52	0.37	0.30	0.22
	ASOSA	0.19	0.19	0.33	0.26	0.15	0.19
Insurance	SA	1.15	1.00	1.10	1.13	1.47	1.17
	SOSA	1.04	0.63	0.55	0.67	0.71	0.72
	ASOSA	1.06	0.63	0.58	0.52	0.64	0.69
Alarm	SA	1.15	1.00	1.07	1.10	1.48	1.16
	SOSA	0.85	0.67	0.70	0.78	0.78	0.76
	ASOSA	0.62	0.55	0.60	0.72	0.78	0.65

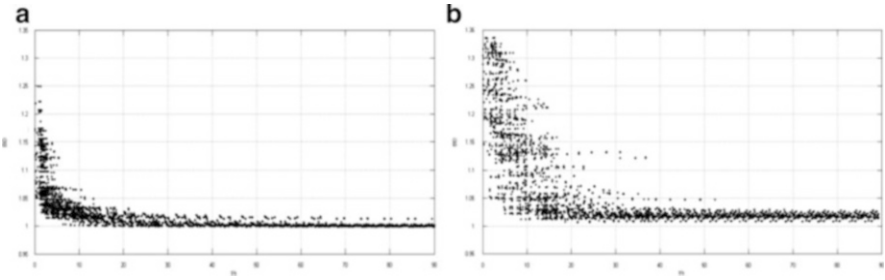

Fig. 60.1 (**a** and **b**) The performance record of ASOSA and SA

algorithm calculated by SOSA and ASOSA as the comparison object, and the result is normalized with the computation time of the SA, as shown in Table 60.3.

It can be found from the comparison of the runtime of the algorithms that the time of ASOSA is smaller than the other two algorithms, but there is little difference in SOSA and ASOSA.

In order to better compare the convergence of the algorithms, the performance process of the Asia network learning results from 500 samples which is obtained by ASOSA and SA is recorded. The cross axle is time, and the axis of ordinates is the BD score result normalized for the correct network score as shown in Fig. 60.1.

It can be found from the figure above that ASOSA converges rapidly. The searching directivity of the initial algorithm is conspicuous. The final convergence result is in the low level. The convergence process of the traditional SA is slow, and the final convergence result is worse than that of ASOSA.

60.6 Conclusion

This chapter introduces the ASOSA algorithm for BN structure learning, which fuses the independence analysis method and the score searching method of the BN structure learning, by the agency of the searching optimal network of the simulated annealing intelligence algorithm. The comparison of learning in three different

samples from multi-aspect is among the algorithms ASOSA, SOSA, and SA. The result shows that the ASOSA is superior to SOSA and SA in the learning accuracy and the time consumption.

References

1. Singh, M., & Valtorta, M. (1995). Construction of Bayesian network structures from data: A brief survey and an efficient algorithm[J]. *International Journal of Approximate Reasoning, 12*(2), 111–131.
2. Wong, M. L., & Leung, K. S. (2004). An efficient data mining method for learning Bayesian networks using an evolutionary algorithm-based hybrid approach[J]. *IEEE Transactions on Evolutionary Computation, 8*(4), 378–404.
3. Pinto, P. C., Nagele, A., et al. (2009). Using a local discovery ant algorithm for Bayesian network structure learning[J]. *IEEE Transactions on Evolutionary Computation, 13*(4), 767–777.
4. Kirkpatrick, S., Gelatt, C. D., et al. (1983). Optimization by simulated annealing[J]. *Science, 220*(4598), 671–680.
5. Lauritzen, S. L., Spiegelhalter, D. J., et al. (1988). Local computations with probabilities on graphical structures and their application to expert systems[J]. *Journal of the Royal Statistical Society, Series B, 50*(2), 157–224.
6. Binder, J., Koller, D., et al. (1997). Adaptive probabilistic networks with hidden variables [J]. *Machine Learning, 29*(2–3), 213–244.
7. Beinlinch, I. A., Suermondt, H. J., et al. (1989). The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks[C]. In *Proc. 2nd Europ. Conf. on Artificial Intelligence in Medicine Care* (pp. 247–256). Berlin: Springer-Verlag.