

현실 세계의 불완전한 데이터를 위한 베이지안 네트워크 파라미터의 온라인 학습

(Online Learning of Bayesian Network Parameters for
Incomplete Data of Real World)

임 성 수 [†] 조 성 배 ^{**}
(Sung-Soo Lim) (Sung-Bae Cho)

요 약 최근 현실 세계의 불확실한 환경을 극복하기 위한 방법 중 하나로 베이지안 네트워크 (Bayesian network; BN)가 부각되고 있다. BN의 파라미터 학습은 주어진 평가 척도에 따라 데이터의 훈련집합에 가장 잘 부합되는 네트워크 파라미터를 구하는 것으로, BN 설계에 드는 시간과 노력을 줄이기 위해 연구되어 왔다. 기존의 오프라인 학습은 학습에 필요한 충분한 양의 데이터를 모으기에는 많은 노력과 시간이 필요하다. 또한 현실세계는 불완전성을 포함하고 있어 완전한 데이터를 얻기 힘들다. 본 논문에서는 불완전한 데이터로부터 온라인으로 BN 파라미터를 학습하는 방법을 제안한다. 이 방법은 불완전한 데이터로부터 학습이 가능하도록 하여 학습의 유연성을 높이고, 실시간 학습을 통해 변화하는 환경에 대한 적용성을 높인다. Cohen 등이 제안한 온라인 파라미터 학습방법인 Voting EM 알고리즘과 비교 실험한 결과, 완전한 데이터를 가지고 학습한 경우에는 동일한 학습 결과를, 그리고 불완전한 데이터의 경우에는 보다 나은 학습 결과를 얻었다.

키워드 : 베이지안 네트워크, 온라인 파라미터 학습, Voting EM 알고리즘

Abstract The Bayesian network (BN) has emerged in recent years as a powerful technique for handling uncertainty in complex domains. Parameter learning of BN to find the most proper network from given data set has been investigated to decrease the time and effort for designing BN. Off-line learning needs much time and effort to gather the enough data and since there are uncertainties in real world, it is hard to get the complete data. In this paper, we propose an online learning method of Bayesian network parameters from incomplete data. It provides higher flexibility through learning from incomplete data and higher adaptability on environments through online learning. The results of comparison with Voting EM algorithm proposed by Cohen at el. confirm that the proposed method has the same performance in complete data set and higher performance in incomplete data set, comparing with Voting EM algorithm.

Key words : Bayesian network, Online parameter learning, Voting EM algorithm

1. 서 론

현실 세계의 응용 어플리케이션 환경은 시간이 지남에 따라서 변화하며 정확한 판측이 어려워 많은 불확실성을 포함한다. 최근 이러한 등적이고 불확실한 환경을

극복하기 위한 방법 중 하나로 베이지안 네트워크 (Bayesian Network: BN)가 부각되고 있다[1]. BN은 방향성 비순환 그래프(Direct Acyclic Graph: DAG)로 각각의 노드는 확률 변수를 나타내고 각 노드를 연결하는 화살표는 변수들간의 인과관계를 표현한다.

보통 BN은 전문가의 지식을 바탕으로 설계된다. 이 방법은 전문가의 경험을 직접적으로 네트워크에 반영할 수 있는 장점이 있지만, 네트워크의 규모가 커지면 설계에 많은 시간이 들어가고 네트워크의 정확성을 보장할 수 없다. 따라서 사람이 직접 네트워크를 설계하기 보다는 데이터로부터 통계적으로 BN을 설계하는 방법이 연구되었다[2-4]. 만일 학습할 데이터가 완전하고 충분하

· 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 지원 사업의 연구결과로 수행되었음. IITA-2006-(C1090-0603-0046)

[†] 학생회원 : 연세대학교 컴퓨터과학과
lss@sclab.yonsei.ac.kr

^{**} 정 회원 : 연세대학교 컴퓨터과학과 교수
sbcho@cs.yonsei.ac.kr

논문접수 : 2006년 3월 30일

심사완료 : 2006년 9월 20일

게 준비되어 있다면 최대우도 추정(Maximum Likelihood Estimation: MLE) 방법을 이용하면 데이터에 적절한 네트워크에서 노드들의 조건부 확률 테이블(Conditional Probability Table: CPT)을 얻을 수 있다. 그러나 실제로 많은 문제에서는 완전한 데이터를 얻기 어렵다. 과거 몇 년간 이러한 데이터의 불완전함을 극복하기 위해서 순차적 학습, EM 알고리즘, Gibbs 샘플링 등이 연구되어 왔다[5-7]. 하지만 이러한 방법들은 학습을 위해서 충분한 양의 데이터가 필요하므로, 데이터가 충분하지 않은 경우에는 충분한 양의 데이터가 모아질 때까지 기다려야 한다. 또한 주어진 데이터를 기반으로 오프라인 상에서 학습하므로 실시간으로 변화하는 환경의 상태나 사용자의 성향을 파악하는 데에는 한계가 있다.

위와 같은 오프라인 학습의 단점을 극복하기 위해, 온라인 파라미터(CPT) 학습방법에 대해서 논의되어 왔다[8,9]. Cohen 등은 EM(η) 알고리즘을 온라인 학습에 적용하여 Voting EM 알고리즘을 제안하였다[10,11]. Zhang 등은 Voting EM 알고리즘을 홍수 결정 지원 시스템(flood decision supporting system)에 적용하여 그 유용성을 검증하였다[12]. Voting EM 알고리즘은 데이터가 완전한 경우, 비교적 빠른 시간에 적절한 파라미터 값을 얻을 수 있지만, 데이터가 불완전한 경우에는 학습이 부분적이거나 부적절한 방향으로 일어난다. 본 논문에서는 Voting EM 알고리즘의 이러한 한계점을 보완하는 개선된 학습방법을 제안한다. 제안하는 방법은 예측치와 관측치의 차이를 이용한 학습방법으로 불완전한 데이터로부터 학습이 가능하도록 하여 학습의 유연성을 높이고, 실시간 학습을 통해 사용자나 환경의 변화를 잘 모델링한다.

본 논문의 구성은 다음과 같다. 2장에서는 온라인 베이저안 파라미터 학습방법인 Voting EM 알고리즘에 대해서 알아보고, 3장에서는 Voting EM 알고리즘의 한계점을 극복하기 위한 온라인 학습방법을 제안한다. 그리고 4장에서는 제안하는 방법의 우수성을 보이기 위해 아시아 네트워크[13] 상에서 Voting EM 알고리즘과의 비교평가 실험을 수행하고 그 결과에 대해서 설명한다. 마지막으로 5장에서는 결론 및 향후 연구에 대해서 언급한다.

2. Voting EM 알고리즘

x_i 를 $\{x_i^1, x_i^2, \dots, x_i^{p_i}\}$ 의 p_i 개의 값을 갖는 BN의 노드라 하고, π_i 를 $\{\pi_i^1, \pi_i^2, \dots, \pi_i^{q_i}\}$ 의 q_i 개의 상태 조합을 갖는 x_i 의 부모 집합이라고 하면, x_i 에서의 CPT값 $\theta_{ijk} = P(x_i = x_i^k | \pi_i = \pi_i^j)$ 로 정의할 수 있다. 온라인 BN 파라

미터 학습은 주어진 데이터 집합 $D = \{d_1, d_2, \dots, d_t, \dots\}$ 와 현재의 파라미터 집합 Θ_t 을 이용하여 다음 시점의 파라미터 집합 Θ_{t+1} 을 수식 (1)과 같이 구한다.

$$\Theta_{t+1} = \underset{\Theta}{\operatorname{arg\,max}} [\eta L(\Theta | D) - d(\Theta, \Theta_t)] \quad (1)$$

여기서 $L(\Theta | D)$ 는 파라미터 집합 Θ 와 데이터 집합 D 간의 로그 유사도(log likelihood: LL)를 나타내고, $d(\Theta, \Theta_t)$ 는 두 파라미터 집합 Θ 와 Θ_t 간의 거리를 나타낸다. 즉, Θ_{t+1} 은 주어진 데이터 집합 D 와의 LL 값이 높으면서 이전의 파라미터 집합 Θ_t 의 특성을 유지한다. η 는 거리에 대한 LL의 중요도로, 데이터의 학습률을 의미한다. Bauer 등은 모든 i, j 에 대해서 $\sum_k \theta_{ijk} = 1$ 을 만족해야 하는 제약조건을 이용하여 수식 (1)의 최대화 문제를 풀었다[9].

Cohen 등은 Bauer의 EM(η) 알고리즘을 온라인 학습에 적용시켜 Voting EM 알고리즘을 제안했다[10,11]. Voting EM 알고리즘은 데이터로부터 파라미터를 실시간으로 학습하는 방법으로, BN의 구조가 정해져 있고 확률 변수들이 불연속 값을 취할 때 적용할 수 있다[10,11]. 온라인 BN 파라미터 학습은 시점 t 에서 관측된 데이터 $d_t = \{x_1 = s_1, x_2 = s_2, \dots, x_n = s_n\}$ 와 주어진 파라미터 집합 Θ_t 를 이용하여 새로운 파라미터 집합 Θ_{t+1} 을 구하는 것이다. 여기서 노드 x_i 의 상태 값이 관측되지 않은 경우 s_i 는 null 값을 갖는다. Voting EM 알고리즘은 데이터 d_t 를 이용하여 $t+1$ 시점에서의 파라미터 집합 Θ_{t+1} 의 파라미터 값 θ_{ijk}^{t+1} 를 수식 (2)와 같이 구한다.

$$\theta_{ijk}^{t+1} = \begin{cases} (1-\eta)\theta_{ijk}^t + \eta, & \text{if } x_i = x_i^k \text{ at } d_t \text{ and } P(\pi_i^j | d_t, \Theta_t) \neq 0 \\ (1-\eta)\theta_{ijk}^t, & \text{if } x_i \neq x_i^k \text{ at } d_t \text{ and } P(\pi_i^j | d_t, \Theta_t) \neq 0 \\ \theta_{ijk}^t, & \text{otherwise} \end{cases} \quad (2)$$

Voting EM 알고리즘의 학습방법(수식 2)을 다른 시각에서 분석해 보면, 파라미터 집합 Θ_t 의 각 원소(θ_{ijk}^t)를 독립적으로 학습하며 지수 평활법[1]과 매우 유사함을 알 수 있다. 다른 점은 $P(\pi_i^j | d_t, \Theta_t) = 0$ 인 경우에는 θ_{ijk}^t 를 학습하지 않는다는 점이다. θ_{ijk}^t 는 t 시점에서 π_i

1) 가장 최근 데이터에 가장 큰 가중치가 주어지고 시간이 지남에 따라 가중치가 기하학적으로 감소되는 가중치 이동 평균 예측 기법의 하나

가 j 번째 상태조합을 가질 때 x_i 가 k 번째 상태 값일 확률을 나타내므로 $P(\pi_i^j | d_i, \Theta_i) = 0$ 인 경우는 파라미터를 학습하지 않고, $x_i = null$ 인 경우에는 $P(x_i = x_i^k | d_i, \Theta_i) = \theta_{jk}^i$ 이므로 $\theta_{jk}^{i+1} = \theta_{jk}^i$ 가 되어 이 경우도 학습이 일어나지 않는다.

Voting EM 알고리즘은 EM 알고리즘에 근본을 둔다. BN 파라미터 학습을 위한 EM 알고리즘의 목적은 주어진 데이터 $D = \{d_1, d_2, \dots, d_n\}$ 로부터 로그 유사도를 최대화하는 파라미터를 구하는 것으로, 데이터에서 $x_i = x_i^k$ 이고 $\pi_i = \pi_i^j$ 인 샘플의 빈도수 $f(x_i^k, \pi_i^j)$ 를 예측하고(수식 (3)), 이렇게 예측된 빈도수를 바탕으로 파라미터 값을 갱신한다(수식 (4)).

$$f(x_i^k, \pi_i^j) = \sum_j p(x_i^k, \pi_i^j | d_i, \Theta_i) \quad (3)$$

$$\theta_{jk}^{i+1} = \frac{f(x_i^k, \pi_i^j)}{\sum_j f(x_i^k, \pi_i^j)} \quad (4)$$

수식 (3)과 (4)에서 보듯이 EM 알고리즘은 해당 파라미터(θ_{jk})의 값을 구하기 위해서 x_i 와 π_i 의 빈도수만을 고려한다. 이러한 EM 알고리즘에 기반한 Voting EM 알고리즘도 역시 수식 (2)에서 보는 것과 같이 x_i 와 π_i 만을 고려하여 θ_{jk} 를 얻는다. EM 알고리즘의 경우에는 충분한 학습 데이터로부터 빈도수 $f(x_i^k, \pi_i^j)$ 를 적절히 예측할 수 있으므로 불완전한 데이터에 대해서도 높은 정확도를 유지하며 학습이 가능하다. 하지만 Voting EM 알고리즘의 경우는 현실점에 해당하는 한 개의 데이터만을 보고 학습을 진행하므로, EM 알고리즘과 비슷한 접근을 하더라도 정확한 학습을 할 수 없다.

수식 (2)를 보면, Voting EM 알고리즘은 노드 x_i 의 CPT 값을 d_i 에서 x_i 가 관측되었을 경우에만 학습이 가능하다. 또한, x_i 가 관측되더라도 x_i 의 부모 노드의 집합 π_i 가 모두 관측되지 않았을 경우에는 $P(\pi_i^j | d_i, \Theta_i)$ 가 0이 아닌 모든 π_i^j 에 대해서 부모 노드가 나타날 확률과 상관없이 일률적인 비율로 학습이 일어나므로 적절한 학습이 이루어지지 않는다.

만일 그림 1과 같은 BN에서 노드 X3을 제외하고 다른 모든 노드들은 관측되었을 경우($d_i = \{x_1^a, x_2^b, x_3^?, x_4^d\}$), 앞에서 언급한 바와 같이 노드 X3의 CPT는 학습되지 않는다. 그리고 수식 (2)에 의해서 노드 X4의 CPT는

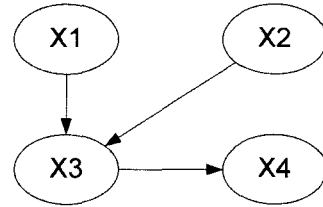


그림 1 간단한 베이저안 네트워크

$P(x_3^j | d_i, \Theta_i) \neq 0$ 인 j 에 대해서 일률적으로 $x_4^k = x_4^d$ 인 CPT 값은 $(1-\eta)\theta_{4jk}^i + \eta$ 가 되고, $x_4^k \neq x_4^d$ 인 CPT 값은 $(1-\eta)\theta_{4jk}^i$ 가 된다. 즉, 노드 X4는 노드 X3이 관측되지 않은 상황에서는 $P(x_4 = x_4^d | d_i, \Theta_i)$ 의 확률만을 학습하고, 노드 간의 인과 관계를 반영한 CPT 값 $(P(x_4 = x_4^d | \pi_4^j, d_i, \Theta_i))$ 은 정확하게 학습하지 못한다.

3. 불완전한 데이터로부터 BN 파라미터의 온라인 학습방법

2장에서는 Voting EM 알고리즘과 그 한계점에 대해서 알아보았다. 본 장에서는 Voting EM 알고리즘의 특징을 유지하면서 그 한계점을 극복하는 온라인 BN 파라미터 학습방법에 대하여 알아본다.

3.1 제안하는 온라인 BN 파라미터 학습방법

온라인 BN 파라미터 학습은 시점 t 에서 관측된 데이터 d_t 와 주어진 파라미터 집합 Θ_t 를 이용하여 새로운 파라미터 집합 Θ_{t+1} 을 구하는 것이다. 제안하는 방법은 관측값과 예측값의 차이를 고려하여 학습하는 방법으로, 데이터 d_t 와 파라미터 집합 Θ_t 를 이용하면 관측값과 예측값을 다음과 같이 나타낼 수 있다.

- 관측값 \hat{x}_i^k : x_i 가 k 번째 상태값을 가지면 1, 그렇지 않으면 0의 값을 가짐
- 예측값 P_i^k : 데이터 d_t 와 파라미터 집합 Θ_t 를 근거로 하여 추론된 x_i 가 k 번째 상태 값을 가질 확률 $(P(x_i = x_i^k | d_t - \{x_i\}, \Theta_t))$

본 논문에서는 현재의 파라미터 관측값 \hat{x}_i^k 와 예측값 P_i^k 를 이용하여 데이터 d_t 를 확률적으로 잘 설명할 수 있는, 즉 관측값 \hat{x}_i^k 와 예측값 P_i^k 간의 차이가 적은 파라미터 집합 $\bar{\Theta}$ 를 확률에 근거하여 구하고, 지수 평활법을 사용하여 다음 시점의 파라미터 집합 Θ_{t+1} 을 구한

다. 수식 (9)는 Θ_i 와 $\bar{\Theta}$ 로부터 Θ_{i+1} 을 계산하는 과정을 보여준다. 여기서 η 는 파라미터 $\bar{\theta}_{ijk}$ 에 대한 비중, 즉 학습률을 나타낸다.

$$\forall ijk, \theta_{ijk}^{t+1} = (1-\eta)\theta_{ijk}^t + \eta\bar{\theta}_{ijk} \quad (9)$$

이제 $\bar{\Theta}$ 를 구하는 방법에 대해서 알아보자. BN의 구조가 주어져 있을 때, x_i 를 제외하고 관측된 노드 집합을 $O = \{x_{o_1}, x_{o_2}, \dots, x_{o_n}\}$ 라 하고, x_i 를 제외하고 관측되지 않은 노드 집합을 $U = \{x_{u_1}, x_{u_2}, \dots, x_{u_{n-1}}\}$ 라고 하면, 예측값 P_i^k 는 수식 (10)과 같이 구하고, 이 수식은 독립성 가정과 체인-룰을 이용하면 수식 (11)과 같이 CPT의 값을 이용한 수식으로 표현할 수 있다.

$$P_i^k = \frac{\sum_{\text{for all state of } x \in U} P(x_{o_1} = x_{o_1}^k, \dots, x_{o_n} = x_{o_n}^k, x_i = x_i^k, x_{u_1}, \dots, x_{u_{n-1}})}{\sum_{\text{for all state of } x \in U \cup \{x_i\}} P(x_{o_1} = x_{o_1}^k, \dots, x_{o_n} = x_{o_n}^k, x_i = x_i^k, x_{u_1}, \dots, x_{u_{n-1}})} \quad (10)$$

$$\frac{\sum_{\text{for all state of } x \in U} P(x_{o_1} = s_1 | \pi_{o_1}) \times \dots \times P(x_{o_n} = s_n | \pi_{o_n}) \times P(x_i | \pi_i) \times P(x_{u_1} | \pi_{u_1}) \times \dots \times P(x_{u_{n-1}} | \pi_{u_{n-1}})}{\sum_{\text{for all state of } x \in U \cup \{x_i\}} P(x_{o_1} = s_1 | \pi_{o_1}) \times \dots \times P(x_{o_n} = s_n | \pi_{o_n}) \times P(x_i | \pi_i) \times P(x_{u_1} | \pi_{u_1}) \times \dots \times P(x_{u_{n-1}} | \pi_{u_{n-1}})} \quad (11)$$

수식 (11)을 정리해보면 $\frac{A}{A+B}$ 의 형태가 된다. 이의 역수를 취하면, 수식 (12)와 같이 정리할 수 있다. 여기서, A 와 B 는 CPT 값들의 곱과 합으로 구성된다.

$$\frac{1}{P_i^k} = 1 + \frac{B}{A} \quad (A = [0,1], B = [0,1]) \quad (12)$$

따라서, \hat{x}_i^k 와 P_i^k 의 차이를 줄이기 위해서, \hat{x}_i^k 가 1인 경우에는 A 의 값이 1이 되도록 관련 파라미터들을 조정하고 B 의 값은 0이 되도록 관련 파라미터들을 조정하며, 반대로 \hat{x}_i^k 가 0인 경우에는 A 의 값은 0이 되도록, 그리고 B 의 값은 1이 되도록 관련 파라미터들을 조정한다.

그러나 모든 i, j, k 에 대해서 수식 (11)을 이용하여 CPT 값($\bar{\Theta}$)을 설정하면, 복잡한 네트워크에서 결국 값이 많은 데이터의 경우에는 많은 양의 연산이 필요하다. 따라서 본 논문에서는 문제를 쉽게 하기 위하여 파라미터를 학습할 때, 노드 x_i 의 자식 노드가 주는 영향을 배제하고 부모 노드는 서로 독립이라고 가정한다. 그러면 수식 (10)은 수식 (13)과 같이 나타낼 수 있다.

$$P_i^k = \sum_{\text{for all } j} P(x_i = x_i^k | \pi_i^j, d_i - \{x_i\}, \Theta_i) P(\pi_i^j | d_i - \{x_i\}, \Theta_i)$$

$$= \sum_{j \in P(\pi_i^j | d_i - \{x_i\}, \Theta_i) \neq 0} \theta_{ijk}^t \prod_{x_c^d \in \pi_i^j} P(x_c^d | d_i - \{x_i\}, \Theta_i) \quad (13)$$

따라서 $\bar{\Theta}$ 를 구하기 위해서 우선 $\bar{\Theta} = \Theta_i$ 가 되도록 설정하고, 예측치 P_i^k 가 관측치 \hat{x}_i^k 와 동일한 값을 가지도록 $\bar{\Theta}$ 를 갱신하면 된다. 그림 2는 본 논문에서 적용한 파라미터 갱신 방법을 의사코드로 보여주고 있다.

```

OnlineLearning(d_i)
1 for all x_i in S do // S is the structure of BN
2 for k=0 to p_i do
3 if x_i^k = 1 then Update(i, k, 1.0)
4 else if x_i^k = 0 then Update(i, k, -1.0) end if
5 end for
6 end for
7 for all omega_jk in Omega do
8 if omega_jk > 0 then theta_jk = (1 - omega_jk) * theta_jk^t + omega_jk
9 else theta_jk = (1 - omega_jk) * theta_jk^t end if
10 end for
11 for all i, j do
12 normalize to sum_{j,k} theta_jk = 1 end for
13 for all theta_jk in Theta do
14 theta_jk^{t+1} = (1 - eta) * theta_jk^t + eta * theta_jk
15 end for

Update(i, k, w)
1 if |w| < threshold then return end if
2 for all pi_i^j in S do
3 omega_jk = omega_jk + w * P(pi_i^j | d_i - {x_i}, Theta_i)
4 end for
5 for all pi_i^j in S do
6 if P(pi_i^j | d_i - {x_i}, Theta_i) != 0
7 for all x_c^d in pi_i^j and x_c at d_i = null do
8 Update(a, b, w * theta_jk^t * prod_{x_c^d in pi_i^j - {x_c^d}} P(x_c^d | d_i - {x_i}, Theta_i))
9 end for
10 end if
11 end for
    
```

그림 2 제안하는 온라인 학습방법의 의사코드

함수 OnlineLearning은 시점 t 에서의 관측된 데이터 d_t 를 입력으로 받아 $t+1$ 시점의 파라미터 집합 Θ_{t+1} 을 구하며, 함수 Update는 수식 (13)을 이용하여 파라미터의 갱신 비중 Ω 를 구하는 함수로, i, k 는 x_i^k 의 상태를 나타내고, w 는 파라미터의 갱신 정도를 나타내는 것

로 -1.0 (P_i^k 값이 0이 되도록 함)에서 1.0 (P_i^k 값이 1이 되도록 함) 사이의 값을 갖는다.

OnlineLearning 함수에서 1~6번째 줄은 모든 i, k 에 대하여 \hat{x}_i^k 가 1인 경우, P_i^k 값이 1이 되도록 하는 파라미터 갱신 비중을 구하고(Update($i, k, 1.0$)), \hat{x}_i^k 가 0인 경우는 P_i^k 값이 0이 되도록 하는 파라미터 갱신 비중을 구한다(Update($i, k, -1.0$)). 그리고 7~10번째 줄에서는 구해진 파라미터 비중을 따라서 $\bar{\Theta}$ 값을 결정하고, 11~12번째 줄에서는 $\bar{\Theta}$ 의 값을 베이저안 네트워크의 제약 조건에 맞도록 조정한다. 마지막으로 13~15번째 줄에서는 각 파라미터의 값을 학습률 η 에 따라 갱신함으로써 학습을 마친다.

Update 함수에서 첫번째 줄은 기준치 이하의 비중은 처리하지 않음으로써 학습시간을 단축시키는 역할을 하며, 2~4번째 줄은 주어진 i, k 에 대해서 $\bar{\theta}_{jk}$ 의 갱신 비중을 정하는 부분으로 수식 (13)에서 $P(\pi_i^j | d_i - \{x_i\}, \Theta_i)$ ($= \prod_{x_a^b \in \pi_i^j} P(x_a^b | d_i - \{x_i\}, \Theta_i)$)의 값이 클수록 $\bar{\theta}_{jk}$ 가 P_i^k 에 미치는 영향이 크므로 $P(\pi_i^j | d_i - \{x_i\}, \Theta_i)$ 와 갱신의 정도를 나타내는 변수 w 를 곱하여 $\bar{\theta}_{jk}$ 의 갱신 비중을 계산한다. 그리고 5~11번째 줄은 $\bar{\theta}_{jk}$ 이외에 노드 x_i 와 관련된 파라미터 중 데이터가 주어지지 않아 학습이 불가능한 노드들의 갱신 비중을 계산하는 부분으로, Update 함수를 재귀적으로 호출하여 노드 x_i 의 각각의 부모 상태를 고려한다. 이때, 노드 x_i 와 멀어질수록 갱신 비중을 줄여주기 위해서 갱신 정도를 나타내는 변수 값으로 $w\theta_{jk}^t \prod_{x_c^d \in \pi_i^j - \{x_i\}} P(x_c^d | d_i - \{x_i\}, \Theta_i)$ (수식 (13)을 전개하여, $P(x_a^b | d_i - \{x_i\}, \Theta_i)$ 에 대해서 정리했을 때, $P(x_a^b | d_i - \{x_i\}, \Theta_i)$ 앞에 오는 계수 $\times w$)을 사용한다.

3.2 제안하는 방법의 동작 과정

본 절에서는 제안하는 방법이 문제에 적용되었을 때 어떻게 파라미터 값을 학습하는지에 대해 알아본다. 제안하는 방법은 완전한 데이터의 경우 Voting EM 알고리즘과 동일하게 동작한다. 수식 (13)에서 완전한 데이터의 경우, $\prod_{x_a^b \in \pi_i^j} P(x_a^b | d_i - \{x_i\}, \Theta_i)$ 의 값은 j 가 노드 x_i 의 부모의 상태일 때 1을, 그렇지 않을 때 0을 가지므로, $P_i^k = \theta_{jk}^t$ (j 는 주어진 완전한 데이터 상의 노

드 x_i 의 부모의 상태)가 된다. 따라서 $\bar{\theta}_{jk}$ 의 값은 $P(\pi_i^j | d_i, \Theta_i) = 1$ 인 경우, d_i 에서 $\hat{x}_i^k = 1$ (x_i 가 k 번째 상태 값을 갖는 경우)이면 1이 되고, $\hat{x}_i^k = 0$ (x_i 가 k 번째 상태 값을 갖지 않는 경우)이면 0이 되며, $P(\pi_i^j | d_i, \Theta_i) = 0$ 인 경우에는 θ_{jk}^t 의 값을 갖게 되므로 결국 Voting EM 알고리즘과 동일하게 된다.

반면 불완전한 데이터의 경우에는 제안하는 방법은 2장에서 살펴본 Voting EM 알고리즘의 한계점을 극복한다. 3.1절에서 살펴본 바와 같이 제안하는 방법은 Update 함수를 통해 관측되지 않은 노드에 대해서 ω 값도 계산되므로(5~11번째 줄) 관측되지 않은 노드에서도 학습이 가능하고, 부모들의 상황을 확률적으로 고려하여 ω 값을 결정하므로(3번째 줄) 노드들 간의 인과관계를 보다 정확히 학습할 수 있다.

예를 들어, 그림 3과 같은 네트워크의 구조와 파라미터 Θ_i 를 갖는 BN이 있다고 가정하자.

이때, $d_i = \{A=1, B=?, C=0, D=1\}$ 의 불완전한 데이터가 관측된다면, Voting EM 알고리즘의 경우는 그림 4와 같이 $\bar{\Theta}$ 의 값이 결정되므로, 노드 A에서는 노드 B와 상관없이 학습이 이루어져서 인과관계를 반영한 학습을 할 수 없고, 노드 B에서는 학습이 일어나지 않게 된다.

반면 제안하는 방법에서는 $d_i = \{A=1, B=?, C=0, D=1\}$ 의 불완전한 데이터에 대해서 그림 2의 알고리즘을 통해 그림 5와 같이 $\bar{\Theta}$ 의 값을 결정한다. 즉, 노드 B가 관측되지 않더라도, 노드 A는 노드 B의 확률에 따라서 학습이 진행되어 노드간의 인과관계를 학습할 수 있고, 노드 B는 노드 A를 학습하는 과정에서 Update 함수의 재귀 호출을 통해서 학습이 가능하다.

4. 실험 및 결과

본 논문에서는 제안하는 방법의 유용성을 검증하기 위하여 아시아 네트워크[13] 상에서 Voting EM 알고리즘과의 비교 실험을 수행하였다. 학습 데이터로는 환경의 변화에 대한 적응성을 알아보기 위하여 10,000개를 준비하였다. 앞의 5,000개의 데이터는 실제 아시아 네트워크에서 데이터를 생성하였고($P(Tub = pre | Visit = ok, \Theta_i) = 0.05$), 뒤의 5,000개의 데이터는 아시아를 방문한 사람의 결핵 발병 확률을 40%로 수정하여 데이터를 생성하였다($P(Tub = pre | Visit = ok) = 0.4$).

실험은 데이터의 손실률과 학습률을 변경시켜가며 수행하였다. 데이터 손실률은 특정 노드(tuberculosis)에서

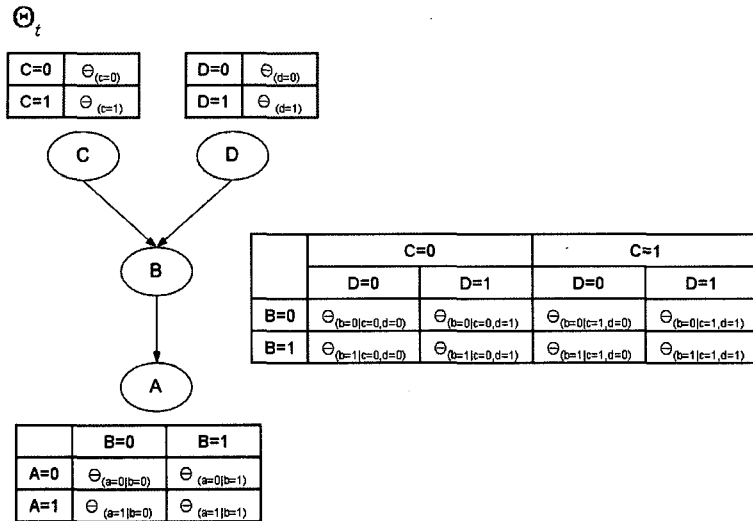


그림 3 시점 t에서의 파라미터

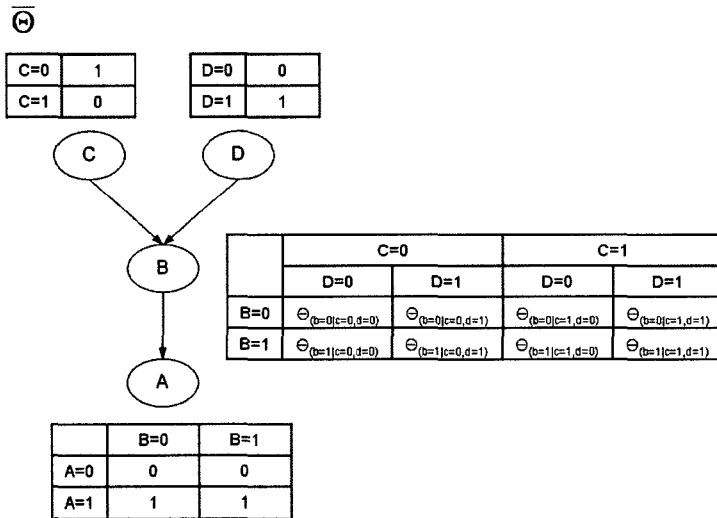


그림 4 Voting EM 알고리즘에서의 학습 과정

의 손실률 0%의 완전한 데이터에서부터 100%에 이르기까지 10%단위로 변경하였고, 학습률은 1~50%로 1% 단위로 변경하였다. 학습된 파라미터는 실제 해와의 거리를 이용하여 평가하였다. 두 파라미터 집합 $\Theta, \bar{\Theta}$ 간의 거리는 수식 (17)과 같이 계산한다.

$$D(\Theta, \bar{\Theta}_t) = \sum_{\text{for all } i, j, k} |\theta_{ijk} - \bar{\theta}'_{ijk}| \quad (17)$$

그림 6은 데이터 손실률에 따른 10,000개의 시점과 50개의 학습률에 대한 평균값을 보여준다. 그림에서 보여주듯이 데이터 손실이 없는 경우에는 Voting EM 알고리즘과 제안하는 방법은 동일한 결과를 얻었으며, 데

이타 손실이 발생한 경우 제안하는 방법이 Voting EM 알고리즘에 비해 좀더 정확한 학습을 수행함을 알 수 있다. 그리고 제안하는 방법은 데이터 손실률에 비례하여 실제 해와의 거리가 증가함에 반해 Voting EM 알고리즘은 0~10% 사이와 90~100%사이에서 급격한 변화가 발생하였다. 0~10% 사이의 변화는 Voting EM 알고리즘이 완전한 데이터에서는 잘 작동하지만 데이터 손실이 존재하는 경우에는 적절한 학습을 수행하지 못하기 때문이고, 90~100% 사이의 변화는 데이터 손실률이 100%일 경우에는 해당 노드(tuberculosis)에서 학습이 전혀 일어나지 않기 때문이다.

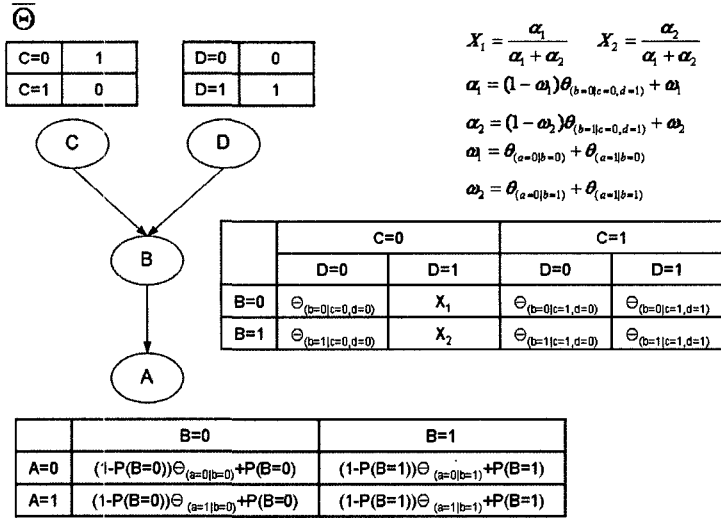


그림 5 제안하는 방법에서의 학습 과정

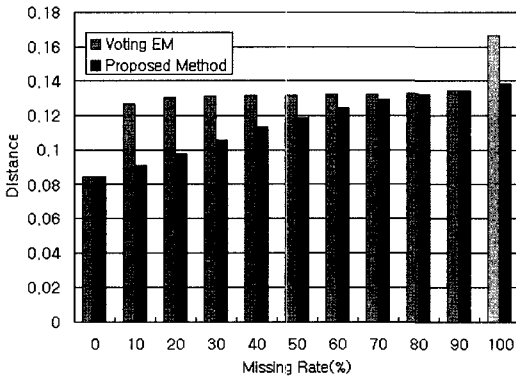


그림 6 Missing Rate에 따른 실제 해와의 거리

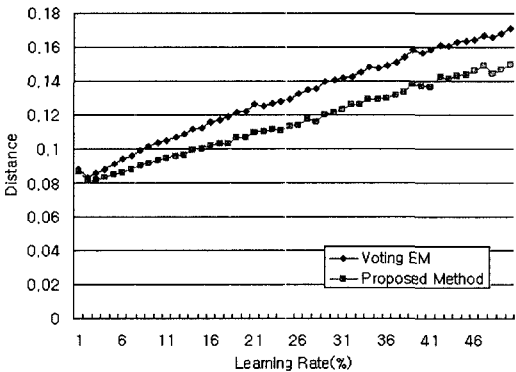


그림 7 Learning Rate에 따른 실제 해와의 거리

그림 7은 학습률에 따른 10,000개의 시점과 11개의 데이터 손실률에 대한 평균값을 보여준다. 이 그림을 통

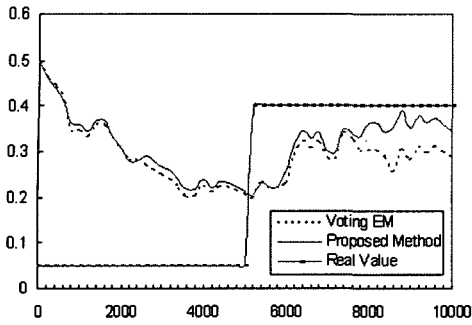
해서도 제안하는 방법이 Voting EM 알고리즘보다 좋은 성능을 보이고 있음을 알 수 있으며 학습률로는 3~5%가 적절함을 보여준다.

그림 8은 시간에 따른 $P(Tub = pre | Visit = ok, \Theta_t)$ 값의 변화를 보여주며, 표 1은 결핵 발병 확률이 변한 시점인 5000과 마지막 시점 10000에서의 $P(Tub = pre | Visit = ok, \Theta_t)$ 값을 보여준다. 학습률이 4%인 경우에는 시점 5000과 시점 1000에서 제안하는 방법이 Voting EM 알고리즘에 비해 실제값 0.05와 0.4에 더 다가감을 확인하였고, 학습률이 10%인 경우에는 시점 5000에서는 Voting EM과 비슷한 결과를 보이며, 시점 10000에서는 Voting EM 보다 좋은 결과를 나타냈다.

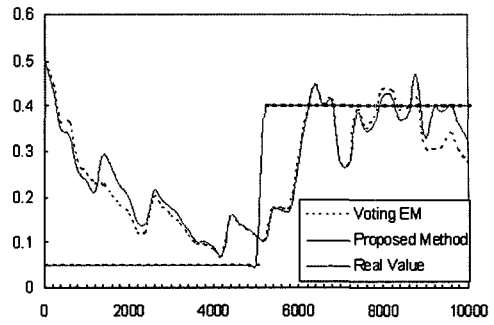
이 밖에 3.2절에서 설명한 바와 같이 데이터 결측률이 0%인 경우에는 Voting EM과 제안하는 방법 모두 동일한 결과를 보였으며, 결측률이 100%인 경우에는 Voting EM의 경우 $P(Tub = pre | Visit = ok, \Theta_t)$ 의 값이 학습되지 않고 0.5를 유지하는 반면, 제안하는 방법에서는 정확하지는 않지만 변화의 패턴을 학습하였다.

5. 결론

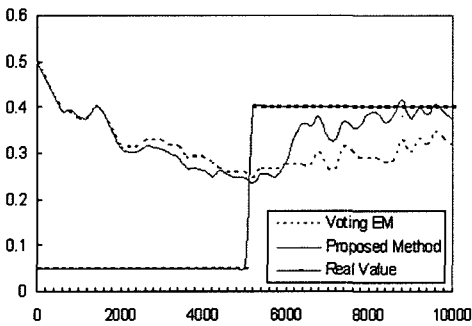
BN의 오프라인 파라미터 학습은 전문가가 직접 BN을 설계시 필요한 시간과 노력을 줄이기 위해 제안되었다. 하지만 이러한 방법들은 학습을 위해서 충분한 양의 데이터가 필요하므로, 데이터가 충분하지 않은 경우에는 충분한 양의 데이터가 모아질 때까지 기다려야 한다. 또한 주어진 데이터를 기반으로 오프라인 상에서 학습하므로 실시간으로 변화하는 환경의 상태나 사용자의



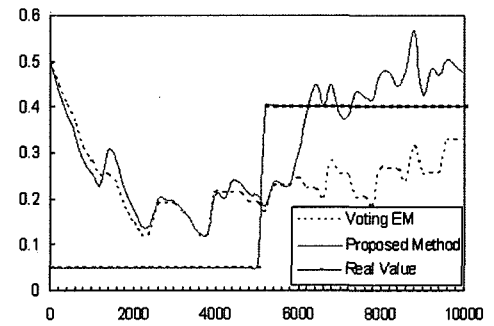
(a) Missing Rate : 30%, Learning Rate : 4%



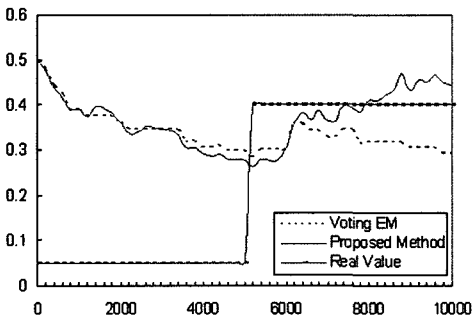
(b) Missing Rate : 30%, Learning Rate : 10%



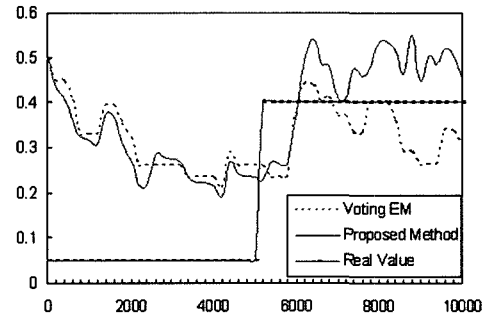
(c) Missing Rate : 50%, Learning Rate : 4%



(d) Missing Rate : 50%, Learning Rate : 10%



(e) Missing Rate : 70%, Learning Rate : 4%



(f) Missing Rate : 70%, Learning Rate : 10%

그림 8 $P(Tub = pre | Visit = ok, \Theta_t)$ 값의 변화

표 1 시점 5000과 10000에서의 $P(Tub = pre | Visit = ok, \Theta_t)$ 값

Learning Rate		4%		10%	
Time	Missing Rate	Voting EM	Proposed Method	Voting EM	Proposed Method
5000	30%	0.206	0.213	0.114	0.116
	50%	0.258	0.247	0.190	0.207
	70%	0.297	0.275	0.260	0.233
10000	30%	0.288	0.344	0.275	0.327
	50%	0.318	0.373	0.330	0.472
	70%	0.292	0.442	0.303	0.453

성향을 파악하는 데에는 한계가 있다.

이러한 오프라인 학습의 한계를 극복하기 위해 Voting EM 알고리즘과 같은 온라인 파라미터 학습 방법이 제

안되었으나, 이 방법은 특정 불완전한 데이터에 대해서 결측값이 나타난 노드를 중심으로 학습이 잘 이루어 지지 않는다는 한계가 있다. 본 논문에서는 Voting EM

알고리즘의 한계점을 보완하기 위해서는 방법을 제안하였다. 제안한 방법은 예측치와 관측치의 차이를 이용한 학습방법으로 불완전한 데이터로부터 학습이 가능하도록 하여 학습의 유연성을 높이고, 실시간 학습을 통해 사용자나 환경의 변화를 잘 모델링한다. Voting EM 알고리즘과 제안하는 방법의 비교 실험 결과, 제안하는 방법의 유용성을 알 수 있었다.

향후 과제로 본 논문에서 수행한 실험의 다각적인 분석과 제안한 방법을 다양한 도메인에 적용하여 그 유용성을 검증할 필요가 있다. 또한 온라인 학습의 효율을 극대화하는 학습률의 동적 적용 방법이 필요하다. 그리고 본 논문에서는 수식 (11)의 많은 계산량을 극복하기 위해서 노드들의 독립성 가정을 통해 수식 (16)을 이용하였는데, 정선 트리[14]를 이용한 추론 등과 같은 방법을 적용하여 연산량을 줄이는 알고리즘이 필요하다.

참 고 문 헌

[1] D. Heckeman, "Bayesian networks for data mining," *Data Mining and Knowledge Discovery*, vol. 1, pp. 79-119, 1997.

[2] H.G. Cooper and E. Herskovitz, "A Bayesian method for the induction of probabilistic networks from data," *Machine Learning*, vol. 9, pp. 159-225, 1994.

[3] W. L. Bunine, "Operations for learning with graphical models," *Journal of Artificial Intelligence Research*, vol. 2, pp. 159-225, 1994.

[4] D. Heckerman, D. Geiger and D.M. Chickering, "Learning Bayesian networks: The combinations of knowledge and statistical data," *Machine Learning*, vol. 20, pp. 197-243, 1995.

[5] R.G. Cowell, A.P. Dawid and P. Sebastiani, "A comparison of sequential learning methods for incomplete data," In *Bayesian Statistics*, vol. 5, pp. 533-542, 1996.

[6] A. Dempster, D. Laird and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, pp. 1-38, 1977.

[7] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 721-741, 1984.

[8] D. Spiegelhalter and S. Lauritzen, "Sequential updating of conditional probabilities on directed graphical structures," *Networks*, vol. 20, pp. 579-605, 1990.

[9] E. Bauer, D. Koller and Y. Singer, "Update rules for parameter estimation in Bayesian networks," *Proceedings of the 13th. Annual Conference on Uncertainty in AI*, pp. 3-13, 1997.

[10] I. Cohen, A. Bronstein and F.G. Cozman, "Online learning of Bayesian network parameters," In Report No. HPL-2001-55, HP Labs, 2001.

[11] I. Cohen, A. Bronstein and F.G. Cozman, "Adaptive online learning of Bayesian network parameters," In Report No. HPL-2001-156, HP Labs, 2001.

[12] S.Z. Zhang, H. Yu, H. Ding, N.H. Yang and X.K. Wang, "An application of online learning algorithm for Bayesian network parameter," *Machine Learning and Cybernetics*, vol. 1, pp. 153-156, 2003.

[13] S.L. Lauritzen and D.J. Spiegelhalter, "Local computations with probabilities on graphical structures and their application to expert systems (with discussion)," *Journal of the Royal Statistical Society, Series B*, vol. 50, pp. 157-224, 1988.

[14] B. D'Ambrosio, "Inference in Bayesian networks," *AI Magazine*, vol. 20, no. 2, pp. 21-36, 1999.



임 성 수

2004년 2월 연세대학교 컴퓨터과학과(학사). 2006년 2월 연세대학교 컴퓨터과학과(석사). 2006년 3월~현재 연세대학교 컴퓨터과학과 박사과정. 관심분야는 인공지능, 지능형 에이전트, 베이지안 네트워크



조 성 배

1988년 연세대학교 전산학과(학사). 1990년 한국과학기술원 전산학과(석사). 1993년 한국과학기술원 전산학과(박사). 1993년~1995년 일본 ATR 인간정보통신연구소 객원 연구원. 1998년 호주 Univ. of New South Wales 초빙연구원. 1995년~현재 연세대학교 컴퓨터과학과 정교수. 관심분야는 신경망, 패턴인식, 지능정보처리