Journal of Systems Engineering and Electronics
Vol. 28, No. 6, December 2017, pp.1248 – 1255

Learning Bayesian network parameters under new monotonic constraints

Ruohai Di, Xiaoguang Gao*, and Zhigao Guo

School of Electronic and Information, Northwestern Polytechnical University, Xi'an 710129, China

Abstract: When the training data are insufficient, especially when only a small sample size of data is available, domain knowledge will be taken into the process of learning parameters to improve the performance of the Bayesian networks. In this paper, a new monotonic constraint model is proposed to represent a type of common domain knowledge. And then, the monotonic constraint estimation algorithm is proposed to learn the parameters with the monotonic constraint model. In order to demonstrate the superiority of the proposed algorithm, series of experiments are carried out. The experiment results show that the proposed algorithm is able to obtain more accurate parameters compared to some existing algorithms while the complexity is not the highest.

Keywords: Bayesian networks, parameter learning, new monotonic constraint.

DOI: 10.21629/JSEE.2017.06.22

1. Introduction

Bayesian networks have become a widely used method in the field of uncertain knowledge modeling [1 – 3]. A Bayesian network, which consists of a graph structure and some parameters, is a concise representation of a joint probability distribution over a set of stochastic variables [4]. Conceptually speaking, the structure of a Bayesian network describes the conditional independence relationship between a set of variables, and the parameters of a Bayesian network describe the quantitative relationship between a set of variables. A major difficulty in applying Bayesian network to solve the practical problem is to establish relationship between nodes from the training data.

When sufficient data are available, some methods have been proposed to learn the structure and parameters of Bayesian networks and have achieved satisfactory results [5 – 7]. Unfortunately, there are some data hard to get or getting the data will cost too much, such as geological hazard data and aerial combat data and so on. Therefore, the data are not sufficient in many cases, which may lead to inaccurate structure and parameters of a Bayesian network. Consequently, such "under-training" Bayesian networks with inaccurate structure and parameters will inevitably give rise to considerable errors when applied in reasoning and decision-making. Therefore, learning the structure and parameters of Bayesian networks when the size of the training samples is small is of fundamental importance both theoretically and practically. In this paper we place our emphasis on the parameter learning of Bayesian networks, whose structure is available, with a small-sized training data.

In order to acquire Bayesian network parameters when the data are not sufficient, some research has been completed. For the research, the crucial part is to incorporate the expert knowledge or domain knowledge into the process of learning parameters. Among these existing researches, different kinds of domain knowledge have been formalized into some types of constraints. Witting [8] used the qualitative influence constraint to construct the penalty function, combine the penalty function with the maximum entropy function to get the objective function, and then apply the adaptive probabilistic networks (APN) algorithm to learn the parameters. Altendorf [9] integrated the monotonic constraint into the objective function, but solve the problem with the gradient-descent algorithm. Feelders [10] transformed the qualitative influence constraint into the order constraint in order to correct the results of the maximum likelihood estimation (MLE). Campos and Niculescu [11 – 13] regarded the parameters learning as a constrains convex problem. Campos [14] proposed the constrained maximum entropy (CME) algorithm based imprecise Dirichlet model. Chang [15,16] put forward the qualitative maximum a posterior (QMAP) algorithm to learn the parameters, which used the Monte Carlo

Manuscript received August 01, 2016.
*Corresponding author.
This work was supported by the National Natural Science Foundation of China (61305133; 61573285) and the Fundamental Research Funds for the Central Universities (3102016CG002).

sampling to get the parameters in accordance with the constraint. And then, the hyperparameters of Dirichlet priors were obtained. Liao [17] proposed the constrained expectation maximum algorithm (CEM) for parameters learning, which combined the constraint and maximum entropy, and applied the gradient-descent algorithm to search the optimal solution. Zhou [18] proposed the multinomial parameters learning with constraints (MPL-C) algorithm, which used the auxiliary hybrid Bayesian network and dynamic discretization junction tree (DDJT) algorithm to learn the parameters. All above are the existing Bayesian network parameters learning methods that employ domain knowledge for learning parameters from small dataset. Besides, there are still a few other methods which need not domain knowledge, such as noisy-or-gates [19,20] and minimum free energies [21]. Based on the careful analysis of existing research, it is found that the description of qualitative influence constraint or monotonic constraint mostly are the sorting of the parameters of Bayesian networks when the status of the child node is the same, but the status of the parent node is different. In order to express the sorting of the parameters of Bayesian networks when the status of the parent node is the same, but the status of the child node is different, a new inequality constraint named new monotonic constraint is proposed. Even though the existing research mentions some methods which could learn the parameters of Bayesian networks under the inequality constraint, such as convex optimization and isotonic regression. The advantages of the two algorithms are their generic for all different inequality constraints. However, these two algorithms have their own shortcomings. The convex optimization has the high complexity and the isotonic regression has the poor learning results. Di [22] gave a method to deal with the new monotonic constraint. However, it brings about higher complexity and lower precision. In order to find a more convenient and effective method to learn the parameters of Bayesian networks under the proposed constraint, a novel algorithm is proposed which is called monotonic constraint estimation (MCE) algorithm.

The reminder of this paper is organized as follows. In Section 2, we briefly review Bayesian networks and give a simple instance to illustrate the problem we are solving in this paper. In Section 3, the new monotonic constraint is presented. In Section 4, the MCE algorithm is introduced in detail. In Section 5, the experiments are executed to evaluate the performance of the algorithm through the comparison with some existing methods. In the final section, a few conclusions are drawn from our work and some interesting directions are pointed out for further research.

2. Preliminaries

In this section, some concepts of Bayesian networks will be briefly reviewed to help readers better understand.

2.1 Bayesian networks

Bayesian networks are represented as a directed acyclic graph which contains some vertices and edges. Vertices represent random variables, while edges represent probabilistic relationship between random variables. For each variable node X, a conditional probability table is specified as $P(X|\pi(X))$, which describes the probability over the possible values of X and possible configurations of parent variables $\pi(X)$. In a Bayesian network, the joint probability can be written as follows:

$$p(x_1, \ldots, x_n) = \prod_{i=1}^{n} p(x_i|\pi(x_i)) \tag{1}$$

where x_i are the state value of variable i and $\pi(x_i)$ denotes the configuration value of parent variables of variable X_i.

To illustrate the Bayesian network more clearly, the lawn moist Bayesian network is employed, which is depicted as Fig. 1. The model will be used as the experimental model later. The variables in the network are binary, the value is 0 or 1, that is, if the variable occurs or is present it has the state 1. Besides, the conditional probabilities are also given in Fig. 1. Our purpose is to learn the parameters in the following Bayesian network.

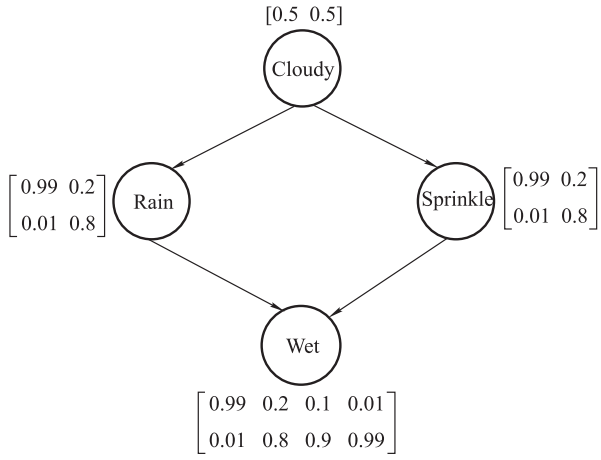

Fig. 1 Lawn moist model

2.2 Parameter learning

There are two algorithms often mentioned for Bayesian network parameters learning. They are Bayesian estimation and MLE. There is a hypothesis that the dataset is complete and has no missing data. The MLE of the pa-

rameters is

$$\theta_{ijk} = p(X_i = k|\pi(X_i) = j) = \frac{N_{ijk}}{\sum_{k'=1}^{r_i} N_{ijk'}} \tag{2}$$

where i is the index of node X, j is the index of parent nodes' configuration, k and k' are X_i's states, r_i is the number of the states of X_i, and N_{ijk} is the number of cases which satisfy $X_i = k$ and $\pi(X_i) = j$ in the dataset.

When Bayesian statistic is applied to get the estimation of the discrete random variables, based on Bayesian theorem, a posterior probability density function $\rho(\theta|d)$ is expressed as follows: $\rho(\theta|d) \propto \rho(\theta)f(d|\theta)$, where d represents the data, $\rho(\theta)$ is the prior distribution and $f(d|\theta)$ is the likelihood function. For the discrete random variables, the likelihood function of the parameters is the multinomial distribution function. The prior and posterior of the multinomial distribution are the Dirichlet distributions. Therefore, the Bayesian network parameters are obtained as

$$\theta_{ijk} = p(X_i = k|\pi(X_i) = j) = \frac{\alpha_{ijk} + N_{ijk}}{\sum_{k'=1}^{r_i} \alpha_{ijk'} + N_{ijk'}}. \tag{3}$$

As indicated in (3), parameters α prevent Bayesian network parameters from getting over-fit, especially when the dataset is small. However, it is difficult for a domain expert to accurately specify the parameters α of a Dirichlet distribution because they are not intuitive for the expert.

2.3 Definition of small data set

When the data are very sufficient, MLE is an ideal algorithm to learn the parameters. Unfortunately, the data set is often small, which makes the MLE get no accurate results. Therefore, there is a problem that how many samples could make the MLE get the expected accurate parameters. With regard to the problem, Dasgupta [23] defined a calculation of the lower limit of the sample number for parameters learning with a given Bayesian network structure. If there is a Bayesian network which has n boolean nodes, and no node has more than k parents, the sample number is calculated with a confidence $(1 - \delta)$ as follows:

$$\text{Number} \geqslant \frac{288 \times n^2 \times 2^k}{\varepsilon^2} \ln^2(1 + \frac{3n}{\varepsilon}) \cdot \ln\left(\frac{1 + 3n/\varepsilon}{\varepsilon\delta}\right). \tag{4}$$

Constant ε is the error and is often set as $\varepsilon = n\sigma$, σ is the Kullback-Leibler (KL) divergence of a parameter.

In the paper, when the given data set is smaller than the data size in (4), the given data set is defined as a small data set.

3. Monotonic constraint model

The monotonic constraint was proposed before, which informally means that higher values of the parent node stochastically result in higher values of the child node. However, the new monotonic constraint in this paper is different from the proposed monotonic constraint. The new monotonic constrain definition is given as follows:

Definition 1 For a Bayesian network G containing a set of variables $\{X_1, \ldots, X_n\}$, without loss of generality, the node X_i has $r$states which contain $\{1, 2, \ldots, k, \ldots, r\}, 1 \leqslant k \leqslant r$. Its parent node set $\pi(X_i)$ has q states which contain $\{1, 2, \ldots, j, \ldots, q\}, 1 \leqslant j \leqslant q$. When $\pi(X_i)$ is equal to j, the parameters $\theta_{ijk} = p(X_i = k|\pi(X_i) = j)$ satisfy

$$\theta_{ij1} \leqslant \theta_{ij2} \leqslant \cdots \leqslant \theta_{ijr} \text{ or } \theta_{ij1} \geqslant \theta_{ij2} \geqslant \cdots \geqslant \theta_{ijr}. \tag{5}$$

Then θ_{ijk} is considered to be consistent with the monotonic constraint, (5) is the model of the monotonic constraint. Suppose the Bayesian network parameters conform to the monotonic constraint. Combining with the criterion of the parameters, every parameter will be transformed into the interval as follows:

$$\begin{cases} \theta_{ij1} + \theta_{ij2} + \cdots + \theta_{ijr} = 1 \\ \theta_{ij1} \leqslant \theta_{ij2} \leqslant \cdots \leqslant \theta_{ijr} \end{cases} \Rightarrow$$

$$\begin{cases} 0 \leqslant \theta_{ij1} \leqslant \dfrac{1}{r} \\ \theta_{ij1} \leqslant \theta_{ij2} \leqslant \dfrac{1 - \theta_{ij1}}{r - 1} \\ \qquad \vdots \\ \theta_{ijr-2} \leqslant \theta_{ijr-1} \leqslant \dfrac{1 - \sum_{k=1}^{r-2} \theta_{ijk}}{2} \\ \theta_{ijr} = 1 - \sum_{k=1}^{r-1} \theta_{ijk} \end{cases}. \tag{6}$$

Suppose $p(\theta)$ conforms to Dirichlet distribution $D(\alpha_1, \alpha_2, \ldots, \alpha_r)$, which is

$$p(\theta) = \frac{\Gamma(\alpha)}{\prod_{i=1}^{r} T(\alpha_i)} \prod_{i=1}^{r} \theta^{\alpha_i - 1} \tag{7}$$

where $\alpha = \sum_{i=1}^{r} \alpha_i$. It is obvious that the Dirichlet distribution $D(\alpha_1, \alpha_2)$ will convert into Beta distribution

$B(\alpha_1, \alpha_2)$ when $r = 2$. Then, the Bayesian estimation of θ can be written as follows:

$$\theta_i^* = \int \theta_i p(\theta_i|D)\mathrm{d}\theta_i = \frac{m_i + \alpha_i}{m + \alpha}. \tag{8}$$

Considering that the joint distribution of the Bayesian network parameters is Dirichlet distribution and the marginal distribution of Dirichlet distribution is Beta distribution, the prior distribution of parameter θ will be depicted with Beta distribution.

$$p(\theta) = \frac{\Gamma(\alpha_1 + \alpha_2)}{\Gamma(\alpha_1)\Gamma(\alpha_2)}\theta^{\alpha_1 - 1}(1 - \theta)^{\alpha_2 - 1} \tag{9}$$

Equation (6) gives the upper and lower bounds of the Bayesian network parameters. If no additional prior information is available, the parameter θ whose scope is given by (6) can be presumed to conform to the uniform distribution $U(\theta_1, \theta_2)$. Then the problem is converted to how to use the Beta distribution to approach the uniform distribution. Therefore, we apply the method given by (10) to approach the uniform distribution using a Beta distribution.

$$\min_{\alpha_1 \alpha_2} \int_0^1 (U(\theta_1, \theta_2) - B(\alpha_1, \alpha_2))^2 \mathrm{d}\theta \tag{10}$$

where α_1 and α_2 are the parameters of Beta distribution. As solving the problem above leads to high computation, the approximate method is used as follows:

$$\begin{cases} m_1 = \int_0^1 U(\theta)\theta \mathrm{d}\theta \\ m_2 = \int_0^1 U(\theta)\theta^2 \mathrm{d}\theta \end{cases} \tag{11}$$

$$\begin{cases} M_1 = \int_0^1 B(\theta)\theta \mathrm{d}\theta = \dfrac{\alpha_1}{\alpha_1 + \alpha_2} \\ M_2 = \int_0^1 B(\theta)\theta^2 \mathrm{d}\theta = \\ \quad \dfrac{\alpha_1 \alpha_2}{(\alpha_1 + \alpha_2)^2(\alpha_1 + \alpha_2 + 1)} + M_1^2 \end{cases} \tag{12}$$

$$\begin{cases} M_1 = m_1 \\ M_2 = m_2 \end{cases} \Rightarrow \begin{cases} \alpha_1 = \dfrac{m_1^2 - m_1 m_2}{m_2 - m_1^2} \\ \alpha_2 = \dfrac{m_1 - m_2}{m_2 - m_1^2}(1 - m_1) \end{cases}. \tag{13}$$

Based on (11)–(13), we can figure out the parameters α_1 and α_2. Then α_1 and α_2 are used to calculate Bayesian network parameters as virtual samples using (8). Fig. 2 illustrates how to use a Beta distribution to approach the objective uniform distribution. The x-axis of Fig. 2 represents the random variable which belongs to $[0, 1]$. The y-axis represents the probability of the random variable.

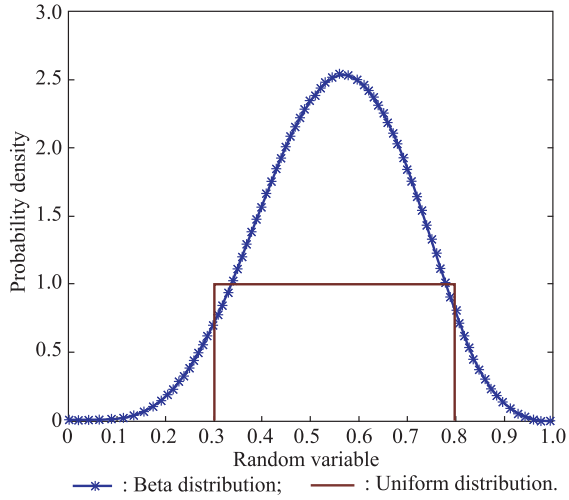

Fig. 2 Uniform and Beta distribution

4. MCE algorithm

The new monotonic constraint and how to convert it to a Beta distribution have been given in Section 3. Then this section will explain how to use Beta distribution in the process of Bayesian network parameters learning. If Bayesian network parameters θ_{ij^*} subject to the monotonic constraints, then the parameter learning process is given as follows:

Step 1 Establish the monotonic constraint based on the domain knowledge.

Step 2 Obtain interval constraint using (5) and (6), and the scope of θ_{ij1} will be obtained first.

Step 3 Convert uniform distribution into Beta distribution using (10)–(13) and obtain virtual sample information.

Step 4 Merge the virtual sample information into Bayesian estimation and get the estimation of parameters:

$$\theta_{ijk} = \frac{\alpha_1 + N_{ijk}}{\alpha_1 + \alpha_2 + N_{ij}} \tag{14}$$

where N_{ijk} is the number of samples when the value of node i is k and the value of its parent is j.

Step 5 Take obtained parameters as the lower bound of the next parameter, return to Step 2 and repeat Steps 2–5 until all parameters are obtained.

The process of the MCE algorithm is reported in Fig. 3. The MCE is an iterative algorithm until all the parameters θ_{ij^*} have been obtained.

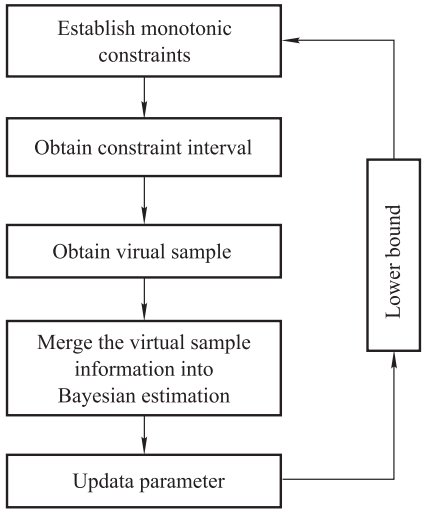

Fig. 3 Process of the MCE

5. Experiments

In this paper, the classical lawn moist Bayesian network, which has been shown in Fig. 1, is applied for simulation. Our objective is to learn Bayesian network parameters using the MCE algorithm based on the new monotonic constraints. According to domain knowledge and common sense, we give the monotonic constraints among the parameters of the lawn moist network as follows:

$$P(R=1|C=1) > P(R=0|C=1)$$

$$P(S=1|C=1) < P(S=0|C=1)$$

$$P(R=1|C=0) < P(R=0|C=0)$$

$$P(S=1|C=0) > P(S=0|C=0)$$

$$P(W=1|R=1,S=1) > P(W=0|R=1,S=1)$$

$$P(W=1|R=1,S=0) < P(W=0|R=1,S=0)$$

$$P(W=1|R=0,S=1) < P(W=0|R=0,S=1)$$

$$P(W=1|R=0,S=0) < P(W=0|R=0,S=0).$$

It means that the event will occur when the value of the variable is 1, oppositely, the event will not occur. Accordingly, $P(R=1|C=1) > P(R=0|C=1)$ means the probability of raining is higher when the weather is cloudy.

In order to analyze the performance of the proposed algorithm, MCE is compared with MLE, convex optimization estimation (COE) [10] and isotonic regression estimation (IRE) [11] in the experiment with the same constraints. The training data are generated from the network with manually assigned parameters using Bayesian network toolbox designed by Murphy. Learning results are calculated by averaging KL divergence between the learned parameters and the true parameters. The KL divergence is defined as follows:

$$\mathrm{KL}(Pr', Pr) = \sum_X Pr(X) \lg \frac{Pr(X)}{Pr'(X)} \tag{15}$$

where Pr and Pr' represent the true parameters and parameters learned by some samples respectively.

In the simulation experiment, four different algorithms are executed under different sample sizes. All the algorithms are repeated for 50 times and the mean KL divergence is calculated as the performance criteria. The results are depicted in Fig. 4 and Fig. 5. Fig. 5 is the local enlarged drawing of the Fig. 4.

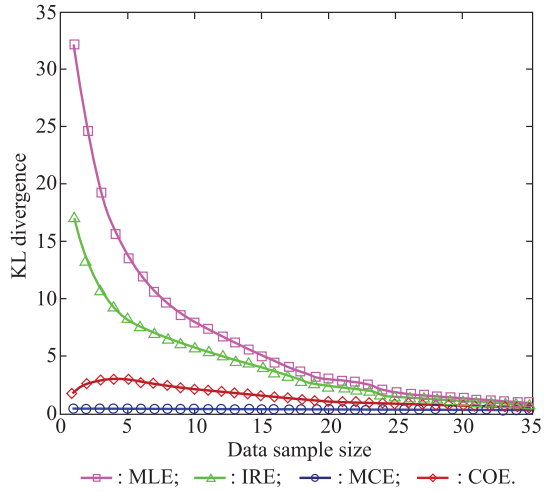

Fig. 4 KL divergence of all algorithms

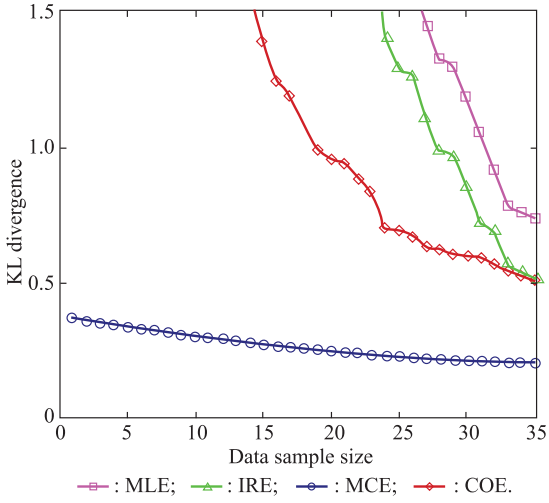

Fig. 5 KL divergence of all algorithms

The experimental results show that the KL divergence of the proposed algorithms (MCE) is the minimum of all the algorithms when the sample size is less than 35. The conclusion is especially obvious when the sample size is

less than 10. In other words, the MCE has the highest accuracy of all. The accuracy of the IRE and COE are worse than MCE. MLE is the worst. The KL divergences of some experiments are given in Table 1. In addition, the sample size these algorithms need to get for the same KL divergence is given in Fig. 6. As you can see in Fig. 6, the MCE algorithm needs the smallest data size to achieve the same KL divergence.

Table 1 KL divergence under different sample sizes

Data size	5	10	15	20	25	30	35
MLE	13.41	7.79	5.00	2.82	1.74	1.18	0.73
IRE	8.11	5.49	3.92	2.34	1.28	0.84	0.51
COE	2.82	2.02	1.38	0.95	0.69	0.60	0.50
MCE	0.33	0.29	0.27	0.24	0.23	0.21	0.19

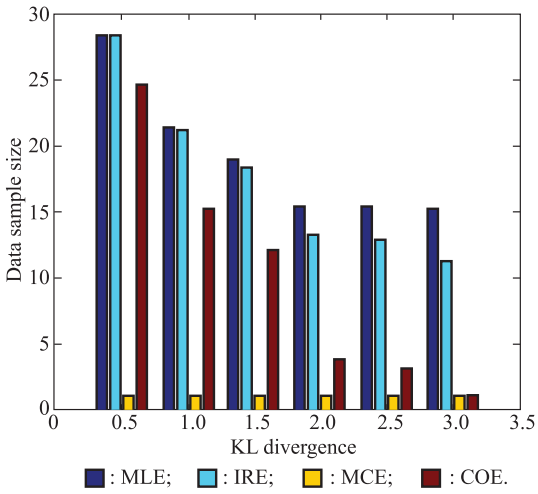

Fig. 6 Samples needed by different algorithms to achieve the same KL divergence

Average simulation results are given in the previous paragraph. In order to further demonstrate the superiority of the proposed algorithm. The KL divergence of 10 experiments is given in Fig. 7 when the sample size is 10. Because of the sample randomness, the results are different every time. Fig. 7 clearly shows that the MCE algorithm has the best stability and accuracy of all the algorithms. Meanwhile, the learned parameters are applied for reasoning. In the experiment, $P(C|R,S)$ will be computed when the evidence R is 0 and S is 1. With the increase of the data sample size, $P(C=0|R,S)$ becomes bigger and bigger while $P(C=1|R,S)$ becomes smaller and smaller in Fig. 8. This condition could make the reasoning more clearly. All in all, it is concluded that the MCE algorithm can learn parameters most accurately from small dataset by employing monotonic constraints.

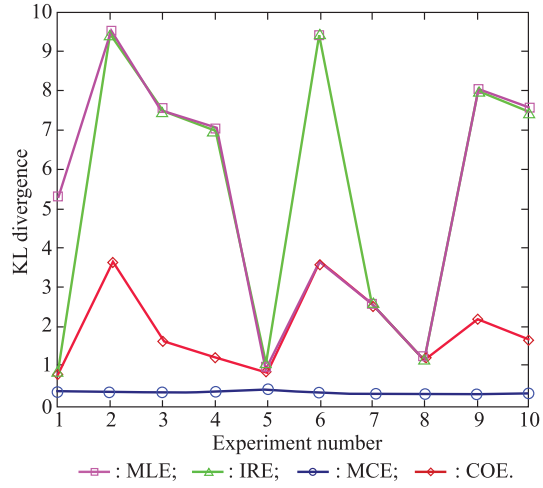

Fig. 7 KL of different algorithms with 10 samples

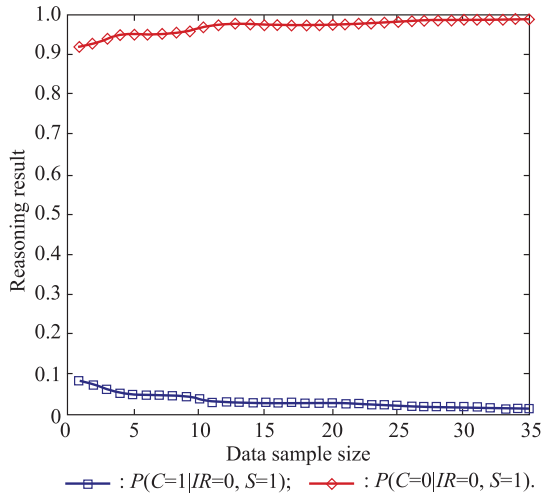

Fig. 8 Reasoning result of MCE

In order to analyze the complexity of every algorithm, the runtime is used as a criterion. Similar to the previous KL divergence, the experiment will be repeated for 50 times to get the mean values of the runtime. Fig. 10 is the local enlarged drawing of Fig. 9. From Fig. 9 and Fig. 10, the conclusions can be obtained that MCE algorithm costs less runtime than the COE algorithm and costs more runtime than the MLE algorithm and the IRE algorithm. The MLE algorithm costs the least runtime and COE algorithm costs the most. The runtime of all the algorithms are given in Table 2. The runtime of the MCE algorithm is 0.04 s longer than MLE and IRE. Similar to the KL divergence, the runtime of 10 experiments are given in Fig. 11 and Fig. 12 when the sample size is 10. Fig. 12 is the local enlarged drawing of Fig. 11. The conclusions can be obtained that the COE algorithm costs the most runtime and the MLE algorithm costs the least runtime.

Generally speaking, the proposed algorithm has higher complexity than MLE and IRE and lower complexity than COE.

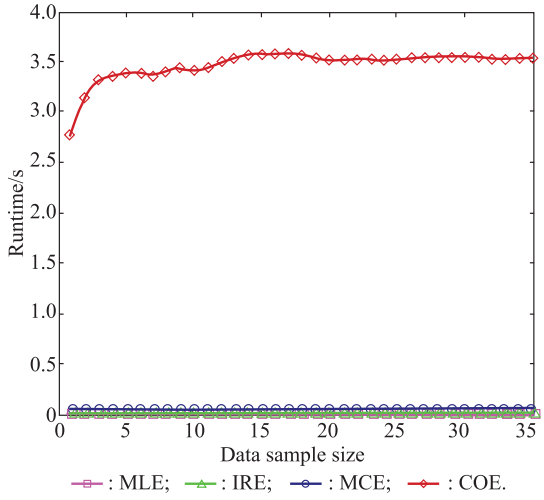

Fig. 9 Runtime of different algorithms

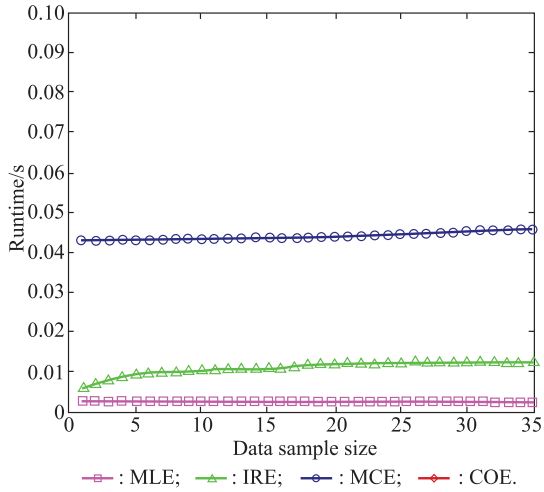

Fig. 10 Runtime of different algorithms (local)

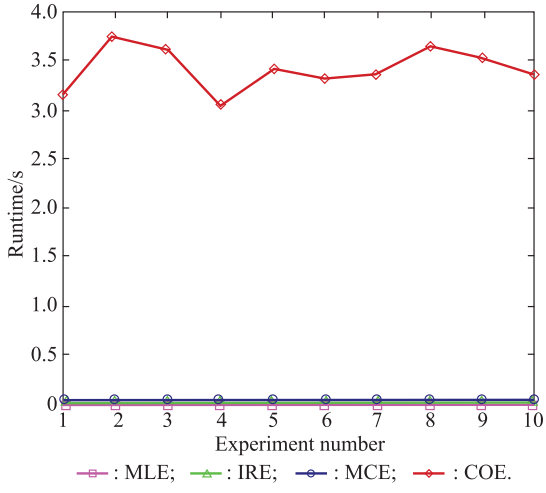

Fig. 11 Runtime of different algorithms with 10 samples

Table 2 Runtime of different algorithms s

Data size	5	10	15	20	25	30	35
MLE	0.002 3	0.002 3	0.002 3	0.002 3	0.002 3	0.002 4	0.002 4
IRE	0.009 1	0.010 1	0.010 6	0.011 3	0.011 7	0.011 8	0.012 3
COE	3.377 2	3.426 6	3.581 1	3.518 8	3.524 6	3.524 7	3.533 9
MCE	0.042 8	0.043 2	0.043 6	0.044 0	0.044 5	0.044 9	0.045 2

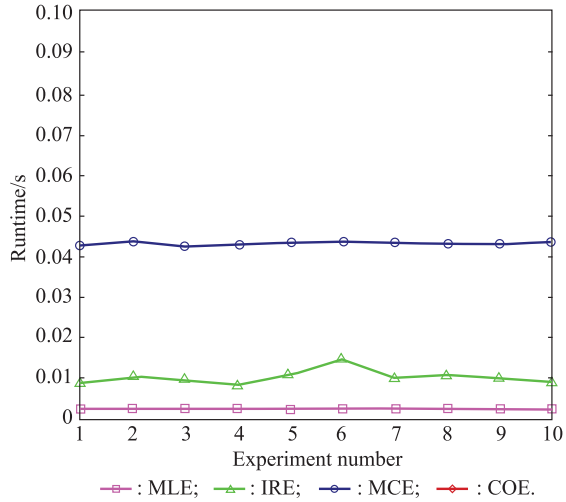

Fig. 12 Runtime of different algorithms with 10 samples (local)

6. Conclusions and future research

In this paper, a new monotonic constraint has been proposed to describe the new domain knowledge of parameter learning. And then, a novel algorithm is proposed to learn the parameters under the new monotonic constraints. The proposed algorithm is also suitable for learning Bayesian network parameters from small-sized training data under interval constraint. The proposed algorithm is compared with some existing methods in the experiments. The results show that the proposed algorithm improves the accuracy of parameter learning, while leads to a little higher complexity. The paper provides a new frame for Bayesian network parameters learning when the training data are not enough.

In addition, we find some interesting directions for further research. First, when the monotonic constraint is partly known, the proposed algorithm cannot deal with it, then how to solve that problem. Second, in order to get more accurate parameters, integrating other types of constraints (such as additive synergy or product synergy constraints) with the monotonic constraint to improve the accuracy could deserve further focus.

References

[1] C. Chen, J. Liang, X. Zhu. Gait recognition based on improved dynamic Bayesian networks. *Pattern Recognition*, 2011, 44(4): 988 – 995.

[2] G. Infantes, M. Ghallab, F. Ingrand. Learning the behavior model of a robot. *Autonomous Robots*, 2011, 30(2): 157 – 177.

[3] L. Du, P. H. Wang, H. W. Liu. Bayesian spatiotemporal multitask learning for radar HRRP target recognition. *IEEE Trans. on Signal Processing*, 2011, 59(7): 3182 – 3197.

[4] J. Pearl. Probabilistic reasoning in intelligent systems. *Computer Science Artificial Intelligence*, 1988, 70(14): 521 – 538.
[5] G. F. Cooper, E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 1992, 9(4): 309 – 347.
[6] I. Tsamardinos, L. E. Brown, C. F. Aliferis. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 2006, 65(1): 31 – 78.
[7] L. D. Campos. A new approach for learning belief networks using independence criteria. *International of Approximate Reasoning*, 2000, 24(1): 11 – 37.
[8] F. Wittig, A. Jameson. Exploiting qualitative knowledge in the learning of conditional probabilities of Bayesian networks. *Proc. of the 29th Conference on Uncertainty in Artificial Intelligence*, 2013: 644 – 652.
[9] E. E. Altendorf, A. C. Restificar, T. G. Dietterich. Learning from sparse data by exploiting monotonicity constraints. *Computer Science*, 2012: 2012arXiv207.1364A.
[10] A. Feelders, L. C. V. D. Gaag. Learning Bayesian networks parameters under order constraints. *Journal of Approximate Reasoning*, 2006, 42(1): 37 – 53.
[11] C. P. D. Campos, Q. Ji. Improving Bayesian network parameter learning using constraints. *Proc. of the 19th Conference on Pattern Recognition*, 2009: 1 – 4.
[12] R. S. Niculescu, T. M. Mitchell, R. B. Rao. Bayesian network learning with parameter constraints. *Journal of Machine Learning Research*, 2006, 7(1): 1357 – 1383.
[13] R. S. Niculescu. *Exploiting parameter domain knowledge for learning in Bayesian networks*. Pittsburgh: Carnegie Mellon, 2005.
[14] C. P. D. Campos, Q. Ji. Bayesian networks and the imprecise dirichlet model applied to recognition problems. *Lecture Notes in Computer Science*, 2011, 6717: 158 – 169.
[15] R. Chang, W. Wang. Novel algorithm for Bayesian network parameter learning with informative prior constraints. *Proc. of International Joint Conference on Neural Networks*, 2010: 1 – 8.
[16] R. Chang, R. Shoemaker, W. Wang. A novel knowledge-driven systems biology approach for phenotype prediction upon genetic intervention. *IEEE Trans. on Computational Biology & Bioinformatics*, 2011, 8(5): 1170 – 1182.
[17] W. H. Liao, Q. Ji. Learning Bayesian networks parameters under incomplete data with domain knowledge. *Pattern Recognition*, 2009, 42(11): 3046 – 3056.
[18] Y. Zhou, N. Fention, M. Neil. Bayesian network approach to multinomial parameter learning using data and expert judgments. *Journal of Approximate Reasoning*, 2014, 55(5): 1252 – 1268.
[19] A. Onisko, M. J. Druzdzel, H. Wasyluk. Learning Bayesian network parameters from small data sets: application of noisy-or gates. *Journal of Approximate Reasoning*, 2001, 27(2): 165 – 182.
[20] J. H. Bolt, L. C. V. D. Gaag. An empirical study of the use of the noisy-or model in a real-life Bayesian network. *Communications in Computer & Information Science*, 2010, 80: 11 – 20.
[21] T. Isozaki, N. Kato, M. Ueno. Minimum free energies with "data temperature" for parameter learning of Bayesian networks. *International Journal of Artificial Intelligence Tools*, 2009, 18(5): 653 – 671.
[22] R. H. Di, X. G. Gao, Z. G. Guo. Discrete Bayesian network parameter learning based on monotic constraint. *Systems Engineering and Electronics*, 2014, 36(2): 38 – 43. (in Chinese)
[23] S. Dasgupta. The sample complexity of learning fixed-structure Bayesian networks. *Proc. of the 14th Conference on Machine Learning*, 1997: 165 – 180.

Biographies

Ruohai Di was born in 1986. He received his M.S. and Ph.D. degrees in system engineering of Northwest Polytechnical University, in 2011 and 2016, respectively. He currently does postdoctoral research in Northwestern Polytechnical University. His topics of interests include Bayesian networks (structure and parameter learning algorithm), and data mining.
E-mail: xfwtdrh@163.com

Xiaoguang Gao was born in 1957. She received her M.S. and Ph.D. degrees in systems engineering from Northwest Polytechnical University, in 1986 and 1989, respectively. She is currently a professor in Department of Systems Engineering, School of Electronics and Information, Northwestern Polytechnical University. Her topics of interests include Bayesian networks, deep learning, and advanced control theory and application.
E-mail: cxg2012@nwpu.edu.cn

Zhigao Guo was born in 1986. He received his M.S. degree in pattern recognition and intelligent system from Shenyang Aerospace University, in 2010. He is currently a Ph.D. student in system engineering in Department of Systems Engineering, School of Electronics and Information, Northwestern Polytechnical University. His topics of interests include Bayesian networks (parameter learning), and data mining.
E-mail: guozhigao2004@163.com