기초 인공지능 : Assignment03(Bayes Nets)

전공: 컴퓨터공학 학년: 3학년 학번: 20201635 이름: 전찬

**0. 목차**

1. 구현 목표

2. 새로운 출력 Factor의 unconditionedVariable과 conditionedVariable 구하는 방법

3. CPT가 주어졌을 때, unconditioned variable을 제거하는 과정 서술

4. output Factor에서 query variable과 evidence dictionary의 역할 또는 의미

**1. 구현 목표**

이번 과제에서는, Bayes Nets을 바탕으로 CPT(Conditional Probability Table)의 크기를 줄일 수 있는 방법을 알아본다. 이를 구현하기 위해 Bayes Net을 구성하며, join Factor과 eliminate을 수행하며, 실제 inference 과정을 구현해야 한다. 이번 보고서에는 문제 2, 3, 4 과정에서의 구현 방법에 관련된 문제를 해결해야 한다.

**2. 새로운 출력 Factor의 unconditionedVariable과 conditionedVariable 구하는 방법**

이 방법은 설명 pdf 파일 상 (2) 에 대한 설명이다. 문제 2번 상 여러 Factors을 join한 결과로 새로운 출력 Factor을 생성해 내기 위해, 각 unconditionedVariable과 conditionedVariable을 파악해야 한다. 이를 위해 사용되는 일반화된 방법은 아래와 같다.

(1) Factors의 각 unconditioned / conditioned을 하나의 집합(python 상 set)으로 저장한다.

(2) conditioned set에서 unconditioned set에 포함되는 원소(variable) 을 제거한다.

(3) 이를 새로운 Factor의 unconditioned / conditioned에 적용한다.

이를 통해

위와 같은 방법을 통해 계산한 각 joinFactors의 unconditioned / conditioned는 아래와 같다.

1. joinFactors(P(A, B, C, D, E, F | K, L, M, N), P(C, D, E, K, M | A, C, F))

(1) unconditioned = {A, B, C, D, E, F, K, M} / conditioned = {K, L, M, N, A, C, F}

(2) conditioned set – unconditioned set = {L, N}

(3) Factor가 의미하는 확률 = P(A, B, C, D, E, F, K, M | L, N)

2. joinFactors(P(U, V, X, Y | Z, M, Q, B), P(Z, M, U, X, Q | Y, V, A, B))

(1) unconditioned = {U, V, X, Y, Z, M, Q} / conditioned = {Z, M, Q, B, Y, V, A}

(2) conditioned set – unconditioned set = {B, A}

(3) Factor가 의미하는 확률 = P(U, V, X, Y, Z, M, Q | B, A)

3. joinFactors(P(X, K, V | L, E, S, F), P(R, S, U | K, V, L))

(1) unconditioned = {X, K, V, R, S, U} / conditioned = {L, E, S, F, K, V}

(2) conditioned set – unconditioned set = {L, E, F}

(3) Factor가 의미하는 확률 = P(X, K, V, R, S, U | L, E, F)

**3. CPT가 주어졌을 때, unconditioned variable을 제거하는 과정 서술**

문제 3에서 구현한 eliminate 함수의 알고리즘은 아래와 같다.

(1) unconditioned / conditioned set을 설정한다.

(2) unconditioned set에서 삭제할 variable을 제거하며, 새로운 Factor을 만들어 낸다.

(3) 새로운 Factor의 각 할당 가능한 값에 대해, 해당 할당 값에서 삭제할 variable을 제거하며 marginalization을 수행한다. 계산한 확률을 해당 Factor에 저장한다.

이를 통해 unconditioned variable을 제거한 새로운 CPT을 구현해낼 수 있다. 또한 이 알고리즘을 위 문제에 적용시키면 아래와 같다.

(1) unconditioned / conditioned set을 설정한다.

unconditioned set = {V, W, L}

conditioned set = {}

(2) unconditioned set에서 삭제할 variable을 제거하며, 새로운 Factor을 만들어 낸다.

unconditioned set = {V, W}

conditioned set = {}

새로 구성한 CPT 는 아래와 같다.

|  |  |  |
| --- | --- | --- |
| V | W | Prob. |
| Taxi | Sun | - |
| Taxi | Rain | - |
| Bus | Sun | - |
| Bus | Rain | - |
| Subway | Sun | - |
| Subway | Rain | - |

<새롭게 만들어 낸 CPT>

(3) 새로운 Factor의 각 할당 가능한 값에 대해, 해당 할당 값에서 삭제할 variable을 제거하며 marginalization을 수행한다. 계산한 확률을 해당 Factor에 저장한다.

각 V, W의 assignment에 대해 marginalization해 구한 결과는 아래와 같다.

|  |  |  |
| --- | --- | --- |
| V | W | Prob. |
| Taxi | Sun | 0.0135 + 0.0315 |
| Taxi | Rain | 0.012 + 0.008 |
| Bus | Sun | 0.162 + 0.378 |
| Bus | Rain | 0.018 + 0.012 |
| Subway | Sun | 0.0945 + 0.2205 |
| Subway | Rain | 0.03 + 0.02 |

<각 CPT의 assignment에 대해 확률을 계산한 결과>

위 과정을 통해, 주어진 CPT에 대해 unconditioned variable을 제거할 수 있다.

**4. output Factor에서 query variable과 evidence dictionary의 역할 또는 의미**

Bayesian Network에서 추론 과정을 수행하는 방식은 크게 2 가지로, 열거(enumeration), 변수 제거(elimination) 가 존재한다. 문제 4는 elimination을 구현하는 것이며, 이는 조건부 확률에서 conditioned 부분이 고정될 때, P(unconditioned | fixed conditioned) 을 구하는 것과 동일하다. 따라서 output Factor을 만들어낼 때, 해당 output은 P(query variables | evidence dictionary) 에 대한 Factor 을 구해내야 한다. 여기에서 query variable과 evidence dictionary는 아래와 같다.

- query variable : inference query에서 unconditioned variable 부분이다. 따라서 주어진 evidence에 대해, 조건부 확률로 알고 싶어하는 확률 값이라고 할 수 있다.

- evidence dictionary : inference query에서 fixed conditioned variable 부분이다. 여기에서 fixed라는 말은 해당 conditioned variable이 다양한 값을 가질 수 있는 형태가 아니라, 각 variable key에 대해, 한 value 값만 저장함을 의미한다. 따라서 구현에서 여러 번 사용한 variableDomainsDict 와 달리 해당 dictionary는 각 value 값으로 가능한 할당 값 string을 저장하고 있다.