💜

CH7: Bayesian Network

7.1 Recap on Probability

7.1.1 Probability

확률은 어떤 일이 일어날 가능성을 측량하는 단위로 비율이나 빈도로 나타난다. 확률은 수학적 확률과 경험적 확률로 분류된다. 수학적 확률은 모든 경우의 수에 대해 그 일이 일어날 확률을 계산한 것이다. 반대로, 경험적 확률은 실제 그 일이 무수히 반복디었을 때 나타나는 확률로 기존의 경험을 바탕으로 한 추측 값이다.

> It is the relative frequency with which an outcome would be obtained if the process where repeated a large number of times under similar conditions

7.1.2 Conditonal Probability

조건부 확률은 한 사건이 일어났다는 전제 하아서 다른 사건이 일어날 확률을 뜻한다. 예를 들어, 사건 B가 일어나는 경우에 사건 A가 일어날 확률을 '사건 B에 대한 A의 조건부확률'이라고 하고 $P(A|B)$라고 쓴다.

7.1.3 Joint Probability

결합확률은 사건 A와 B가 동시에 발생할 확률이다. 즉, $P(A \cap B), P(A, B)$로 나타낼 수 있다.

7.1.4 Computing with Probabilities

(1) Law of Total Probabilities

전체 확률의 법칙은 한 사건의 확률을 표본공간의 파티션을 이용하여 표현하는 방법이다. 수식적으로는 다음과 같다.

표본공간 S를 n개로 분할하여 사상 $\{B_1, B_2, ..., B_n\}$을 얻는다. $P(B \neq 0)$일때, S의 임의의 사건 A에 대해 다음이 성립한다. 이때, 모든 사건들은 서로 베타적이고 모두 합쳐서 전체가 되어야 한다. (MECE, Mutually Exlusive Collectively Exhaustive)

$$P(A) = \sum_B P(A, B) = \sum_B P(B)P(A|B)$$

위와 같은 경우를 b가 random variable일때, B에 대해 summing out, marginalize 한다고 한다. 식을 살펴보면, A를 제외한 모든 변수들을 marginal out 하고 있다는 것을 확인할 수 있다. 이 조건은 변수가 더 많아지더라도 사용할 수 있다. 이때 사용하는 것은 'marginal probability'라는 것을 유의해두자.

$$P(C|B) = \sum_A \sum_B P(A, C, D|B) = \frac{1}{P(B)} \sum_A \sum_B P(A, B, C, D)$$

(2) Chaine Rule of Factorization

알다시피, 결합 확률은 많은 정보를 가지고있고 계산하기 어렵다. 따라서 체인 룰을 사용해서 식을 다음과 같이 조금 더 쉽게 만들 수 있다.

$$P(A, B, C, ..., Z) = P(A|B, C, ...Z)P(B, C, ..., Z) = P(A|B, C, ..., Z)P(B|C, ..., Z)...P(Z)$$

이는 베이즈 정리의 확장판이라고 생각할 수도 있는데, 복수의 사건 $X_1, X_2, ..., X_N$에 대한 조건부 확률을 다음과 같이 쓸수도 있다. 물론 위와 같은 식이지만, 곱의 형태로 나타낸 것이다.

$$P(X_1, ..., X_N) = P(X_1) \prod_{i=2}^{N} P(X_i|X_1, ..., X_{i-1})$$

7.1.5 Independence

Conditional Independce와 Marginal Independce는 모두 독립을 의미하지만 둘 사이에는 미묘한 차이가 있다.

(1) **Conditional Independnce**

조건부 독립은 $P(A|B, C) = P(A|B)$인 경우이다. 조건부 독립은 full joint probability를 구할 때, factorization에 사용되기 때문에 중요하다.

(2) **Marginal Independence**

Marginal Independence는 $P(A|B) = P(A)$인 경우이다. 예시를 통해 알아보자. 어떤 중령이 두 장교 A, B에게 명령이 내리는 경우에 대해 생각해보자. 중령의 명령을 알 지 못하는 경우에는 A와 B는 독립적이지 않다. 왜냐하면, 한 장교의 행동을 보고 다른 장교가 따라할 수 있기 때문이다. 하지만, 둘 다 명령을 아는 경우에는, 한 장교가 어떤 행동을 취하여도 중령의 명령을 알고 있기 때문에, 다른 장교의 행동에 영향을 미치지 않는다.

$$P(\text{장교 A의 행동}|\text{장교 B의 행동, 중령의 명령}) = P(\text{장교 A의 행동}|\text{ 중령의 명령})$$

7.2 Bayesian Network

Probabilistic graphical model이란, 확률 분포를 해석하기 위한 그래프 도식 방법이다. 이 모델이 같은 장점은 다음과 같다.

- 확률 모델의 구조를 아주 쉽게 시각화하여 새로운 모델을 디자인하는데 도움을 준다
- 그래프화된 구조를 분석함으로써 모델 속성을 직관적으로 알 수 있다 (e.g. 조건부 분포의 독립성 쉽게 알 수 있음)
- 복잡한 학습과 추론 과정을 가지는 모델의 계산 과정을 그래픽적인 요소로 표현이 가능하다.

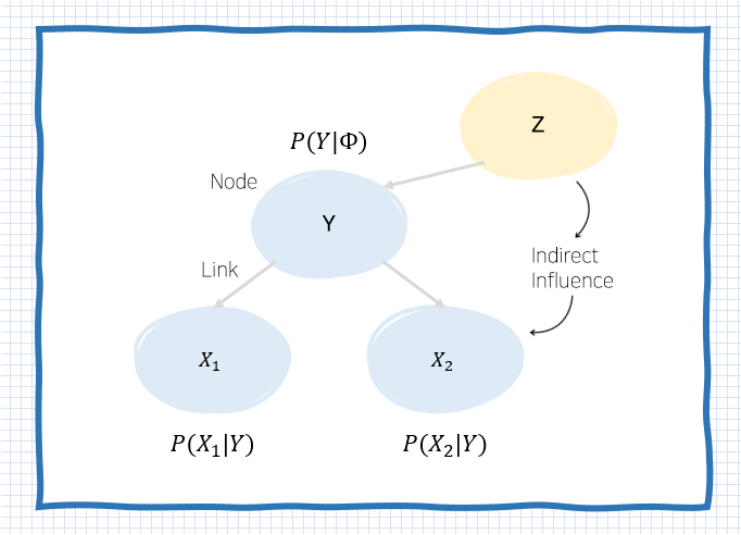
베이즈 네트워크는 Probabilistic graphical model중 하나로 방향성 그래프 모델(directed graphical models)이다. 여기서 방향성이란, 링크가 양방향이 아닌 한쪽 방향으로의 관계만 성립한다는 것이다. ~~뭔소린지 모르겠죠..? 정상입니다...~~

본격적으로 베이지안 네트워크에 대해 알아보기전에 용어부터 정리하고 가자.

- 일반적으로 그래프는 노드(node)와 엣지(edge)로 표현되잠, 확률 그래프에서는 조금 의미가 다름

- 노드(Node): 랜덤 변수(random variable)을 하나의 노드로 표현
- 링크(Link): 엣지와 동일한 의미로, 랜덤 변수 사이의 확률적인 관계를 나타냄

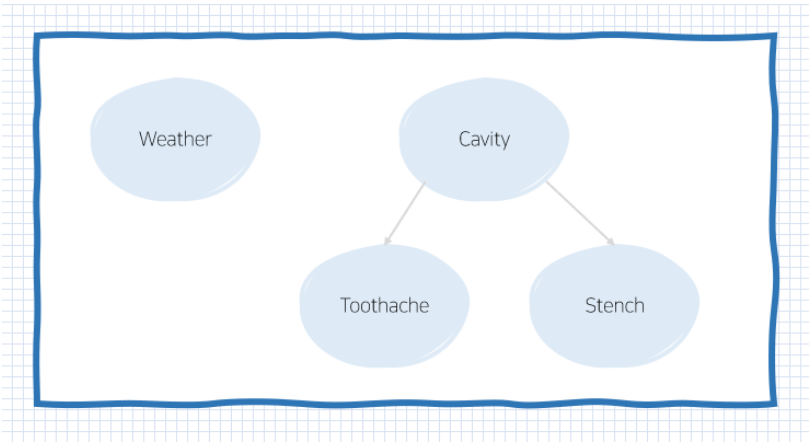
그래서 정확히 베이지안 네트워크란 것이 무엇일까? 현재 많은 문제에 대해 딥러닝 기반의 모델들이 좋은 성능을 내고 있지만, 이들은 블랙박스 모델이기 때문에 설명력이 부족하다. 즉, 모델이 예측은 잘 하는데, 어떻게 그런 결과를 얻게 되는지는 모르겠다는 것이다. 딥러닝 모델이 많은 발전을 이루기 전에도 많은 기법들이 연구되고 있었는데, 그 중 주목받던 모델이 베이지안 네트워크이다. 베이지안 네트워크는 **확률 변수 간의 인과 관계를 그래프로 나타낸 후, 주어진 데이터에 대해 확률 변수의 분포를 학습하는 확률 그래프 모형** (Probabilistic graphical model, PGM)이다. 즉, 그래프 표기를 통해 확률 변수간의 관계에 따른 conditional independence를 나타낸 것이다! 이 표기를 통해 full joint distribution을 쉽게 얻을 수 있다.



위의 그림은 베이지안 네트워크의 예시이다. 이를 살펴보면, 베이지안 네트워크는 확률 변수간의 관계를 한 방향으로 나타내고, 비순환(Acyclic) 구조를 가지고 있음을 알 수 있다. 그리고 앞에서 정의했듯이, 각 노드의 직접 영향을 주는 것은 부모 노드 밖에 없다. 즉, 부모 노드와 아이 노드는 조건부 독립 관계인 것이다! 간접적인 영향 또한 줄 수 있다. 위의 그림에서 나타있다 싶이, Y의 부모노드인 Z는, Y의 링크를 통해 X_1과 X_2에 간접적인 영향을 주고 있다.

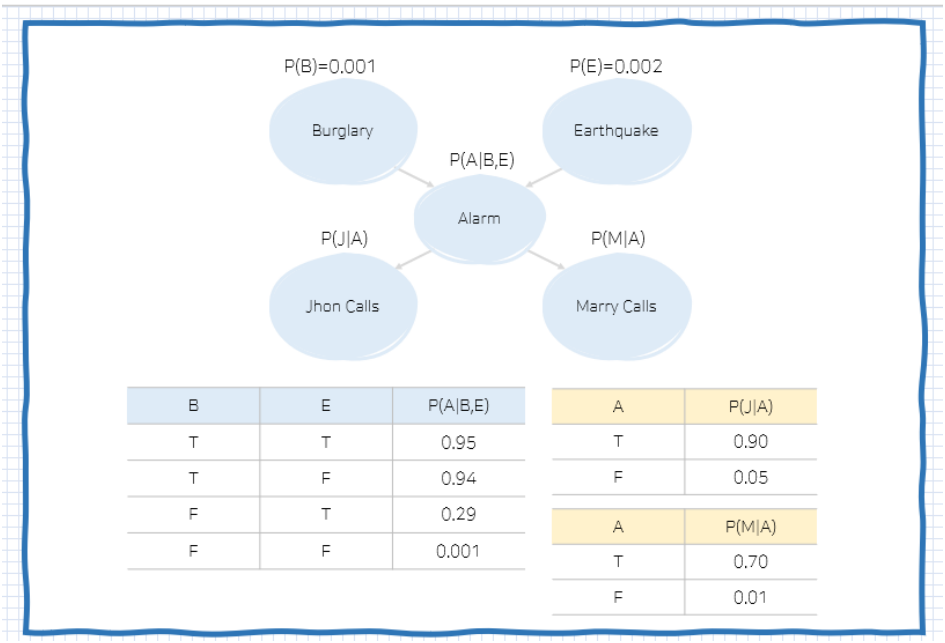
7.2.1 Interpretation of Bayesian Network

보통 네트워크의 토폴로지(Topology)는 일반 상식을 이용하거나 해당 분야의 전문가의 의견을 통하여 구성한다. 여기서 토폴로지란, 네트워크의 요소들을 물리적으로 연결해 놓은 것, 또는 그 연결 방식을 뜻한다. 아래와 같은 네트워크 토폴로지를 해석해보자.



먼저 Weather은 다른 변수와 연결되어 있지 않기 때문에 다른 변수에게 아무런 영향을 주기 못하기 때문에 다른 변수와 독립이다. 그래프에서 보이다시피, Cavity는 Toothache와 Stench에 직접적으로 연결되어 있기 때문에, 직접적인 영향을 준다는 것을 알 수 있다. 반대로, Toothache와 Stench는 Cavity가 주어질 때, 조건부 독립이다. 즉, Cavity에 대한 정보를 얻었을 때, Tootache와 Stench는 서로 독립이다.

다른 예시를 한 번 살펴보자. 일을 나가 있을 때, 도둑이 들거나 지진이 났을 때 알람이 울린다. 이때, 옆집에 살고있는 존 또는 메리가 전화를 준다. 이 상황을 다음과 같은 베이지안 토폴로지로부터 아래의 표와 같이 CPD(Conditional Probability Distribution)을 만들 수 있다. CPD를 이용하여 다양한 상황의 확률을 구할 수 있는데, 아래의 그림처럼, 알람이 울렸을 때, 도둑이 들었을 확률을 구할 수 있다.



B	E	P(A|B,E)
T	T	0.95
T	F	0.94
F	T	0.29
F	F	0.001

A	P(J|A)
T	0.90
F	0.05

A	P(M|A)
T	0.70
F	0.01

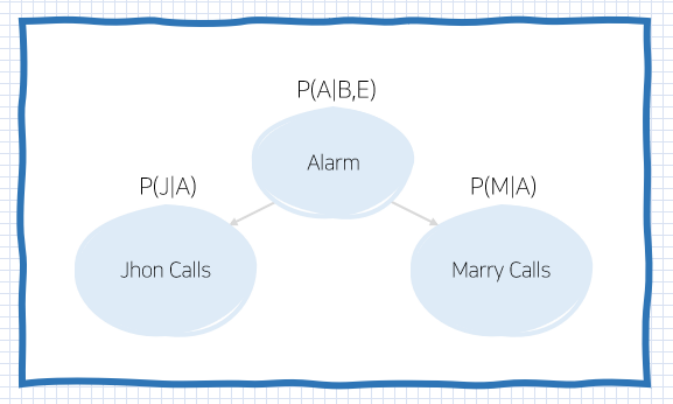
베이지안 네트워크는 Qualitative components와 Quantitative components로 구성되어 있다. Qualitative components는 베이지안 토폴로지의 형태로 각 변수간의 관계에 대한 사전정보 또는 데이터를 통한 학습하는 방법을 이용하여 구성한다. 즉, 구조적인 내용을 설명하고 있다. 반대로, Quantitative components는 조건부확률을 나타내는 테이블로, 각 노트에 주어진 확률 분포 값(CPD)를 의미한다.

7.2.2 Typical Local Structures

다음과 같은 예를 생각해보자. 메리가 전화를 했을 때, 집에 도둑이 들었을 확률은 어떻게 구할까? 메리의 전화에 관한 CPD만을 이용해서는 구할 수 없기 때문에, 베이지안 네트워크의 Quantitative와 Qualitative components 둘 다 이용하여 구할 수 있다.

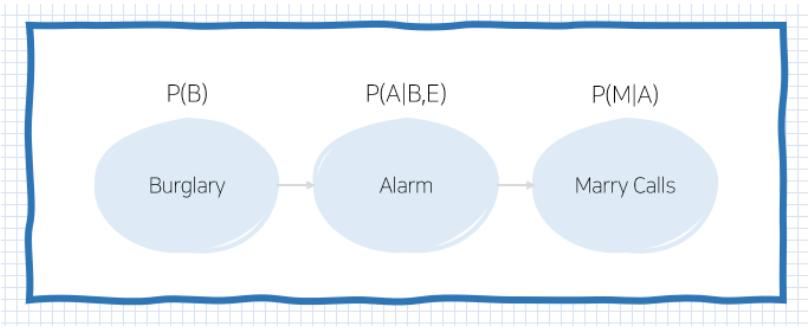
이를 알기위해, 베이지안 네트워크 구성하는 3가지의 대표적인 구조에 대해 알아보자.

(1) Common Parent



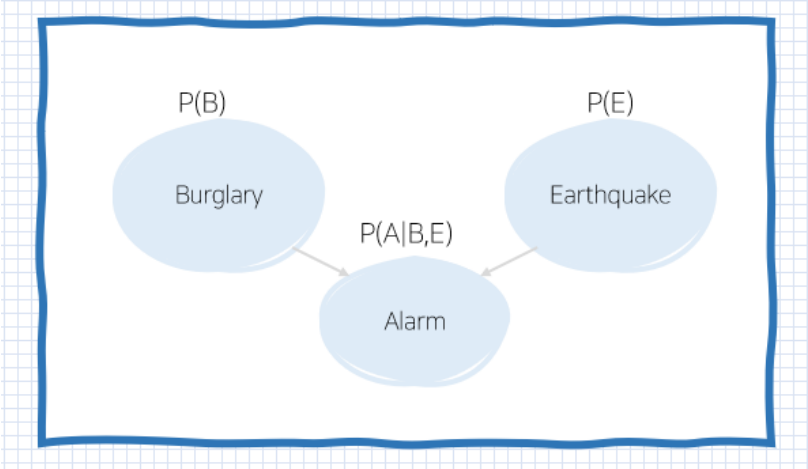
Common Parent는 두 개의 자식 노드가 하나의 부모 노드를 가지는 경우로, 부모 노드의 정보를 알고 있다면, 두 자식 노드는 서로 독립이다. 위 예시에 적용하다면, Alarm이 공통부모가 되고, Jhon Calls와 Marry Calls가 자식노드이다. 이 둘은 앞서 말했듯이, $J \perp M|A$ 관계를 가진다. 즉, 이를 수식적으로 나타내면, $P(J, M|A) = P(J|A)P(M|A)$가 된다.

(2) Cascading



Cascading은 3개의 노드가 순차적으로 연결된 경우로, 선형적인 관계를 가진다. 이때, 가운데에 위치한 노드의 정보를 알게 되면 양 끝의 노드는 서로 독립이다. 위 예시를 적용하게 된다면, Alarm에 대한 정보가 주어진다면, Buglary의 정보는 Marry Calls에 아무런 정보를 주지 않는다. 즉, Buglary와 Marry Class는 독립이 된다. 즉, $B \perp M|A$이기 때문에, $P(M|B, A) = P(M|A)$가 된다. (약간 Markov Chain에서 전전 시점이 현재 시점에 영향을 안주는 것과 유사함)

(3) V-structure



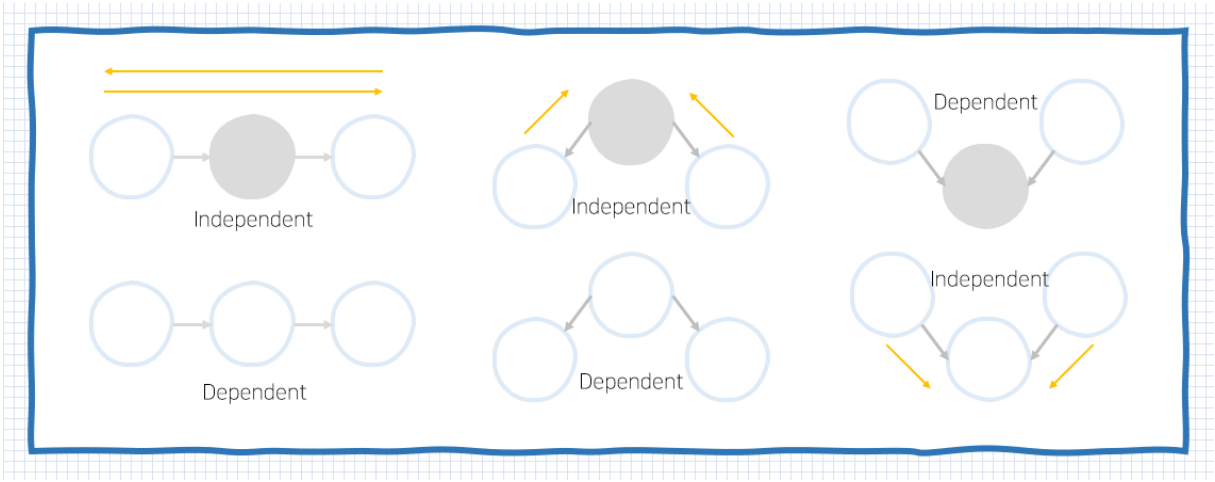
V-structure은 Cascading과 반대로, 하나의 노드가 두 개의 부모 노드를 가지는 경우로, 앞의 두 구조와는 다르게, 자식 노드의 정보를 알면 두개의 부모 노드가 종속이 된다. 즉, 어떤 정보를 알면 관계가 생기는 경우이다. 이를 위에 예시에 적용하면, Alarm이 울렸을 때, Earthquake가 발생하지 않았다면, 이는 Alarm이 발생한 원인이 Earthquake에 있지 않나는 추가 정보를 Buglary에 제공하게 된다. 따라서, 더이상 Buglary와 Earthquake가 독립이지 않다. 이를 수식적으로 나타내면, $\sim (B \perp E|A)$이되기 때문에, $P(B,E,A) = P(B)P(E)P(A|B,E)$로 나타낼 수 있다.

7.2.3 Bayes Ball Algorithm

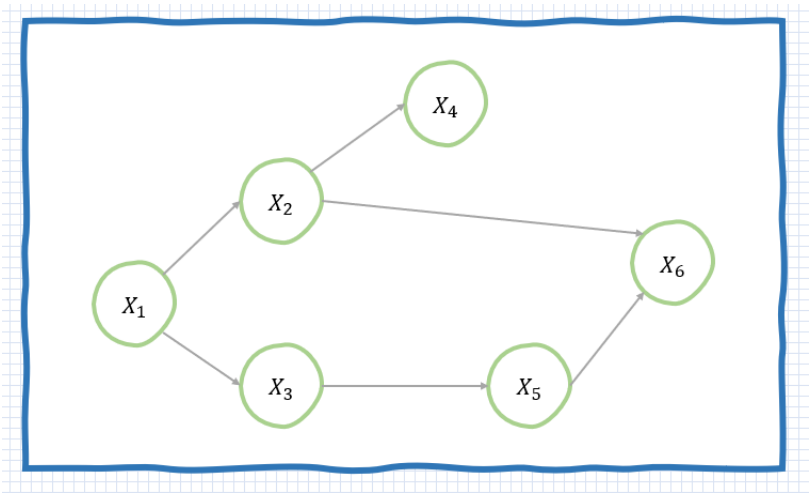
Bayes ball 알고리즘은 가상의 공을 굴려 베이즈 네트워크에서 굴러가는지 안 굴러가는지에 따라 노드간에 독립을 판단하는 알고리즘이다. 이 알고리즘의 목적은 $X_A \perp X_B | X_c$을 확인하는 것이다. 이 부분을 알아보기 전에 상황을 정리하고 가자.

- 회색으로 칠한 부분은 X_c노드이다
- 공은 X_A로부터 굴러간다
- 모든 공은 bayes ball algorithm을 따른다
- 우리는 공이 X_B까지 굴러가는지를 알아보고자 한다.

주어진 네트워크 구조가 Common parent 또는 Cascading인 경우, 가운데 노드에서 정보가 주어지지 않으면, 가상의 공이 굴러가고, V-struture인 경우에는 반대로 가운데 노드의 정보가 주어져야 공이 굴러가게 된다. 즉, 노드들의 관계가 독립인 경우에는 공이 굴러가지 않는 것이다! 따라서, 공이 굴러가는 경로에 있는 노드의 관계는 종속으로 판단할 수 있다.



아래와 같은 상황이 주어졌을 때, 각 상황에 대해서 어떤 경우에 독립인지 알아보자.



- $X_1 \perp X_4 | \{X_2\}$: Independent

  X_1, X_2, X_4는 세개의 노드가 순차적으로 연결된 경우이고, X_2가 주어졌기 때문에, Cascading한 상황이다. 따라서, X_1과 X_4는 독립이다.

- $X_2 \perp X_5 | \{X_1\}$: Independent

  먼저, X_2와 X_5가 독립이기 위해서는 2가지 경우를 모두 만족해야 한다. 먼저, X_1을 기준으로, X_2와 X_5는 common parent를 가지기 때문에 독립이다. 그 다음, X_2와 X_5는 V-structure를 가지고 있지만, X_6의 정보는 알지 못하기 때문에 독립이다. 따라서 $X_2 \perp X_5 | \{X_1\}$이 독립이 되는 것이다.

- $X_1 \perp X_6 | \{X_2, X_3\}$: Independent

  이 관계에서는, 위와 아래가 모두 cascading한 경우로 독립이 된다.

- $X_2 \perp X_3 | \{X_1, X_6\}$: Dependent

  이 경우는 X_6이라는 공통 자식 노드가 생겼음으로 V-structure의 조건이 만족된다. 따라서, 두 개의 부모 노드인, $X_2 \& X_3$이 종속이 된다.

특정 노드에서 Blanket에 해당하는 노드의 정보가 제공되는 경우, 특정 노드와 Blanket에 포함되지 않은 노드의 관계는 독립이다. 이 blanket을 **Markov blanket**이라고 한다. 이때, Blanket이란, 특정 노드의 부모 노드와 자식 노드, 그리고 자식 노드의 부모 노드로 구성된다. 이때, 자식 노드의 부모 노드가 필요한 이유는 V-structure에 따른 정보 전달을 막기 위함이다.

▼ ✨ **Markov Blanket 좀 더 있어보이게 설명해보자** ✨

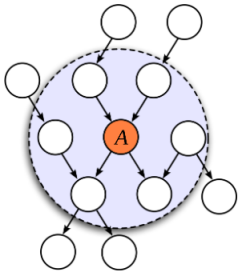
D개의 노드를 가진 바향 그래프로 표현된 결합 분포 $p(x_1, ..., x_D)$를 고려하자.
변수
x_i 즉, 어떤 노드의 조건부 분포를 고려하는데 남은 노드의 $x_{i \neq j}$가 조건인 경우 다음과 같이 식을 쓸 수 있다.

$$P(X_i | X_{j \neq i}) = \frac{p(X_1, ..., X_D)}{\int p(X_1, ..., X_D) d_{X_i}} = \frac{\prod_k P(X_k | p_{a_k})}{\int \prod_k P(X_k | p_{a_k}) d_{X_i}}$$

위의 식을 정리하면 변수 x_i에 종속성이 없는 요소 $P(X_k | p_{a_k})$는 x_i의 적분식에서 사라진다.

노드 x_i 자신인 조건부 분포 $P(X_k | p_{a_k}), x_i \in p_{a_k}$인 노드 x_k를 위한 조건부 분포 $P(X_k | p_{a_k}), (X_i \in p_{a_k})$를 x_i중심으로 도식화하면 아래의 그림과 같다. 보라색 원이 A 노드에 대한 Markov Blanket이다. 네드튀크의 모든 노드가 조건부 독립을 만족한다면, 특정 노드 B에 대하여 A 노드는 다음과 같은 식을 가진다.

$$P_r(A|MB(A), B) = P_r(A|MB(A))$$



이와 유사한 개념으로는 **D-Seperation**이 있다. D-Seperation은 특정 노드에 대한 정보가 주어졌을 때, 시작 노드에서 Bayes Ball Algorithm을 이용하여 공을 굴려서 목적 노드에 도달하지 못하는 경우를 의미한다.

> X is d-seperated (directly seperated) from Z given Y ($X \perp Z|Y$) if we cannot send a ball from any node in X to any node in Z using the Bayes ball algorithm

7.2.4 Factorization of Bayesian Network

베이지안 네트워크의 factorization은 full joint distribution을 구할 때, 개별 노드의 조건부 확률의 조건에 포함되는 노드를 각 노드의 부모 노드만 고려함으로써 계산에 사용되는 파라미터를 줄여주는 역할을 한다. 즉, 이는 우리가 앞서 알아봤듯이, full joint distribution보다 conditionally independent한 관계가 있을 때 연산량이 줄어드는 특징을 사용한 것이다. 예를 들어 $X_1 \perp X_3|X_2$인 경우 $P(X_1, X_2, X_3)$을 구해보자. 이는 다음과 같이 두 가지 경우로 쓸 수 있다.

$$P(X_1, X_2, X_3) = P(X_1|X_2, X_3)P(X_2|X_3)P(X_3) = P(X_1|X_2)P(X_2)P(X_3)P(X_3)$$

식에서도 볼 수 있듯이 $P(X_1|X_2, X_3)$을 계산하기 위해서는 4개의 파라미터가 필요하다. 반대로, $P(X_1|X_2)$는 2개의 파라미터가 필요하다. 여기서 말하는 파라미터란, discrete binary random variable에 대한 파라미터로 0/1중 1이 나올 확률인 p다. 따라서, $P(X_1|X_2)$는 두가지 파라미터 (p_1, p_2)를 가진다. 두가지 파라미터는 각각 $p_1 = P(X_1 = 1|X_2 = 0)$과 $p_2 = P(X_1 = 1|X_2 = 1)$을 의미한다. 마찬가지로 $P(X_1|X_2, X_3)$은 (p_1, p_2, p_3, p_4)를 추론 변수로 가지게 된다.

쉽게 말하면, 여기서 말하는 파라미터란 1이 나올 경우의 수인 것이다..! 참고로 condition이 더 적게 붙으면 붙을수록 추론해야 하는 파라미터의 개수는 exponentially 줄어들게 된다. 따라서, 조건이 더 적은 $P(X_1|X_2)$가 더 빠르다는 것을 알 수 있다. 따라서 우리는 factorization을 통해 연산량을 빠르게 할 수 있는 것이다.

> 🤔 **그런데 왜 연산량이 줄어드는 것일까?**
>
> 확률 자체를 계산하기 위해서는 각 변수의 확률을 구하여 곱해줘야하지만, 저장해야 확률값을 생각하면 달라진다. Full joint distribution의 경우에는 각 변수가 가질 수 있는 모든 경우에 대한 확률값을 저장해야하기 때문에, 곱하는 방식이 필요하다. 하지만,
> $X \rightarrow Y \rightarrow Z$와 같이 conditional independence를 가정한다면, $P(X, Y, Z) = P(X)P(Y|X)P(Z|Y)$로 계산되기 때문에 $P(X), P(Y|X), P(Z|Y)$가 무엇인지 알면 $P(X, Y, Z)$를 쉽게 계산할 수 있으며, 각 값에 대한 경우의 수의 덧셈이 되는 것이다.

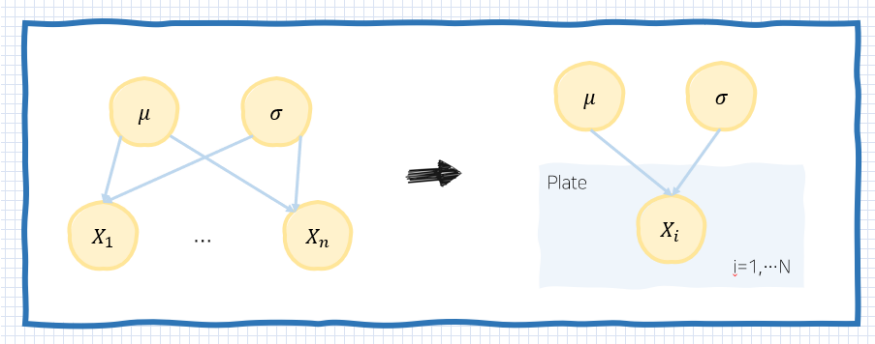
따라서, 부모 노드의 기준으로 factorize하게 되면, 다음과 같이 식을 쓸 수 있다. 이때, X_{π_i}는 X_i의 부모 노드의 set이다.

$$P(X) = \prod_i P(X_i|X_{\pi_i})$$

여기서 유의해야할 점은 베이즈 네트워크의 factorization은 부모 노드만 보면 된다는 것이다...! 따라서 $P(X_1, X_2, X_3) = P(X_1|X_2)P(X_2|X_3)P(X_3)$으로 쓸 수 있는 것이다. ~~어디서 많이 본것 같지 않나..? 당연하다.. 베이즈 배우면 이것만 한다...~~

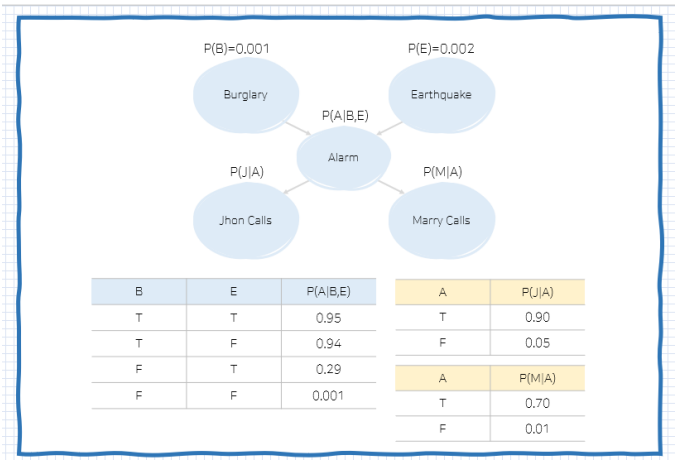
7.2.5 Plate Notation

Random Variable이 많은 경우, 그래프를 하나하나 그리는 것은 시간과 자원 낭비다. 그래서 plate notation을 사용하여 그래프를 간편화하는 것이다. 아래는 $X_1, ..., X_n$개의 변수에 대한 가우시안 모델을 Plate notation으로 표현한 예시이다. Plate Notation 'for-loop'을 그림을로 나타내는 것이라고 생각하면 편하다.



7.3 Inference on Bayesian Networks

앞서 베이지안 네트워크로부터 도둑(Buglary)이 들고, 메리의 전화(Marry Calls)가 왔을 때, 알람을 울렸을 확률을 conditional proability로부터 구할 수 있다. 이를 수식적으로 나타내면



B	E	P(A|B,E)
T	T	0.95
T	F	0.94
F	T	0.29
F	F	0.001

A	P(J|A)
T	0.90
F	0.05

A	P(M|A)
T	0.70
F	0.01

용어 정리

- $X = \{X_1, ..., X_N\}$: 모든 random variable
- $X_V = \{X_{k+1}, ..., X_N\}$: 관심 변수 (evidence variable)
- $X_H = X - X_V = \{Y, Z\}$: hidden variable
  - Y: Interested hidden variables
  - Z: Uninterested hidden variables

Hidden variable X를 관심 대상인 Y와 관심 대상이 아닌 Z로 나눌 수 있다. 이때, Evidence가 주어질 때, Conditional probability를 구하기 위해서는 아래의 general form의 유도과정의 마지막 식과 같이 full joint distribution의 형태로부터 구할 수 있도록 식을 만들줘서 계산해야 한다. 즉, full join distribution의 형태로 만든 식에서 Marginalize out을 통해 conditional probability를 구할 수 있다.

7.3.1 Most Probable Assignment

$$\begin{aligned} P(Y|x_v) &= \sum_z P(Y, Z=z|x_v) \\ &= \sum_z \frac{P(Y,Z,x_v)}{P(x_v)} \\ &= \sum_z \frac{P(Y,Z,x_v)}{\sum_{y,z} P(Y=y, Z=z, x_v)} \end{aligned}$$

베이지안 네트워크를 이용하면, Evidence가 주어졌을 때, 가장 일어날만한 사건을 conditional probability로부터 각 사건에 대한 확률을 계산하여 구할 수 있다. 이를 most probable assignment라고 부른다.

Most probable assignment를 통해 다음과 같은 2가지 응용이 가능하다.

- Evidence가 주어졌을 때, 가장 높은 확률을 가진 사건을 예측 결과로 정하여 예측을 진행 $(B, E \rightarrow A)$
- 결과가 주어졌을 때, 가장 일어날 만한 원인을 구함으로써 결과에 대한 진단 진행 $(A \rightarrow B, E)$

7.3.2 Marginalization and Elimination

앞서 확인했듯이, 결합 확률을 계산하려면, 많은 연산량이 요구된다. 위의 베이지안 네트워크에서 $P(a=true, b=ture, mc=true)$는 우리가 관심없는 변수, JC와 E에 대해 marginalize out을 통해 구할 수 있다. 이는 다음과 같이 쓸 수 있다.

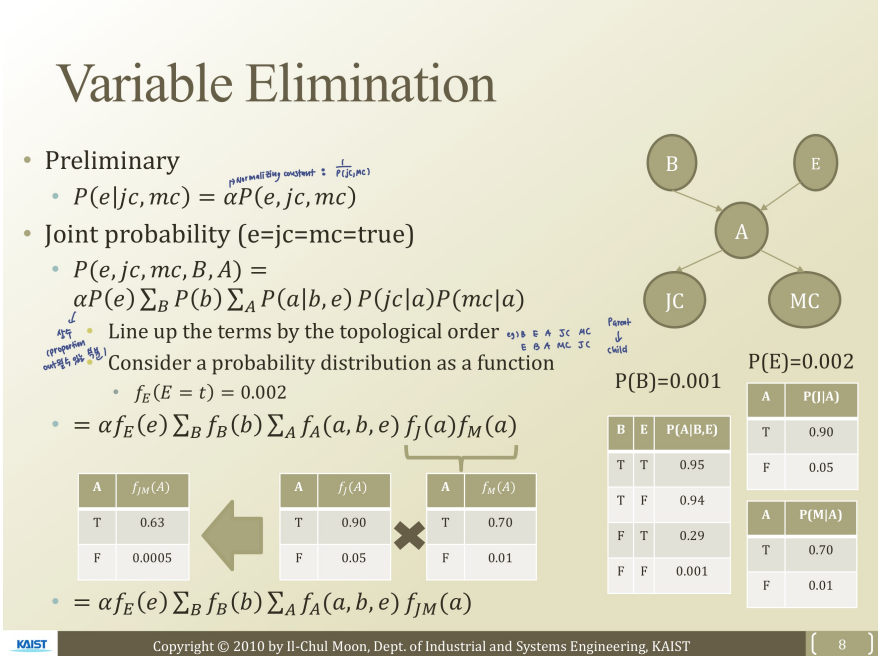
$$\begin{aligned} P(a=true, b=true, mc=true) &= \sum_{JC}\sum_{E} P(a,b,E,JC,mc) \\ &= \sum_{JC}\sum_{E} P(JC|A)P(mc|a)P(a|b,E)P(E)P(b) \end{aligned}$$

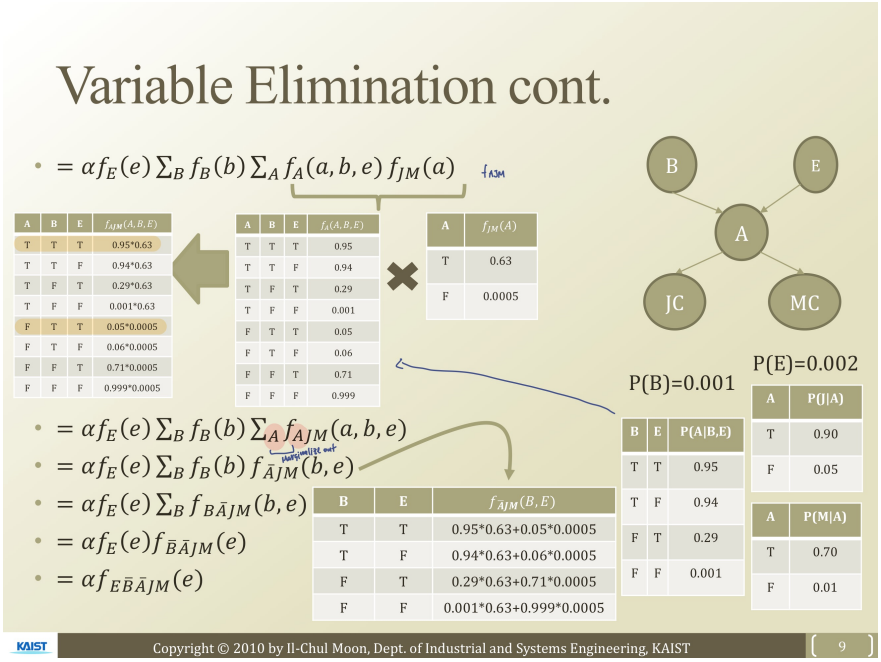
식을 변형하기 전보다는 계산량이 줄었지만, JC나 E가 없는 항에 대해서도 계속해서 덧샘과 곱샘을 진행을 하기 때문에 여전히 계산량이 많다. 계산량을 줄이기 위해, marginalize out의 관심 변수가 포함되지 않은 probability를 옮길 수 있다. 즉, 아래의 식과 같이 관심 변수가 포함되지 않은 probability가 각 변수에 대한 summation 과정에 포함되지 않게 만들어 전체 곱의 연산량을 줄어들게 만들면 된다.

$$\begin{aligned} P(a=true, b=true, mc=true) &= \sum_{JC}\sum_{E} P(a,b,E,JC,mc) \\ &= P(b)P(mc|a)\sum_{JC} P(JC|A)\sum_{E} P(a|b,E)P(E) \end{aligned}$$

7.3.3 Variable Elimination

Variable Elimination은 서로 다른 확률이 같은 변수에 종속되어 있고, 해당 변수에 대하여 marginalization을 수행할 때, 두 확률의 테이블(벡터)을 곱하여 하나의 테이블(벡터)로 만들어 주는 방법으로, 이 방법을 통해서도 계산량을 줄일 수 있다. 예시를 통해 더 자세히 알아보자…!

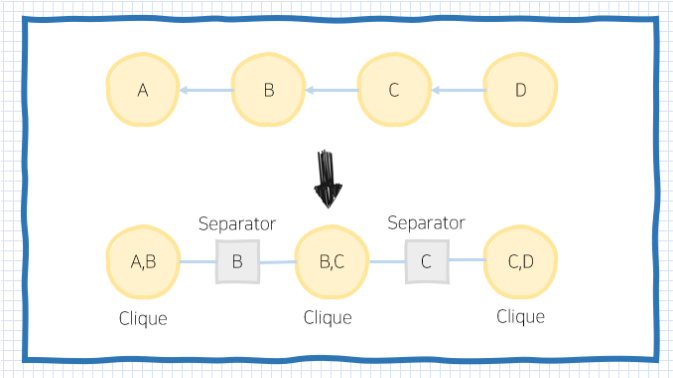




7.3.4 Potential Functions

Potential function은 잠재적으로 확률이 되는 함수로써, 아직 확률 조건을 만족하기에는 충분하지 않은 함수를 의미한다. 만약, 조건을 만족하게 된다면, pdf가 된다. 이 함수는 뒤에 등장할 belief propagation을 통해 확률로 변하게 된다. 이 내용은 나중에 알아보자…

> A function which is not a probability function yet, but once normalized it can be a probability distribution function



위의 예시는 potential function을 설명하기 위한 베이지안 네트워크로 Clique와 Seperator로 나타낼 수 있다. Clique는 해당 노드 간의 fully connected된 상태를 말한다 (Fully connected subset). Seperator은 연결된 Clique가 공통으로 가지고 있는 요소(random variable)을 의미한다. 즉, seperator을 통해 clique가 연결된다.

이를 수식적으로 나타내보면 다음과 같다.

- Potential function on nodes (Clique): $\psi(a,b), \psi(b,c), \psi(c,d)$
- Potential function on links (Seperator): $\phi(b), \phi(c)$

베이지안 네트워크의 full joint distribution을 구하기 위해 모든 Clique의 potnetial function의 곱에 모든 potential function의 곱으로 나눈 경우로 정의를 하겠다. Clique와 Seperator로 정의된 full joint distribution을 실제 factorization된 결과와 동일하게 되도록 potential function을 정의해야 한다. 아래는 이 과정을 2가지 방법으로 정의했다.

첫 번째 방법은 Clique의 potential function을 conditional distribution으로 정한 경우이다.

$$P(A,B,C,D) = P(U) = \frac{\prod_N \psi(N)}{\prod_L \phi(L)} = \frac{\psi(a,b)\psi(b,c)\psi(c,d)}{\phi(b)\phi(c)}$$

$$\psi(a,b) = P(A|B), \psi(b,c) = P(B|C), \psi(c,d) = P(C|D)P(D)$$
$$\phi(b) = 1, \phi(c) = 1$$

두 번째 방법은 joint distribution을 통해 정의하였다.

$$P(A,B,C,D) = P(U) = \frac{\prod_N \psi(N)}{\prod_L \phi(L)} = \frac{\psi^*(a,b)\psi^*(b,c)\psi^*(c,d)}{\phi^*(b)\phi^*(c)}$$

$$\psi^*(a,b) = P(A,B), \psi^*(b,c) = P(B,C), \psi^*(c,d) = P(C,D)$$
$$\phi^*(b) = P(B), \phi(c)^* = P(C)$$

Seperator의 potential function을 고려했을 때, 두 번째 방법이 합리적인 것처럼 보이지만, 실제로 우리가 가지고 있는 정보는 conditional distribution에 대한 정보이기 때문에, 두 번째 방법에 적용하는 것은 쉽지 않다.

7.3.5 Absorption in Clique Graph

만약, A에 데이터가 추가되어 update된다면, $\psi(A,B)$의 Marginalizaiton을 통해 얻어진 $\phi(B)$도 바뀔 것이다. 바뀐 $\phi(B)$를 통해 $\psi(B,C)$도 바뀔 것이고, 바뀐 $\psi(B,C)$는 다음 seperator을 update하는데 적용될 것이다. 이러한 과정의 반복을 통해 추가된 데이터 A의 데이터에 대한 Belief propagation이 진행된다. 이 방법을 Absorption rule이라고 한다.

이를 조금 더 자세히 살펴보자, 먼저 다음과 같은 가정을 하자

- $P(B) = \sum_A \psi(A,B)$
- $P(B) = \sum_C \psi(B,C)$

- $P(B) = \phi(B)$

ψ에서 ϕ를 찾는 과정은 다음과 같다.

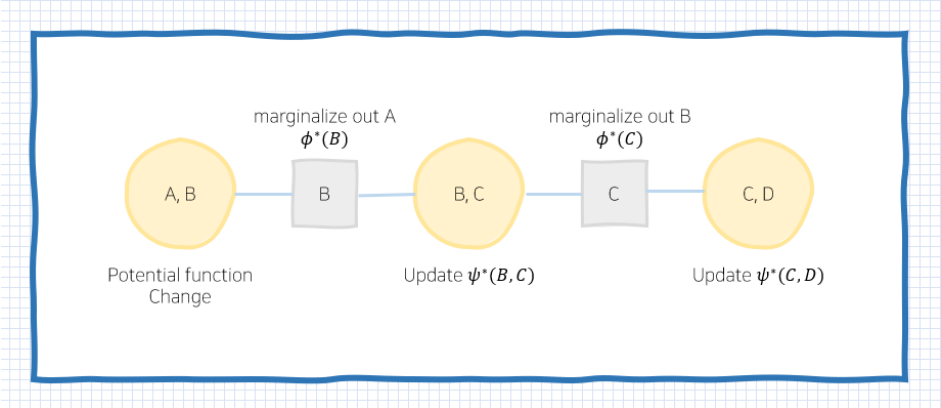
1) When the ψs change by the observations: $P(A, B) \rightarrow P(A = 1, B)$

2) A single ψ change cna result in the change of multiple ψs

3) The effect of the observation propagates thorugh the clique graph

This is Belieft Propagation!

이를 그림과 수식으로 나타내면 다음과 같다.



1) Absorption (update rule)

2) Assume $\psi^*(A, B), \psi(B, C)$ and $\phi(B)$

3) Define the update rule for seperators: $\phi^*(B) = \sum_A \psi^*(A, B)$

4) Define the update rule for cliques: $\psi^*(B, C) = \psi(B, C)\frac{\phi^*(B)}{\phi(B)}$

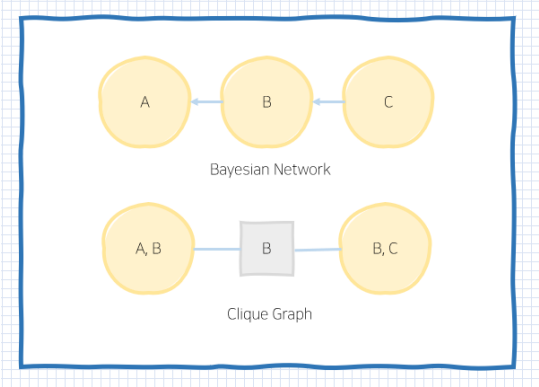
~~이 과정 미묘하게 익숙하지 않은가...? 맞다.. 이 과정 MCMC와 하나의 파라미터를 계속 업데이트한다는 점에서 상당히 유첫 번째 예~~

Back propagation을 통해 바뀐 Clique 간의 potential function의 local consistency가 성립한다는 것은 아래의 증명 과정을 통해 확인할 수 있다. 이는 absorption rule의 적용이 끝난 뒤에 베이지안 네트워크의 global consistency를 보장한다.

$$\begin{aligned}\sum_C \psi^*(B, C) &= \sum_C \psi(B, C)\frac{\phi^*(B)}{\phi(B)} \\ &= \frac{\phi^*(B)}{\phi(B)} \sum_C \psi(B, C) \\ &= \frac{\phi^*(B)}{\phi(B)}\phi(B) \\ &= \sum_A \psi^*(A, B)\end{aligned}$$

7.3.6 Belief Propagation

이 부분에서는 potential function을 set up하여 belief propagation이 진행되는 과정을 알아볼 것이다. 앞에서 말했듯이, 이것이 가능한 이유는, 각 노드에서 얻을수 있는 확률이 conditional distribution이기 때문이다.



다음과 같은 초기정보를 바탕으로 다양한 예시를 알아보자.

- $\psi(a,b) = P(a|b), \psi(b,c) = P(b|c)P(c)$
- $\phi(b) = 1$

첫 번째 예시는 아무런 정보가 주어지지 않았을 때의 $P(b)$를 구하는 과정이다 ($P(b) = ?$). 앞서 1로 세팅해둔 $\phi(b)$가 belief propagation을 통해 $P(b)$로 바뀐 것을 알 수 있다. 또 다시 belief propagation을 진행하여도, 여전히 $P(b)$인 것을 확인할 수 있다. 즉, 이는 local consistency가 성립되는 경우이다.

$$\phi^*(b) = \sum_a \psi(a,b) = 1$$
$$\psi^*(b,c) = \psi(b,c)\frac{\phi^*(b)}{\phi(b)} = P(b|c)P(c) = P(b,c)$$
$$\phi^{**}(b) = \sum_c \psi(b,c) = \sum_c P(b,c) = P(b)$$
$$\psi^*(b,c) = \psi(a,b)\frac{\phi^{**}(b)}{\phi^*(b)} = \frac{P(a|b)P(b)}{1} = P(a,b)$$
$$\phi^{***}(b) = \sum_a \psi^*(a,b) = \sum_a P(a,b) = P(b)$$

두 번째 예시에서는 a와 c에 대한 정보가 주어졌을 때, most probable한 b값을 찾는 과정이다 ($P(b|a=1,c=1) = ?$). 첫 번째 예시와의 다른 점은 새로운 potential function인 δ를 통해 해를 구한다는 것이다. 두 번째 belief porpagation을 통해 $\phi(b)$가 수렴한다는 것을 확인할 수 있고, 모든 $\phi(b)$가 conditional distribution을 통해 나타낼 수 있다는 것을 알 수 있다. 따라서 joint probability를 통해 계산하는 것보다는 효율적으로 계산할 수 있다.

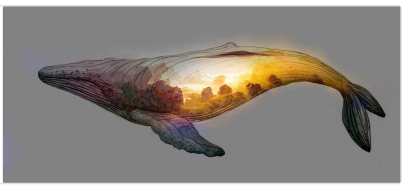
$$\phi^*(b) = \sum_a \psi(a,b)\delta(a=1) = P(a=1|b)$$
$$\psi^*(b,c) = \psi(b,c,)\frac{\phi^*(b)}{\phi(b)} = P(b|c=1)P(c=1)\frac{P(a=1|b)}{1}$$
$$\phi^{**}(b) = \sum_c \psi(b,c)\delta(c=1) = P(b|c=1)P(c=1)P(a=1|b)$$
$$\psi^*(a,b) = \psi(a,b)\frac{\phi^{**}(b)}{\phi^*(b)} = P(a=1|b)\frac{P(b|c=1)P(c=1)P(a=1|b)}{P(a=1|b)} = P(b|c=1)P(c=1)P(a=1|b)$$
$$\phi^{***}(b) = \sum_a \psi^*(a,b)\delta(a=1) = P(b|c=1)P(c=1)P(a=1|b)$$

▼ ✨ **참고사이트 & 추가자료** ✨

Bayesian Network

baysian networks 2020-05-26 베이지안 네트워크(Bayesian networks) 베이지안 네트워크란? 그래프 모델로서 조건부 확률 분포를 확률 방향성으로 나타낸다. 베이지안 네크워크의 제약은 확률 방향성이 사이클(Cycle)을 형성 하지 않는데 있다. 베이지안 네트워크의 핵심은 다음과 같다. 변수들 간의 확률적 의존 관계를 나타내는 방

https://5bluewhale.tistory.com/6



아래 사이트 한 번 읽어보면 좋을 듯 (유용한 내용 많음)

0. Graphical Model

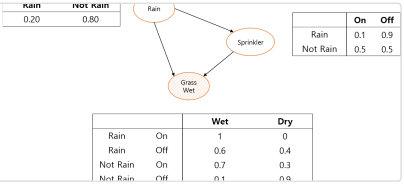
http://norman3.github.io/prml/docs/chapter08/0

베이지안 네트워크는 확률 변수의 **인과 관계를 그래프로 나타내고 분석**하는 확률 모형입니다. 현재 많은 주목을 받는 딥러닝과 비교했을 때 설명력(Interpretability)에서 강점을 가지지만, 모델의 주관이 반영된 모델이므로 주관에 따라 모델의 성능이 달라집니다. 베이지안 네트워크를 통해 데이터에 대해 여러가지 추론 문제에 답을 할 수 있습니다.
현재까지도 많이 사용되는 모델인 Gaussian Mixture Model (GMM)이나 Hidden Markov Model (HMM)은 모두 베이지안 네트워크의 종류입니다. 다음 글을 통해서 이 두 모델에 대해 각각 알아보겠습니다.

Introduction to Bayesian Network (Bayesian Network vol 1)

Introduction to Bayesian Network (Bayesian Network vol 1) Written by JunKeon Park 현재 많은 문제(Task)에 대해 딥러닝 기반의 모델들이 훌륭한 성능을 내고 있어 학계와 산업계에서 딥러닝이 큰 각광을 받고 있습니다. 그러나 아직까지 딥러닝 모델은 블랙 박스 모델이라는 평가와 함께 설명력(Interpretability)이 부족하다

https://tmaxai.github.io/post/intro_bayes/



이것도 읽어보면 좋음…!

graphical-models.pdf