

Artificial Intelligence

人工智能

第5章 不确定性知识表示 和推理

概率图模型

课件来源：中山大学刘咏梅教授；多伦多大学Sheila McIlraith教授；
浙江大学吴飞教授；海军工程大学贲可荣教授；Chris Bishop等

不确定知识表示和推理

□ 背景

□ 概率论与图论基础

□ 概率图模型的表示

□ 概率图模型的推理

□ 概率图模型的学习

背景

- 人工智能系统的表示与推理过程实际上就是一种思维过程。其中，推理是从已知事实出发，通过运用相关的知识逐步推出某个结论的过程。人们常常对原因和结果的推理很感兴趣。比如，我感冒症状减轻了，是不是因为服用了维生素C片导致的？但是，由于事件都带有随机性，导致看似直截了当的问题，却不容易回答。例如，我可能仅仅喝白开水，感冒也会自己消失。
 - **已知事实**(证据)，用以指出推理的出发点及推理时应使用的知识；
 - **知识**是推理得以向前推进，并逐步达到最终目标的依据。
- 按照所用知识的确定性，可以分为确定性和不确定性两种类别。
 - **确定性推理**是建立在经典逻辑基础上的，经典逻辑的基础之一就是集合论。这在很多实际情况中是很难做到的，如高、矮、胖、瘦就很难精确地分开；
 - **不确定性推理**就是从不确定性初始证据出发，通过运用不确定性的知识，最终推出具有一定程度的不确定性但却是合理或者近乎合理的结论的思维过程。

背景

- 常识(common sense)具有不确定性。
 - 一个常识可能有众多的例外, 一个常识可能是一种尚无理论依据或者缺乏充分验证的经验。
- 常识往往对环境有极强的依存性。
 - “鸟是会飞的”
- 把指示确定性程度的数据附加到推理规则, 并由此研究不确定强度的表示和计算问题。
- 处理数据的不精确和知识的不确定所需要的一些工具和方法, 包括:
 - **基于Bayes理论的概率推理**
 - 基于可信度的确定性理论
 - 基于信任测度函数的证据理论
 - 基于模糊集合论的模糊推理等

背景

□ 不确定知识表示和推理方法

■ 确定性理论

- 该理论由Shortliffe提出，并于1976年首次在血液病诊断专家系统MYCIN中得到了成功应用。
- 在确定性理论中，不确定性是用可信度来表示的。

■ 证据理论

- 用于处理不确定性、不精确以及间或不准确的信息。
- 引入了信任函数来度量不确定性，引用似然函数来处理由于“不知道”引起的不确定性。

■ 模糊逻辑和模糊推理

- 模糊集合论是1965年由Zadeh提出的，随后，他又将模糊集合论应用于近似或模糊推理，形成了可能性理论。
- 模糊逻辑可以看作是多值逻辑的扩展。模糊推理是在一组可能不精确的前提下推出一个可能不精确的结论。

背景

- 不确定知识表示与推理是人工智能的核心模块之一，其理论基础包括概率论和可能性理论等。
 - 概率论处理的是由随机性引起的不确定性；
 - 可能性理论处理的是由模糊性引起的不确定性。
- 李德毅院士在统一主观认知和客观现象中的随机性和模糊性方面提出了不确定性人工智能的研究问题。不确定性人工智能认为，随机性和模糊性常常是联系在一起的，在人类思维 and 智能行为中难以区分并独立存在，研究不确定性需要研究随机性和模糊性之间的关联性。
- 本讲主要以概率论为基础，介绍概率图模型的表示、推理和学习离等内容。

背景

□ 概率图模型

- 概率论与图论结合的产物, 为统计推理和学习提供了一个统一的灵活框架;
- 概率图模型用节点表示变量, 节点之间的边表示局部变量间的概率依赖关系。系统的联合概率分布表示为局部变量分布的连乘积, 该表示框架不仅避免了对复杂系统的联合概率分布直接进行建模, 而且易于引入先验知识。

□ 概率图模型统一了目前广泛应用的许多统计模型和方法。

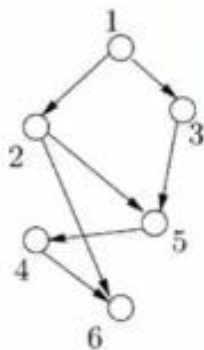
- 马尔科夫随机场(MRF)、条件随机场(CRF)
- 隐马尔科夫模型(HMM)、多元高斯模型
- 卡尔曼滤波、粒子滤波、变分推理等



- Judea Pearl(朱迪亚·佩尔), 贝叶斯网络之父, 加州大学洛杉矶分校计算机科学学院教授、认知系统实验室主任。2011年, 因人工智能概率方法和因果推理算法获得图灵奖。

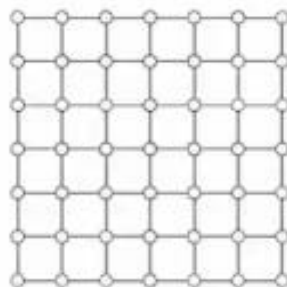
背景

□ 概率图模型例子



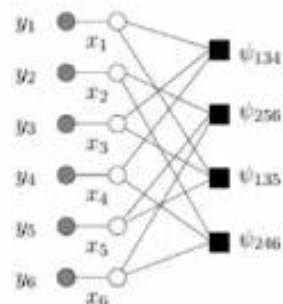
贝叶斯网

$$p(\mathbf{x}) = \prod_s p_s(x_s | x_{\pi(s)})$$



马尔科夫随机场

$$p(\mathbf{x}) = \frac{1}{Z} \prod_p \phi_p(x_p) \prod_{(p,q)} \phi_{pq}(x_p, x_q)$$



因子图

$$p(\mathbf{x}) = \frac{1}{Z} \prod_c \phi_c(\mathbf{x}_c)$$

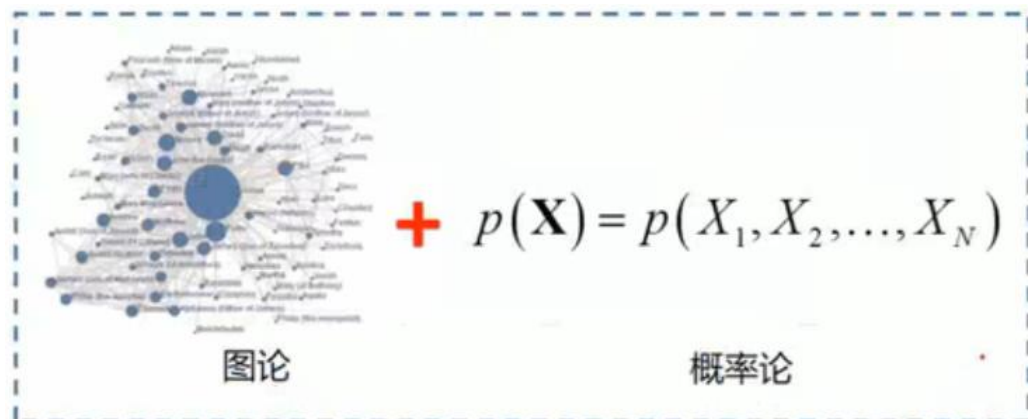
□ 概率图模型发展历程

- 历史上，曾经有来自不同学科的学者尝试使用图的形式表示高维分布的变量之间的依赖关系；
- 在人工智能领域，概率方法始于构造专家系统的早期尝试
- 在20世纪80年代末，在贝叶斯网络和一般的概率图模型中的推理取得重要进展。1988年，Pearl提出了信念传播 (Belief Propagation, BP) 算法，把全局的概率推理过程转变为局部变量间的消息传递，从而大大降低了推理的复杂度
- BP算法引起了国际上学者的广泛关注，掀起了新一轮的研究热潮。
- 如今，概率图模型的推理和学习已经广泛应用于机器学习、计算机视觉、自然语言处理、医学图像处理、计算神经学、生物信息学等研究领域，成为人工智能研究中不可或缺的一门技术。

背景

□ 概率图模型研究内容

概率图模型表示



概率图模型推理

$$p(\mathbf{x}_\alpha) = \sum_{\mathbf{x} \setminus \mathbf{x}_\alpha} p(\mathbf{x})$$
$$\mathbf{x}^* = \arg \max_{\mathbf{x} \in \mathcal{X}} p(\mathbf{x})$$

求边缘概率、最大后验
概率状态等

概率图模型学习

$$D = \{X_1^{(i)}, X_2^{(i)}, \dots, X_N^{(i)}\}_{i=1}^M$$

数据

背景

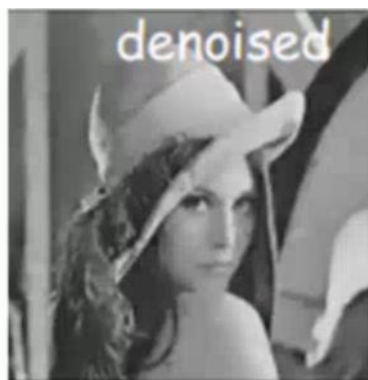
□ 概率图模型应用——图像视觉分析



图像分割



立体视觉



图像去噪



姿态估计

背景

□ 概率图模型应用——医学诊断

Applet started

ON STAGE ESSENTIALS COMMUNICATE FIND

OnParenting
May 14 - May 20, 1997

Fidelity Investments
Fidelity Distributors Corporation

Our home on the web [is where] click here

cover contents news experts fun handbook talk find help feedback

There are two ways to search for specific information in **OnParenting**. In **Find by Word**, type the word(s) you want to find and get a list of titles relevant to that word. **Find by Symptom** will help you get information about children's symptoms. [Help](#) has tips to target your search.

Find by Word
Find by Symptom ▶

Describe the child
in the drop-down boxes at the right. Relevant information will appear below.

Age: Sex:
Complaint:

Localized pain: Can the child localize, or point to, the site of the pain?

- ☐ No, unable to localize
- ☐ Below the navel to the child's left
- ☐ Above the child's navel
- ☐ Either of the child's sides
- ☐ Below the navel to the child's right
- ☐ Above the navel to the child's right
- ☐ Above the navel to the child's left
- ☐ Don't Know

Results so far

Disorder	Relevance
Viral gastroenteritis	<div><div></div></div>
Psychosomatic pain	<div><div></div></div>
Urinary tract infection	<div><div></div></div>
Other	<div><div></div></div>

Start Over **Review**
Next **Finish**

□ 概率图模型应用——计算神经学

- 在计算神经学领域, 研究表明, 大脑具有表示和处理不确定信息的能力。大量的生理和心理实验发现, 大脑的认知处理过程是一个概率推理过程。
- Ott和Stoop建立了二值马尔科夫随机场信念传播算法和神经动力学模型的联系, 证明了连续Hopfield网络的方程可以用BP算法的消息传递迭代方程得到。因此, 马尔科夫随机场中的BP算法可以由神经元实现, 每个神经元对应于MRF的一个节点, 神经元之间的突触连接对应于节点之间的依赖关系。
- 现有人工智能技术的两种主流“大脑”
 - 一种是支持人工神经网络的深度学习加速器, 基于研究“**电脑**”的计算机科学, 让计算机运行机器学习算法;
 - 另一种是支持脉冲神经网络的神形态芯片, 基于研究“**人脑**”的神经科学, 无限模拟人来大脑。

不确定知识表示和推理

- 背景
- 概率论与图论基础
- 概率图模型的表示
- 概率图模型的推理
- 概率图模型的学习

概率论与图论基础

□ 频率论学派

- 事件的概率是当我们无限次重复试验时, 事件发生次数的比值
- 投掷硬币、掷骰子等

□ 贝叶斯学派

- 将事件的概率视为一种主观置信度
- 我认为明天下雨的概率是30% ; 他认为明天下雨的概率是80%

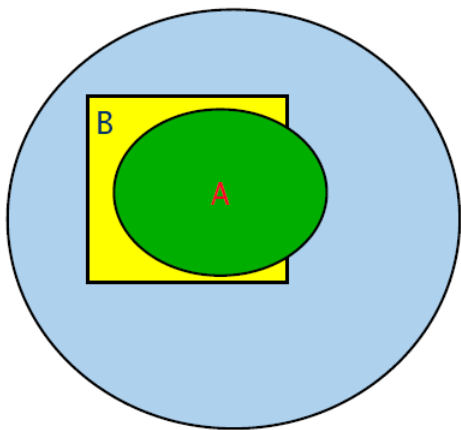


Thomas Bayes (约1701-1761), 英国数学家。约1701年出生于伦敦, 1742年成为英国皇家学会会员。他首先将归纳推理法用于概率论基础理论, 并创立了贝叶斯统计理论, 对于统计决策函数、统计推断、统计的估算等做出了贡献。

概率论与图论基础

□ 条件概率

- 假设变量 A 表示一个事件的集合, 且 $Pr(A) > 0$
- 在变量 A 发生的前提下, 另一个事件的集合(用变量 B 表示)发生的概率记为条件概率 $Pr(B|A)$



假设 B 覆盖了整个事件集合空间的30%, 且覆盖了 A 的80%, 那么 $Pr(B) = 30\%$, 且 $Pr(B|A) = 80\%$

如果 $Pr(B|A) = Pr(B)$, 则 B 和 A 独立

如果 $Pr(B|A, C) = Pr(B|A)$, 则给定 A 的前提下, B 和 C 条件独立

概率论与图论基础

□ 乘法/链式法则

联合概率 $Pr(A,B)=Pr(A)Pr(B|A)=Pr(B,A)=Pr(B)Pr(A|B)$

$Pr(A,B_1,B_2,B_3)=Pr(A)Pr(B_1|A)Pr(B_2|A,B_1)Pr(B_3|A,B_1,B_2)$

概率论与图论基础

□ 加法法则

$$\begin{aligned} Pr(A) &= Pr(A, B) + Pr(A, B^c) \\ &= Pr(B)Pr(A|B) + Pr(B^c)Pr(A|B^c) \end{aligned}$$

$$\begin{aligned} Pr(A) &= \sum_B Pr(A, B) = \sum_{i=1}^n Pr(A, B_i) \\ &= \sum_{i=1}^n Pr(A|B_i)Pr(B_i) \end{aligned}$$

• 贝叶斯定理

$$\begin{aligned} Pr(B|A) &= \frac{Pr(A, B)}{Pr(A)} \\ &= \frac{Pr(B)Pr(A|B)}{Pr(A)} \\ &= \frac{Pr(B)Pr(A|B)}{Pr(A, B) + Pr(A, B^c)} \end{aligned}$$

概率论与图论基础

□ 加法法则

$$\begin{aligned} Pr(A) &= Pr(A, B) + Pr(A, B^c) \\ &= Pr(B)Pr(A|B) + Pr(B^c)Pr(A|B^c) \end{aligned}$$

$$\begin{aligned} Pr(A) &= \sum_B Pr(A, B) = \sum_{i=1}^n Pr(A, B_i) \\ &= \sum_{i=1}^n Pr(A|B_i)Pr(B_i) \end{aligned}$$

• 贝叶斯定理

后验概率

$$Pr(B|A) = \frac{Pr(A, B)}{Pr(A)}$$

先验概率

$$= \frac{Pr(B)Pr(A|B)}{Pr(A)}$$

似然度

标准化常量

$$= \frac{Pr(B)Pr(A|B)}{Pr(A, B) + Pr(A, B^c)}$$

概率论与图论基础

- 假设有一盒骰子，里面有4面的（点数为1、2、3、4），6面的、8面的、12面的、20面的均匀骰子各1个。如果我随机从盒子中选一个骰子，投掷它得到了点数5。那么我选中的骰子为4面、6面、8面、12面、20面的概率各是多少？

概率论与图论基础

- 假设有一盒骰子，里面有4面的（点数为1、2、3、4），6面的、8面的、12面的、20面的均匀骰子各1个。如果我随机从盒子中选一个骰子，投掷它得到了点数5。那么我选中的骰子为4面、6面、8面、12面、20面的概率各是多少？

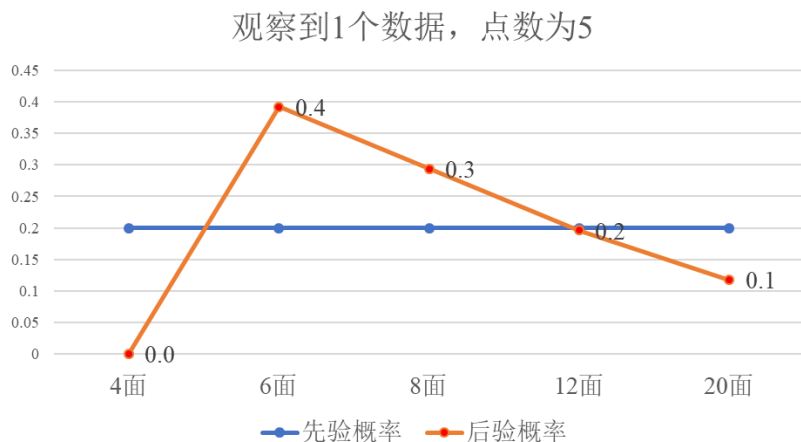
$$Pr(\text{骰子}=4 \mid \text{点数}=5) = \frac{Pr(\text{骰子}=4)Pr(\text{点数}=5 \mid \text{骰子}=4)}{Pr(\text{点数}=5)} = \frac{0.2 \times 0}{Pr(\text{点数}=5)}$$

$$Pr(\text{骰子}=6 \mid \text{点数}=5) = \frac{Pr(\text{骰子}=6)Pr(\text{点数}=5 \mid \text{骰子}=6)}{Pr(\text{点数}=5)} = \frac{0.2 \times \frac{1}{6}}{Pr(\text{点数}=5)}$$

...

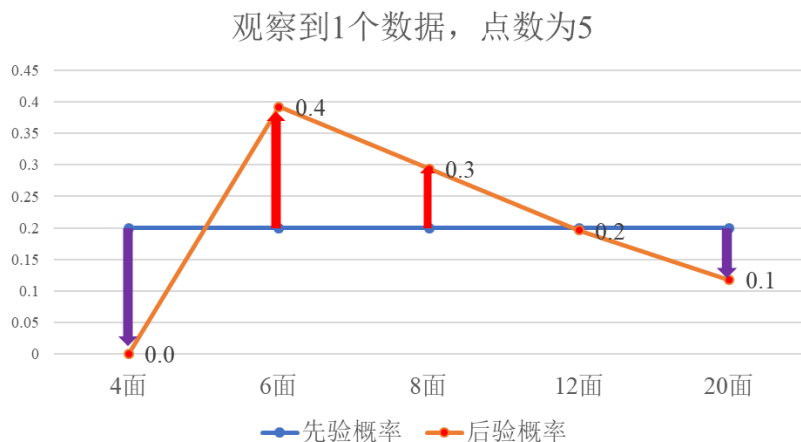
概率论与图论基础

- 假设有一盒骰子，里面有4面的（点数为1、2、3、4），6面的、8面的、12面的、20面的均匀骰子各1个。如果我随机从盒子中选一个骰子，投掷它得到了点数5。那么我选中的骰子为4面、6面、8面、12面、20面的概率各是多少？



概率论与图论基础

- 假设有一盒骰子，里面有4面的（点数为1、2、3、4），6面的、8面的、12面的、20面的均匀骰子各1个。如果我随机从盒子中选一个骰子，投掷它得到了点数5。那么我选中的骰子为4面、6面、8面、12面、20面的概率各是多少？



置信度发生改变！

概率论与图论基础

- **例1.1.1** 一种诊断某癌症的试剂, 经临床试验有如下记录: 癌症病人试验结果是阳性的概率为 95%, 非癌症病人试验结果是阴性的概率为 95%. 现用这种试剂在某社区进行癌症普查, 设该社区癌症发病率为 0.5%, 问某人反应为阳性时该如何判断他是否患有癌症.

概率论与图论基础

- **例1.1.1** 一种诊断某癌症的试剂, 经临床试验有如下记录: 癌症病人试验结果是阳性的概率为 95%, 非癌症病人试验结果是阴性的概率为 95%. 现用这种试剂在某社区进行癌症普查, 设该社区癌症发病率为 0.5%, 问某人反应为阳性时该如何判断他是否患有癌症.

- 设 A 表示“反应为阳性”的事件, B 表示“被诊断者患癌症”的事件, 则 $B_1 = B$ 和 $B_2 = \bar{B}$ 构成完备事件群. 由题意

$$P(A|B_1) = 0.95, \quad P(A|B_2) = 1 - 0.95 = 0.05,$$

$$P(B_1) = 0.005, \quad P(B_2) = 0.995.$$

- 现在要算的是 $P(B_1|A)$ 和 $P(B_2|A)$. 由贝叶斯公式易得

$$P(B_1|A) = \frac{P(A|B_1)P(B_1)}{P(A|B_1)P(B_1) + P(A|B_2)P(B_2)} = 0.087 = 8.7\%$$

$$P(B_2|A) = 1 - 0.087 = 0.913 = 91.3\%.$$

- 某人真正患癌症的可能性很小, 只有8.7%, 告诉他不必紧

张, 可以到医院去做进一步的检查, 以便排除这一疑点.

概率论与图论基础


三个概念

- 联合概率分布
- 边缘概率分布
- 最大后验概率状态

- 联合概率分布: $p(I, D, G)$


I	D	G	Prob.
i^0	d^0	g^1	0.126
i^0	d^0	g^2	0.168
i^0	d^0	g^3	0.126
i^0	d^1	g^1	0.009
i^0	d^1	g^2	0.045
i^0	d^1	g^3	0.126
i^1	d^0	g^1	0.252
i^1	d^0	g^2	0.0224
i^1	d^0	g^3	0.0056
i^1	d^1	g^1	0.06
i^1	d^1	g^2	0.036
i^1	d^1	g^3	0.024

- 边缘概率: $p(I) = \sum_{D, G} p(I, D, G)$


$$\begin{cases} p(I = i^0) = 0.6 \\ p(I = i^1) = 0.4 \end{cases}$$

$$\begin{aligned} p(i^0) &= \sum_{D, G} p(i^0, D, G) \\ &= p(i^0, d^0, g^1) + p(i^0, d^0, g^2) + p(i^0, d^0, g^3) \\ &\quad + p(i^0, d^1, g^1) + p(i^0, d^1, g^2) + p(i^0, d^1, g^3) \end{aligned}$$

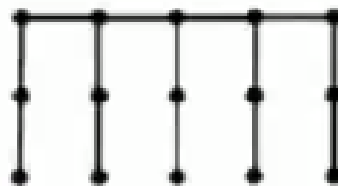
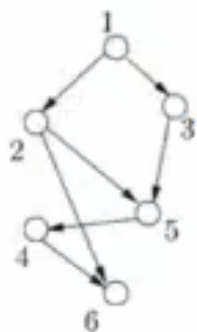
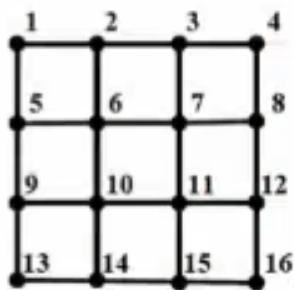
- 最大后验概率状态: $\{I^*, D^*, G^*\} = \arg \max_{I, D, G} p(I, D, G)$


$$\begin{cases} I^* = i^1 \\ D^* = d^0 \\ G^* = g^1 \end{cases}$$

概率论与图论基础

□ 图论基础

- 图定义：由“节点”组成的抽象网络，网络中的各节点通过“边”实现彼此的连接。
- 节点：表示事物、对象或随机变量。
- 边：表示随机变量间的关系。
- 有向图与无向图：边是否有方向。
- 树状图：不包含圈的图称为无圈图(acyclic graph)，连通的无圈图称为树(tree)



不确定知识表示和推理

- 背景
- 概率论与图论基础
- 概率图模型的表示
- 概率图模型的推理
- 概率图模型的学习

三类概率图模型

□ 有向图模型(Directed graphs)

■ 如, 贝叶斯网络、隐马尔科夫模型、卡尔曼滤波

$$p(\mathbf{x}) = \prod_{k=1}^K p(x_k | \text{pa}_k)$$

□ 无向图模型(Undirected graphs)

■ 如, 马尔科夫随机场、条件随机场

$$p(\mathbf{x}) = \frac{1}{Z} \prod_C \psi_C(\mathbf{x}_C)$$

□ 因子图模型(Factor graphs)

$$p(\mathbf{x}) = \prod_s f_s(\mathbf{x}_s)$$

贝叶斯网络定义

- 贝叶斯网络(Bayesian Network, 简称BN)是不确定知识表示与推理的一种有效方法, 它由一个有向无环图(Directed Acyclic Graph, DAG)和一系列条件概率表(衡量了上述关系的强度)组成。

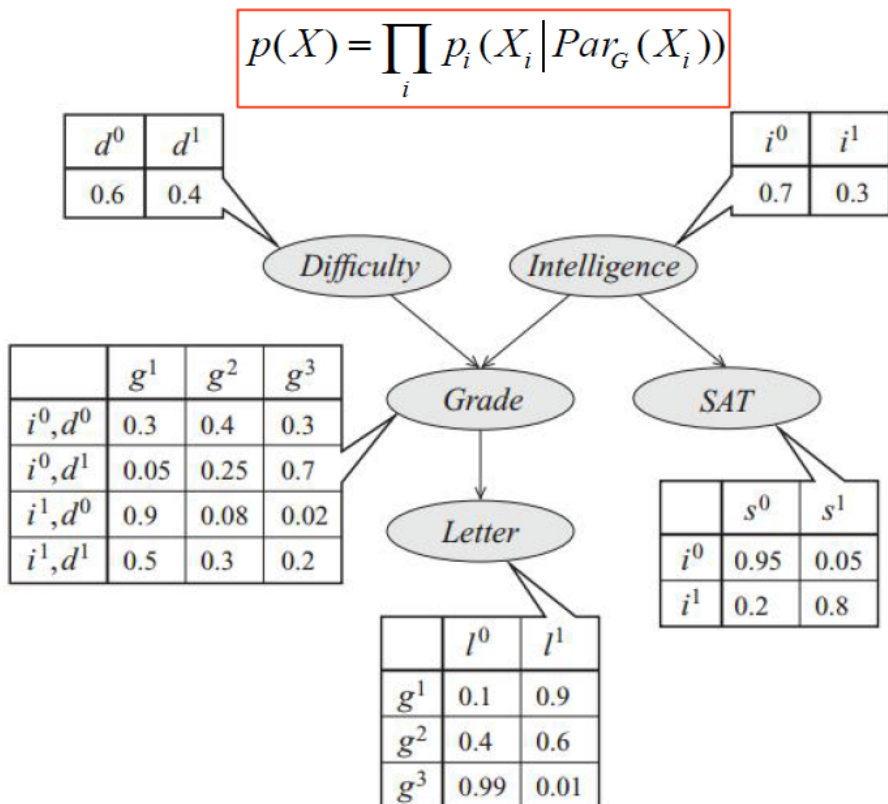
有向无环图指的是一个无回路的有向图, 即从图中任意一个节点出发经过任意条边, 均无法回到该节点。有向无环图刻画了图中所有节点之间的依赖关系。

- 1986年, Judea Pearl提出贝叶斯网络, 用于描述不确定知识表示与推理问题。贝叶斯网络让机器可以回答问题——给出一个从非洲回来的发烧且身体疼痛的病人, 最有可能的解释是疟疾。
- 令 G 为定义在 $\{X_1, X_2, \dots, X_N\}$ 上的一个贝叶斯网络, 其联合概率分布可以表示为各个节点的条件概率分布的乘积:

$$p(X) = \prod_i p_i(X_i | Par_G(X_i))$$

贝叶斯网络定义

□ 示例:

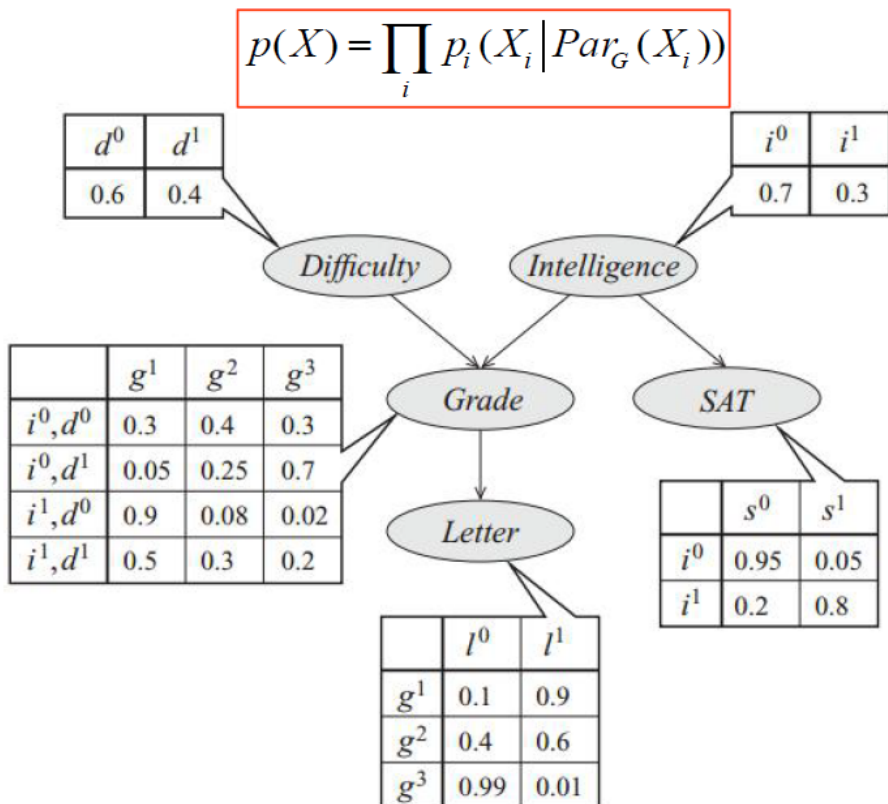


节点定义

- 试题难度 (D) : d^0 (低) , d^1 (高)
- 智力 (I) : i^0 (低) , i^1 (高)
- 考试成绩 (G) : g^1 (A) , g^2 (B) , g^3 (C)
- 高考成绩 (S) : s^0 (低) , s^1 (高)
- 是否得到推荐信 (L) : l^0 (否) , l^1 (是)

贝叶斯网络定义

□ 示例:



联合概率分布:

$$p(D, I, G, S, L)$$

$$= P(D)P(I)P(G|I, D)P(S|I)P(L|G)$$

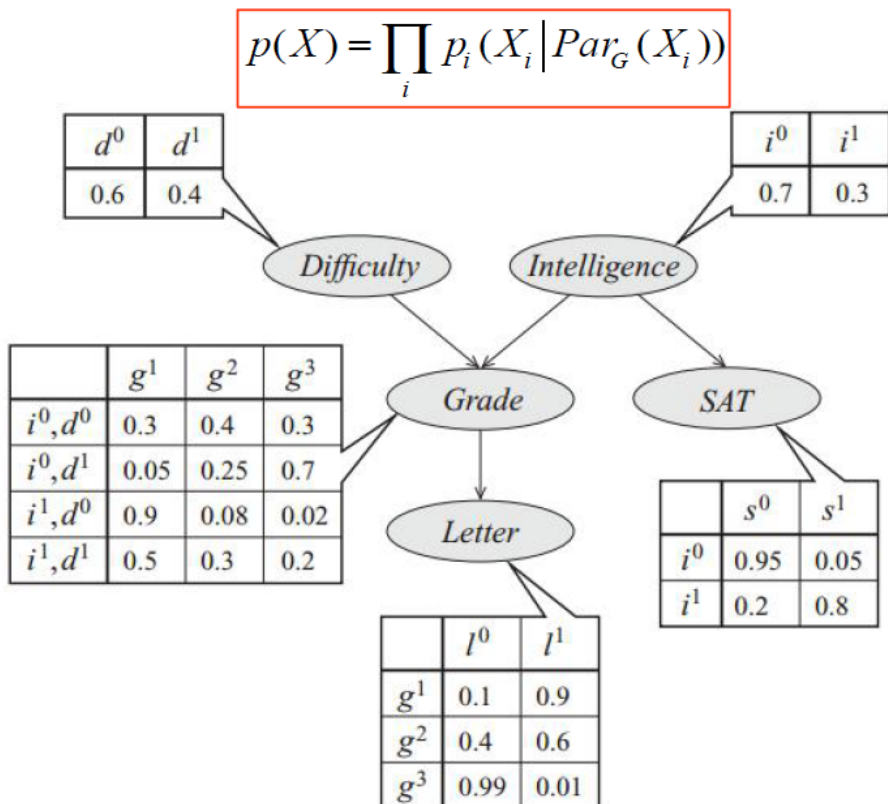
联合概率分布结构化分解的好处:

□ 枚举法: $2*2*3*2*2-1=47$ 个参数

□ 结构化分解: $1+1+8+3+2=15$ 个参数

贝叶斯网络定义

□ 示例:



联合概率分布:

$$p(D, I, G, S, L)$$

$$= P(D)P(I)P(G|I, D)P(S|I)P(L|G)$$

联合概率分布结构化分解的好处:

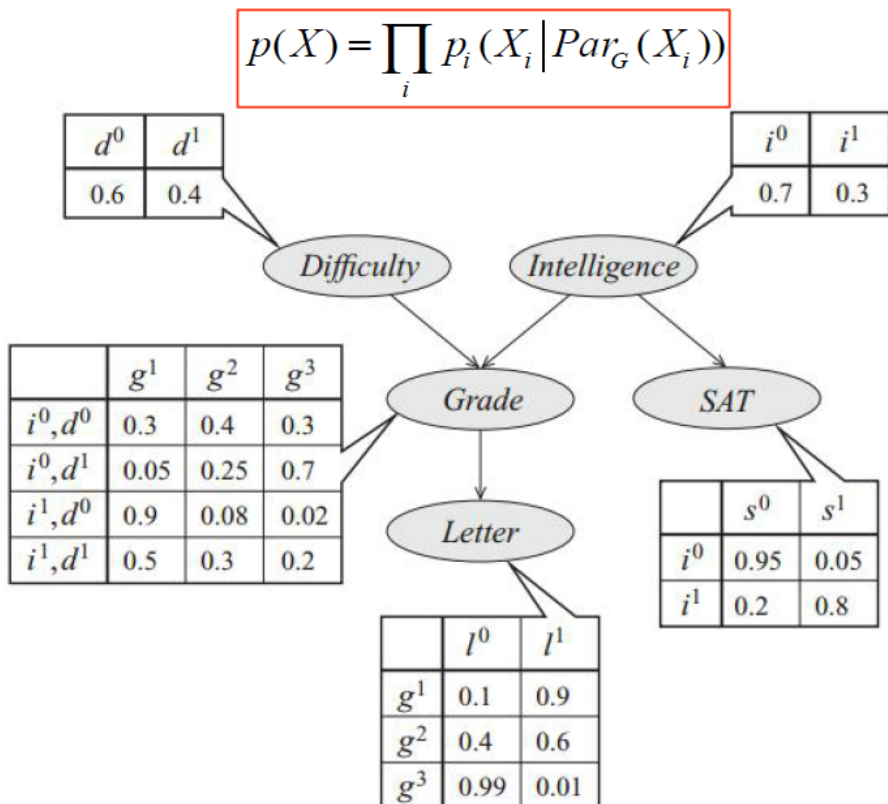
□ 枚举法: $2*2*3*2*2-1=47$ 个参数

□ 结构化分解: $1+1+8+3+2=15$ 个参数

□ 更一般地, 假设 n 个二元随机变量的联合概率分布, 表示该分布需要 2^n-1 个参数。如果用贝叶斯网络建模, 假设每个节点最多有 k 个父节点, 所需要的参数最多为 $n*2^k$, 一般每个变量局部依赖于少数变量。

贝叶斯网络定义

□ 示例:



联合概率分布:

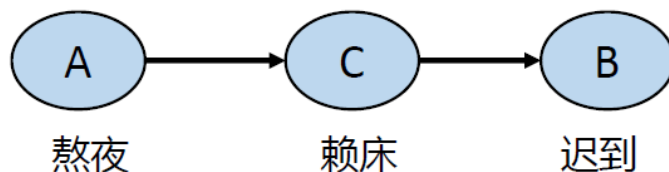
$$\begin{aligned} p(D, I, G, S, L) \\ = P(D)P(I)P(G|I, D)P(S|I)P(L|G) \end{aligned}$$

$$\begin{aligned} p(d^1, i^0, g^1, s^1, l^1) \\ = P(d^1)P(i^0)P(g^1|i^0, d^1)P(s^1|i^0)P(l^1|g^1) \\ = 0.4 \times 0.7 \times 0.05 \times 0.05 \times 0.1 \\ = 0.00007 \end{aligned}$$

$$\begin{aligned} p(d^0, i^1, g^1, s^1, l^1) \\ = P(d^0)P(i^1)P(g^1|i^1, d^0)P(s^1|i^1)P(l^1|g^1) \\ = 0.6 \times 0.3 \times 0.9 \times 0.8 \times 0.9 \\ = 0.11664 \end{aligned}$$

贝叶斯网络定义

□ 联合概率为什么可以表示为局部条件概率的乘积？



联合概率分布链式法则：

$$p(A, C, B) \\ = P(A)P(C|A)P(B|A, C)$$

条件独立：

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

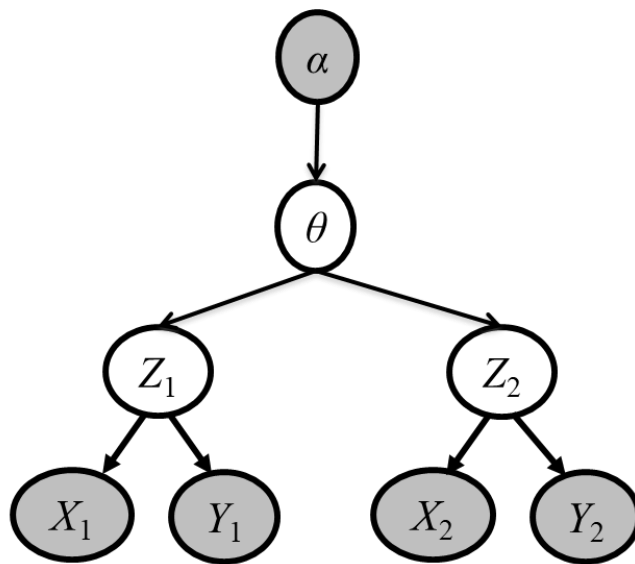
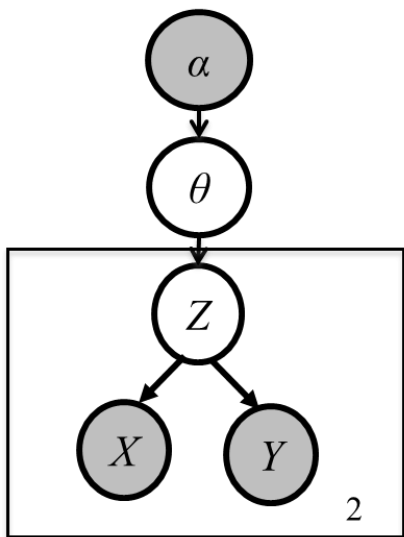


$$p(A, C, B) = P(A)P(C|A)P(B|C)$$

贝叶斯网络的表示

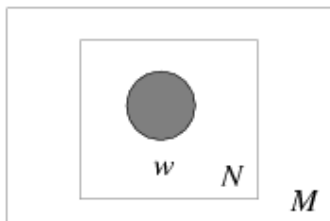
□ 盘式记法 (plate notation)

- 贝叶斯网络的一种简洁的表示方法，其将相互独立的、由相同机制生成的多个变量放在一个方框(盘)内，并在方框中标出类似变量重复出现的个数。此外，方框可以嵌套，且通常用阴影标注出可观察到的变量。
- 示例：



贝叶斯网络的表示

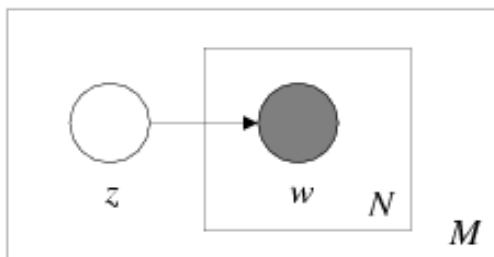
□ 一元语言模型:



$$p(\mathbf{w}) = \prod_{n=1}^N p(w_n)$$

假设文本中每个词都和其他词独立，和它的上下文无关。通过一元语言模型，我们可以计算一个序列的概率，从而判断该序列是否符合自然语言的语法和语义规则。

□ 一元混合语言模型:

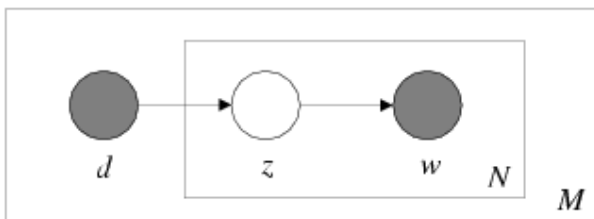


$$p(\mathbf{w}) = \sum_z p(z) \prod_{n=1}^N p(w_n | z)$$

假设给定文本主题 (z) 的前提下，该文本中的每个词都和其他词条件独立。其中， z 为隐变量。

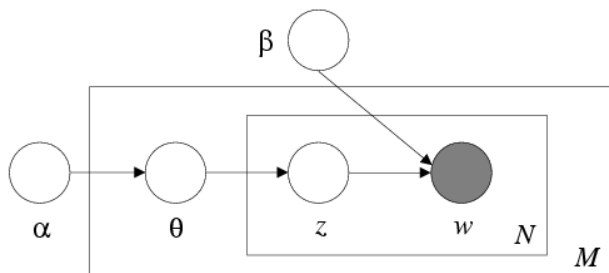
贝叶斯网络的表示

□ 概率潜在语义分析:



$$p(d, w_n) = p(d) \sum_z p(w_n | z) p(z | d)$$

□ 潜在狄利克雷分配:



$$p(\mathbf{w} | \alpha, \beta) = \int p(\theta | \alpha) \left(\prod_{n=1}^N \sum_{z_n} p(z_n | \theta) p(w_n | z_n, \beta) \right) d^k \theta$$

$$p(D | \alpha, \beta) = \prod_{d=1}^M \int p(\theta_d | \alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn} | \theta_d) p(w_{dn} | z_{dn}, \beta) \right) d^k \theta_d$$

贝叶斯网络的构建

□ 给定变量 $\{X_1, \dots, X_n\}$, 构建一个贝叶斯网络的步骤如下:

■ **步骤1:** 在某种变量顺序下, 对所有变量的联合概率应用链式法则

$$Pr(X_1, \dots, X_n) = Pr(X_n | X_1, \dots, X_{n-1}) Pr(X_{n-1} | X_1, \dots, X_{n-2}) \dots Pr(X_1)$$

■ **步骤2:** 对于每个变量 X_i , 考虑该变量的条件集合 X_1, \dots, X_{i-1} , 采用如下方法递归地判断条件集合中的每个变量 X_j 是否可以删除: 如果给定其余变量的集合, X_i 和 X_j 是条件独立的, 则将 X_j 从 X_i 的条件集合中删除。经过这一步骤, 可以得到下式

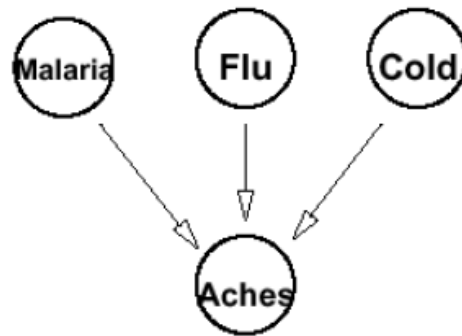
$$Pr(X_1, \dots, X_n) = Pr(X_n | Par(X_n)) Pr(X_{n-1} | Par(X_{n-1})) \dots Pr(X_1)$$

■ **步骤3:** 基于上述公式, 构建一个有向无环图。其中, 对于每个用节点表示的变量 X_i , 其父节点为 $Par(X_i)$ 中的变量集合。

■ **步骤4:** 为每个家庭(即变量及其父节点集合)确定条件概率表的取值。

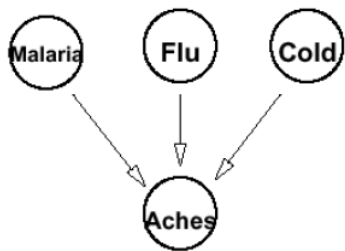
贝叶斯网络的构建

- 对于疟疾(M)、流感(F)、感冒(C)和疼痛(A)这四个变量, 假设我们知道疟疾(M)、流感(F)和感冒(C)都会导致疼痛(A), 那么在构建贝叶斯网络时, 我们可以按照下述顺序完成步骤1: $Pr(M, F, C, A) = Pr(A|M, F, C)Pr(C|M, F)Pr(F|M)Pr(M)$
- 由于M、F和C都会导致A, 所以变量A的条件集合不能删减。此外, 我们通常认为这三种疾病(即M、F和C)的发生是独立的, 因此, $Pr(C|M, F) = Pr(C)$, 且 $Pr(F|M) = Pr(F)$ 。据此构建的贝叶斯网络的有向无环图如下所示:



贝叶斯网络的构建

□ $Pr(M, F, C, A) = Pr(A|M, F, C)Pr(C|M, F)Pr(F|M)Pr(M)$



M
True

F
True

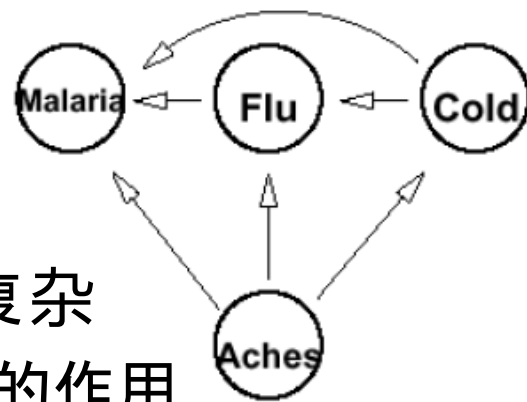
C
True

M	F	C
True	True	True
True	True	False
True	False	True
True	False	False
False	True	True
False	True	False
False	False	True
False	False	False

□ 该贝叶斯网络的参数个数为: $2^3 + 1 + 1 + 1 = 11$

贝叶斯网络的构建

- 对于上述例子，如果我们按照另一种顺序完成步骤1：
 $Pr(M, F, C, A) = Pr(M|F, C, A)Pr(F|C, A)Pr(C|A)Pr(A)$
- 由于每个变量的条件集合都不能删减，据此构建的贝叶斯网络的有向无环图为：
- 该贝叶斯网络的参数个数为：
 $2^3 + 2^2 + 2 + 1 = 15$
- 与前面构建的贝叶斯网络相比，更为复杂
- 由此可见因果关系等先验知识在其中的作用



贝叶斯网络的构建

- 在分类等特定的任务中，为了能够从有限的训练样本中进行标签预测，有时也会忽略因果关系，人为假定一些条件独立性。
- 基于“属性条件独立性假设”的朴素贝叶斯分类模型是通过这种方式构建的一种特殊结构的贝叶斯网络，它在强(朴素)独立性假设的条件下运用贝叶斯定理来计算每个类别的条件概率。
- 虽然朴素贝叶斯分类模型的条件独立性假设太强，但在实际应用中，该模型在很多任务上也能得到很好的结果，并且模型简单，可以有效防止过拟合。

贝叶斯网络的构建

年龄 (A)	收入 (I)	学生 (S)?	信用等级 (C)?	是否买电脑 (B)?
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no
<=30	medium	yes	fair	?

后验概率 $Pr(B = \text{yes} \mid A \leq 30, I = \text{medium}, S = \text{yes}, C = \text{fair}) = ?$

贝叶斯网络的构建

年龄 (A)	收入 (I)	学生 (S)?	信用等级 (C)?	是否买电脑 (B)?
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no
<=30	medium	yes	fair	?

后验概率 $Pr(B = \text{yes} \mid A \leq 30, I = \text{medium}, S = \text{yes}, C = \text{fair}) = ?$

先验概率 $Pr(B = \text{yes}) = 9/14 \approx 0.64$

似然度 $Pr(A \leq 30, I = \text{medium}, S = \text{yes}, C = \text{fair} \mid B = \text{yes}) = 0$

先验概率 $Pr(B = \text{no}) = 5/14 \approx 0.36$

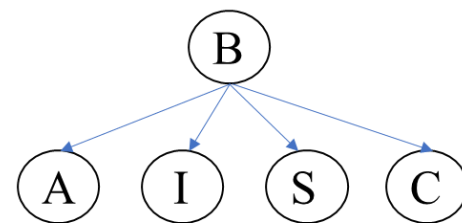
似然度 $Pr(A \leq 30, I = \text{medium}, S = \text{yes}, C = \text{fair} \mid B = \text{no}) = 0$

需要更大规模的训练数据!

贝叶斯网络的构建

年龄 (A)	收入 (I)	学生 (S)?	信用等级 (C)?	是否买电脑 (B)?
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no
<=30	medium	yes	fair	?

假设 $Pr(A, I, S, C | B) = Pr(A|B)Pr(I|B)Pr(S|B)Pr(C|B)$,
即给定 B 的条件下, A 、 I 、 S 、 C 相互独立 \Rightarrow 朴素贝叶斯分类



该贝叶斯网络的参数个数为:
 $1 + 2*2 + 2*2 + 2 + 2 = 13$ 个

无上述假设的网络参数个数:
 $3*3*2*2*2 - 1 = 71$ 个

贝叶斯网络的构建

年龄 (A)	收入 (I)	学生 (S)?	信用等级 (C)?	是否买电脑 (B)?
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no
<=30	medium	yes	fair	?

后验概率 $Pr(B = \text{yes} \mid A \leq 30, I = \text{medium}, S = \text{yes}, C = \text{fair}) = ?$

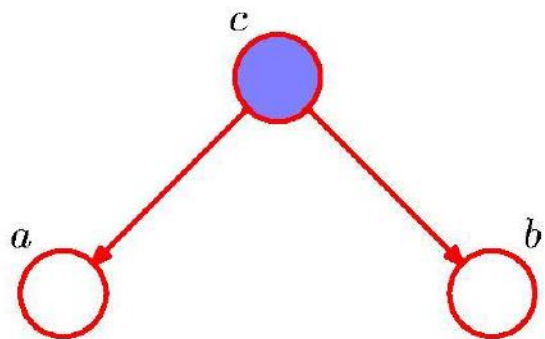
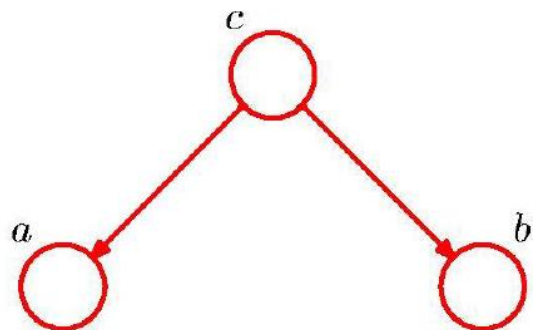
先验概率 $Pr(B = \text{yes}) = 9/14 \approx 0.64$

似然度 $Pr(A \leq 30, I = \text{medium}, S = \text{yes}, C = \text{fair} \mid B = \text{yes}) =$
 $Pr(A \leq 30 \mid B = \text{yes}) *$
 $Pr(I = \text{medium} \mid B = \text{yes}) *$
 $Pr(S = \text{yes} \mid B = \text{yes}) *$
 $Pr(C = \text{fair} \mid B = \text{yes}) =$
 $(2/9) * (4/9) * (6/9) * (6/9)$

归一化后, $Pr(B = \text{yes} \mid A \leq 30, I = \text{medium}, S = \text{yes}, C = \text{fair}) =$
0.8 买电脑的置信度上升!

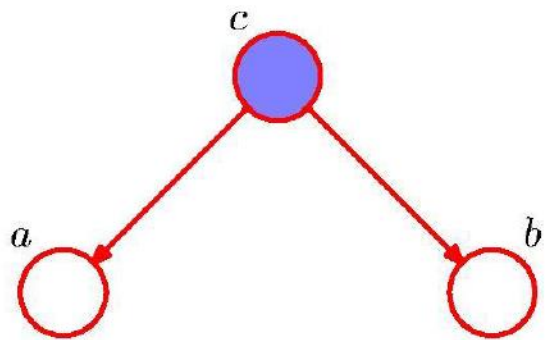
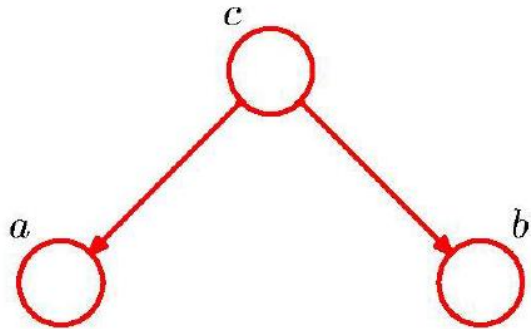
条件独立

□ Fork (tail-to-tail)



条件独立

□ Fork (tail-to-tail)



$$p(a, b, c) = p(c) p(a|c) p(b|c)$$

$$a \perp\!\!\!\perp b \mid \emptyset ?$$

$$p(a, b \mid c) = \frac{p(a, b, c)}{p(c)}$$

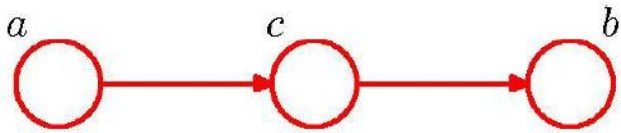
$$= \frac{p(c) p(a|c) p(b|c)}{p(c)}$$

$$= p(a|c) p(b|c)$$

$$\boxed{a \perp\!\!\!\perp b \mid c}$$

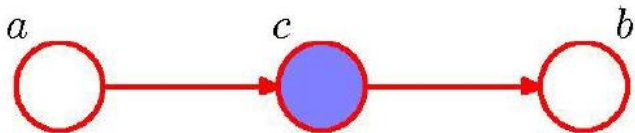
条件独立

□ Chain (head-to-tail)



$$p(a, b, c) = p(a) p(c|a) p(b|c)$$

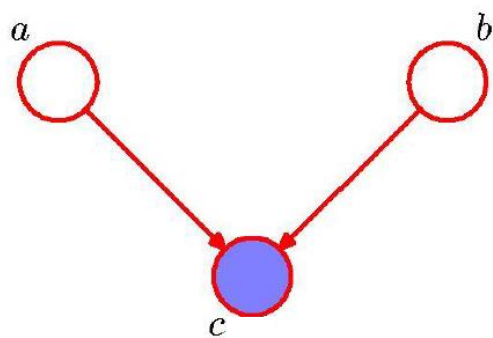
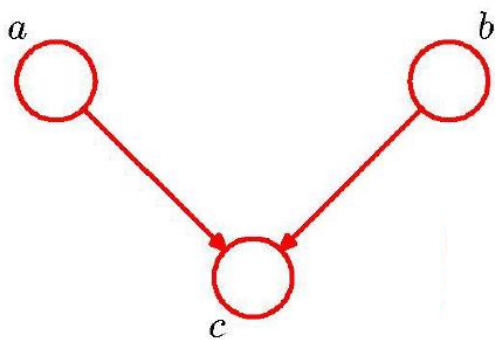
$$a \not\perp b \mid \emptyset$$



$$a \perp b \mid c$$

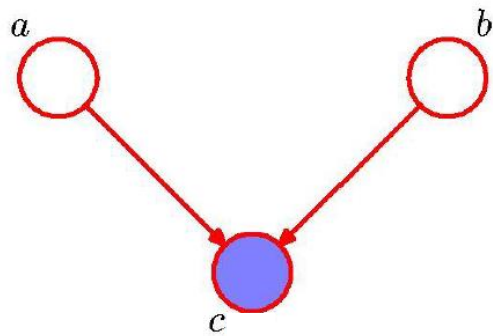
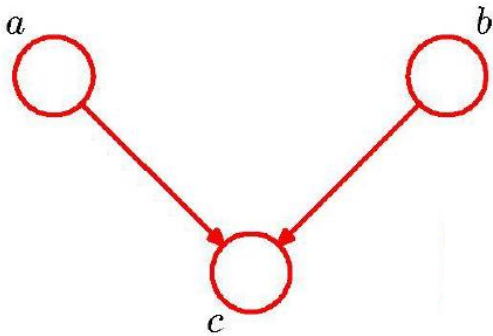
条件独立

❑ Collider (head-to-head)



条件独立

❑ Collider (head-to-head)



$$p(a, b, c) = p(a)p(b)p(c|a, b)$$

$$p(a, b) = \sum_c p(a, b, c)$$

$$= p(a)p(b) \sum_c p(c|a, b)$$

$$= p(a)p(b)$$

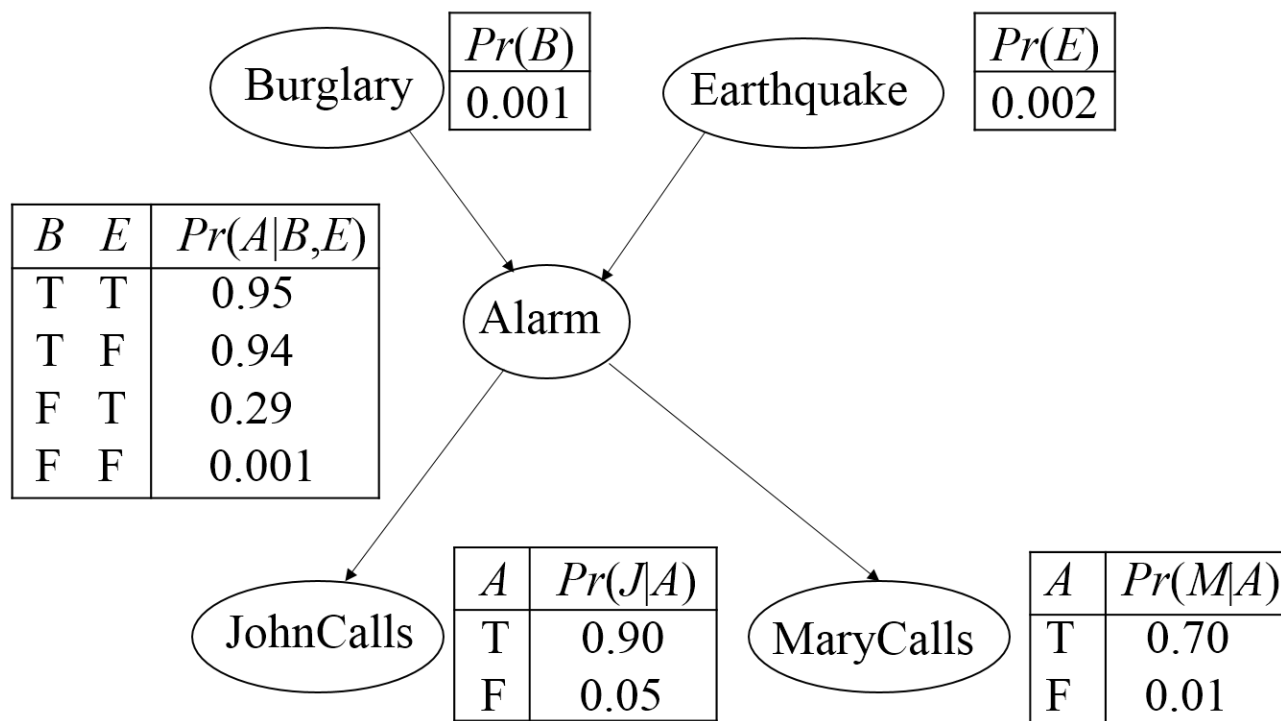
$$p(a, b|c) = \frac{p(a, b, c)}{p(c)} = \frac{p(a)p(b)p(c|a, b)}{p(c)} \neq p(a|c)p(b|c)$$

贝叶斯网络中的D-分离

- 对于一个有向无环图, **D-分离** (D-Separation) 是一种用来判断其变量是否条件独立的图形化方法。
- 在基于贝叶斯网络的不确定性知识推理中, 采用D-分离方法可以简化概率计算, 提高运行速度。
- 示例:
 - 小偷(Burglar)会引发警报(Alarm)
 - 地震(Earthquake)会引发警报(Alarm)
 - 警报(Alarm)会引起邻居约翰打电话(JohnCalls)
 - 警报(Alarm)会引起邻居玛丽打电话(MaryCalls)

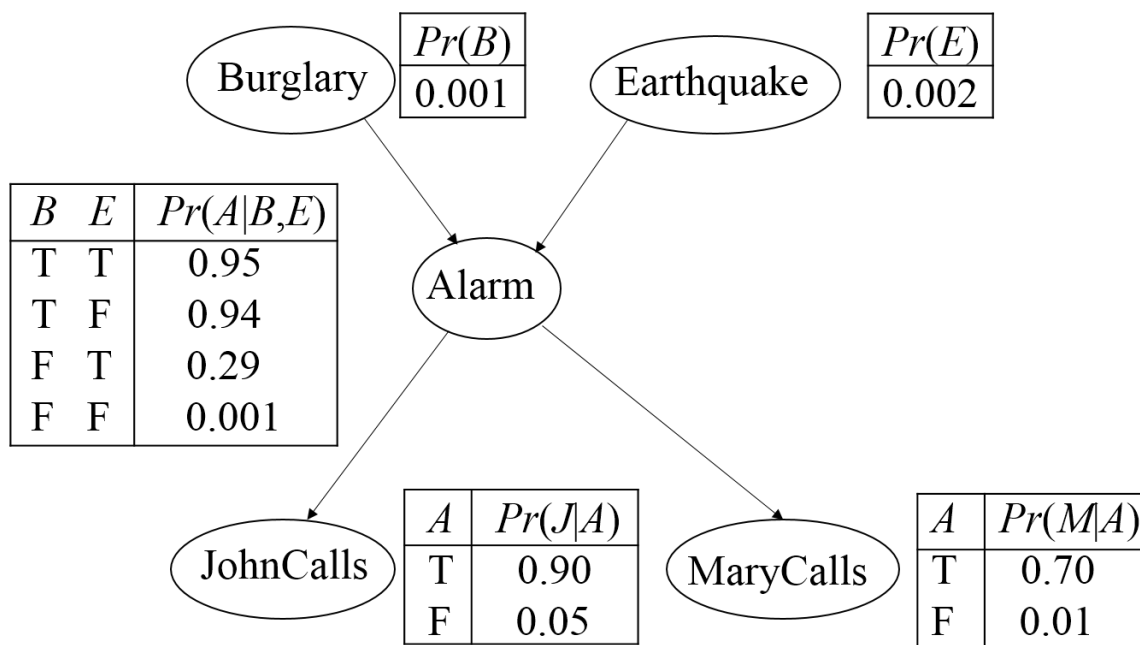
贝叶斯网络中的D-分离

□ 示例：



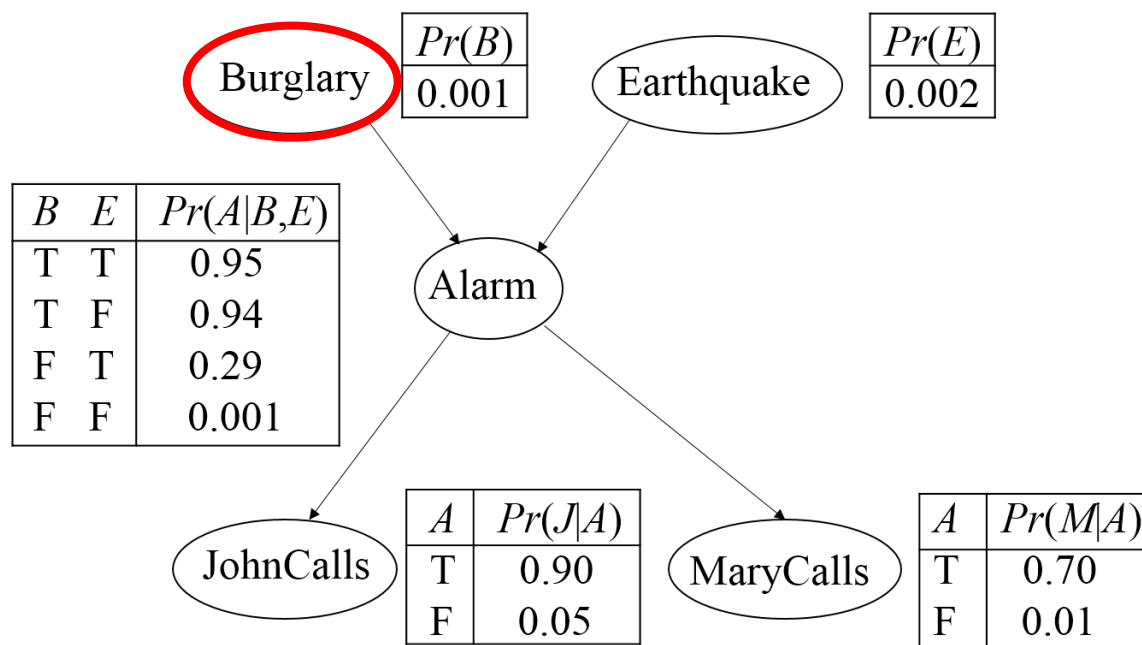
贝叶斯网络中的D-分离

□ 间接的因果作用：



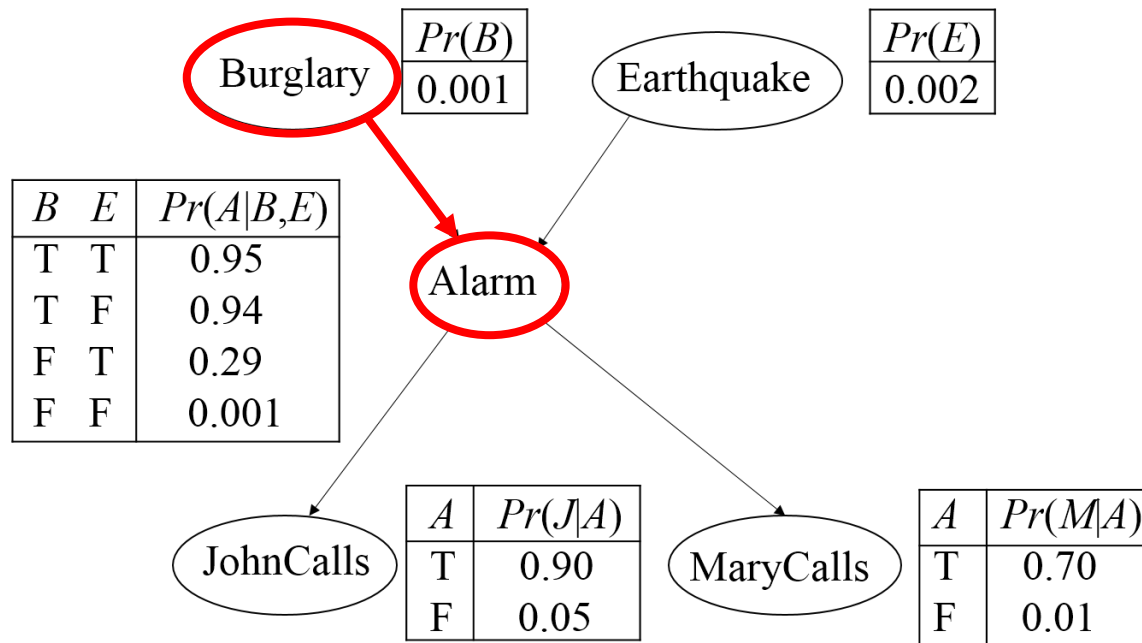
贝叶斯网络中的D-分离

□ 间接的因果作用：



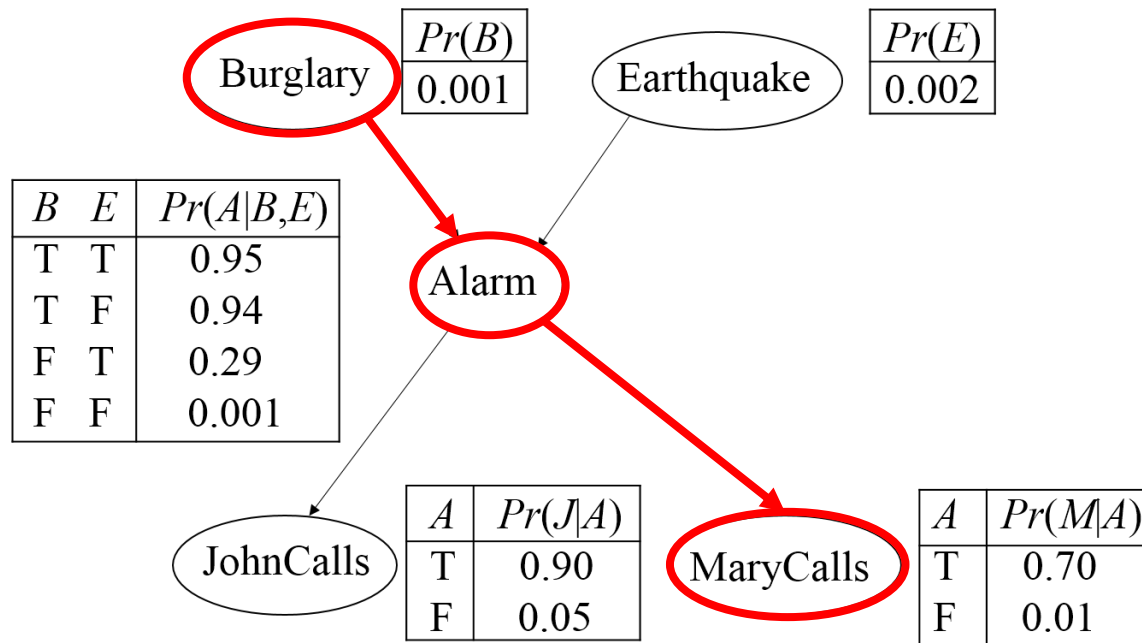
贝叶斯网络中的D-分离

□ 间接的因果作用：



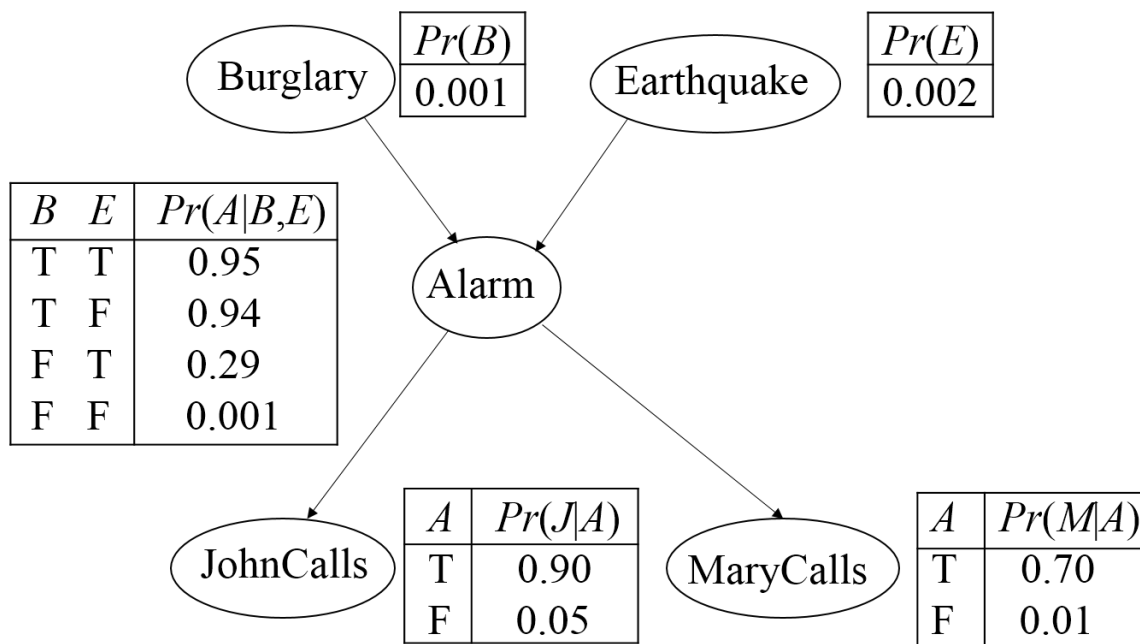
贝叶斯网络中的D-分离

□ 间接的因果作用：



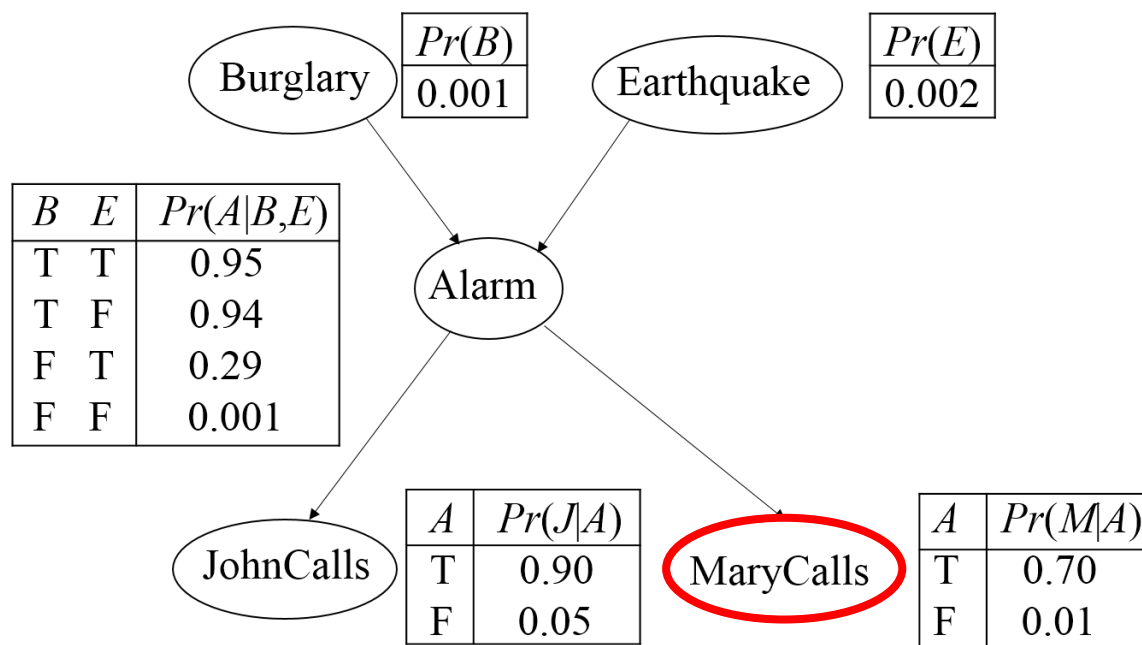
贝叶斯网络中的D-分离

□ 间接的证据作用：



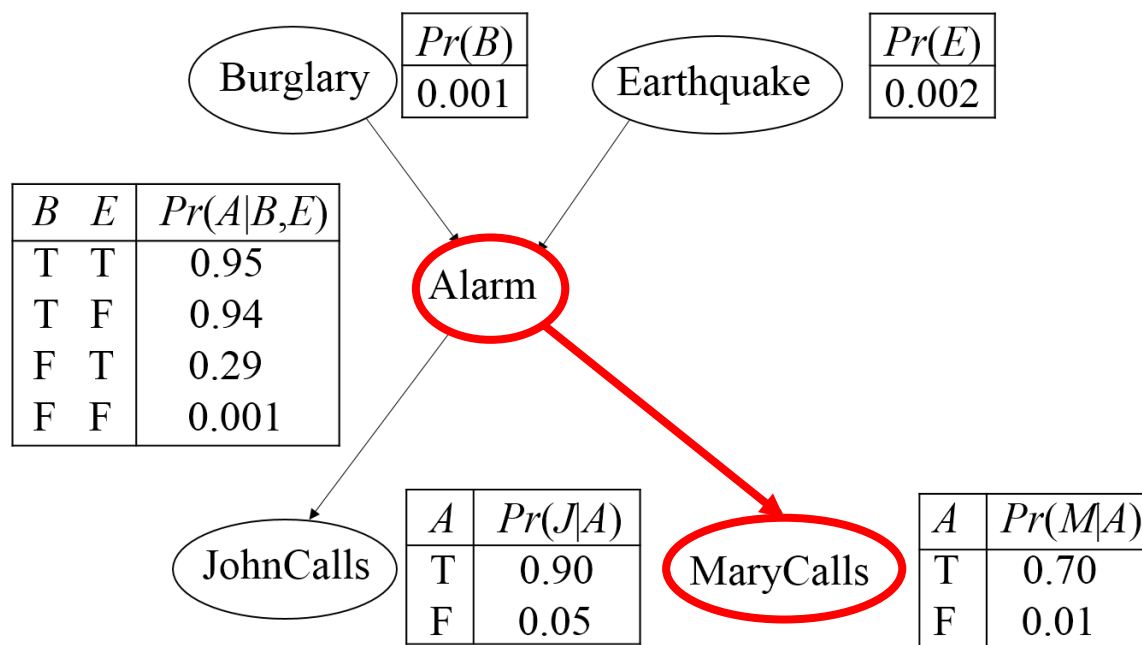
贝叶斯网络中的D-分离

□ 间接的证据作用：



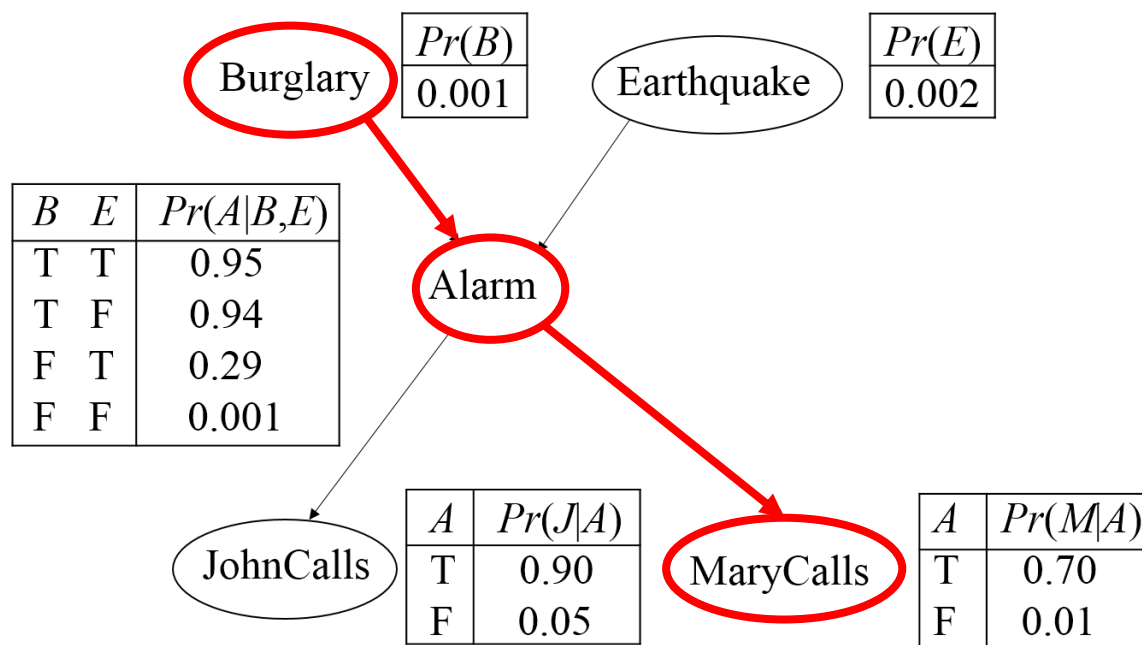
贝叶斯网络中的D-分离

□ 间接的证据作用：



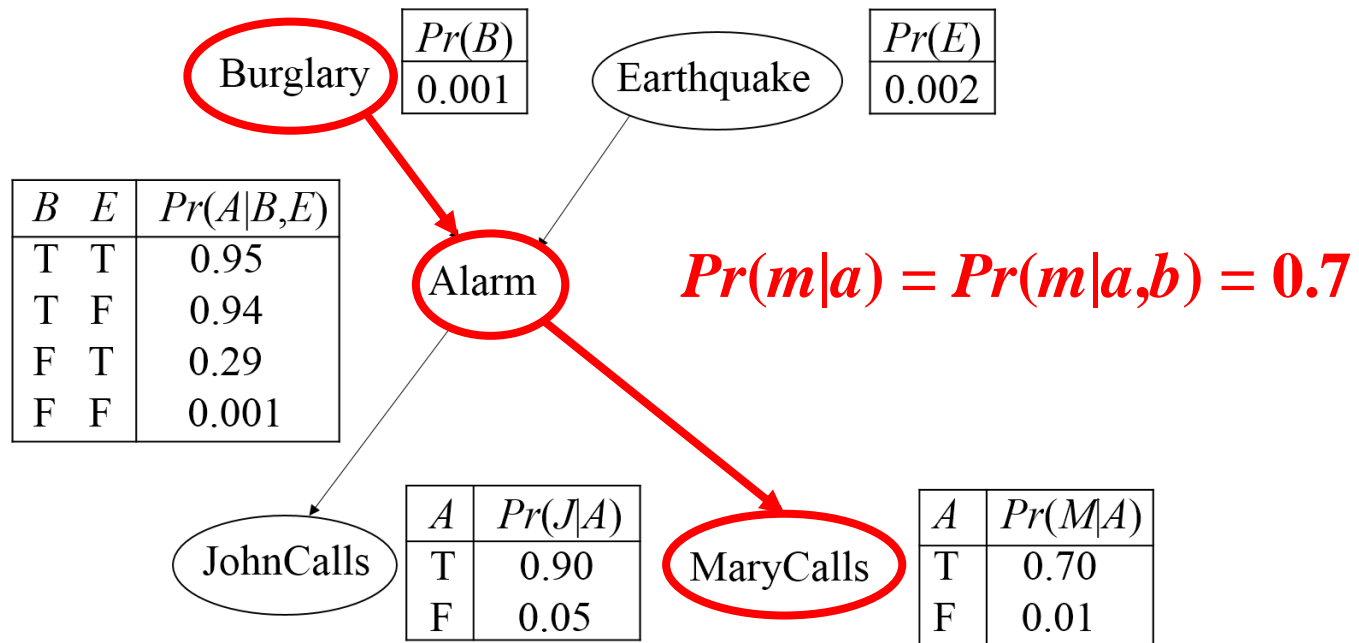
贝叶斯网络中的D-分离

□ 间接的证据作用：



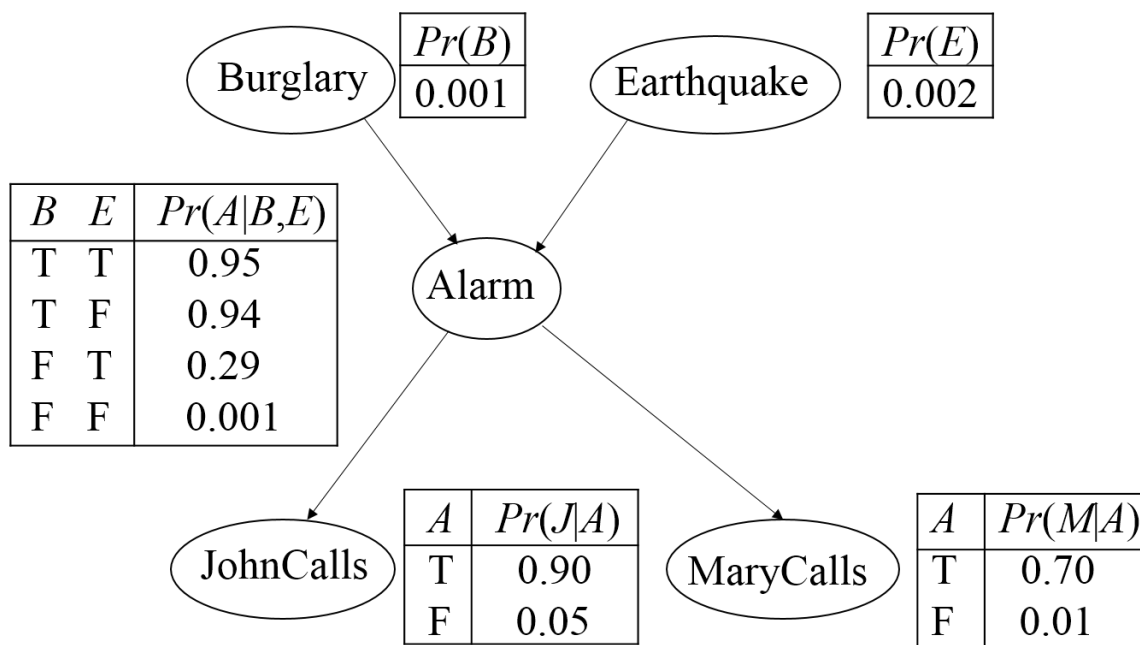
贝叶斯网络中的D-分离

- 间接的因果/证据作用 (Chain): 给定A的前提下, M与B条件独立。



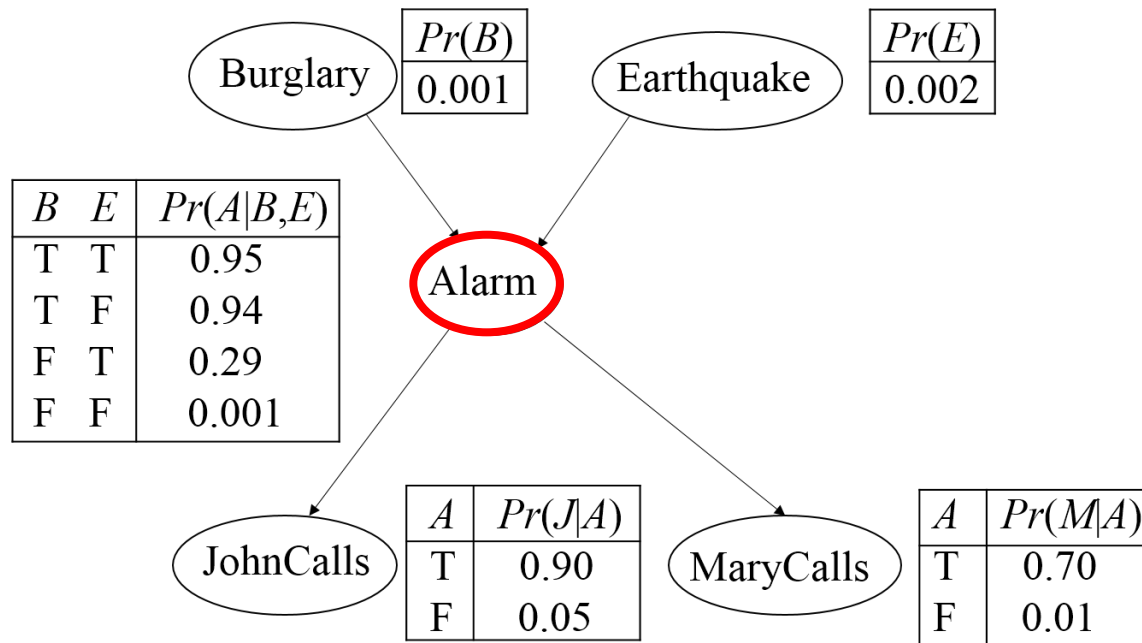
贝叶斯网络中的D-分离

□ 共同的原因：



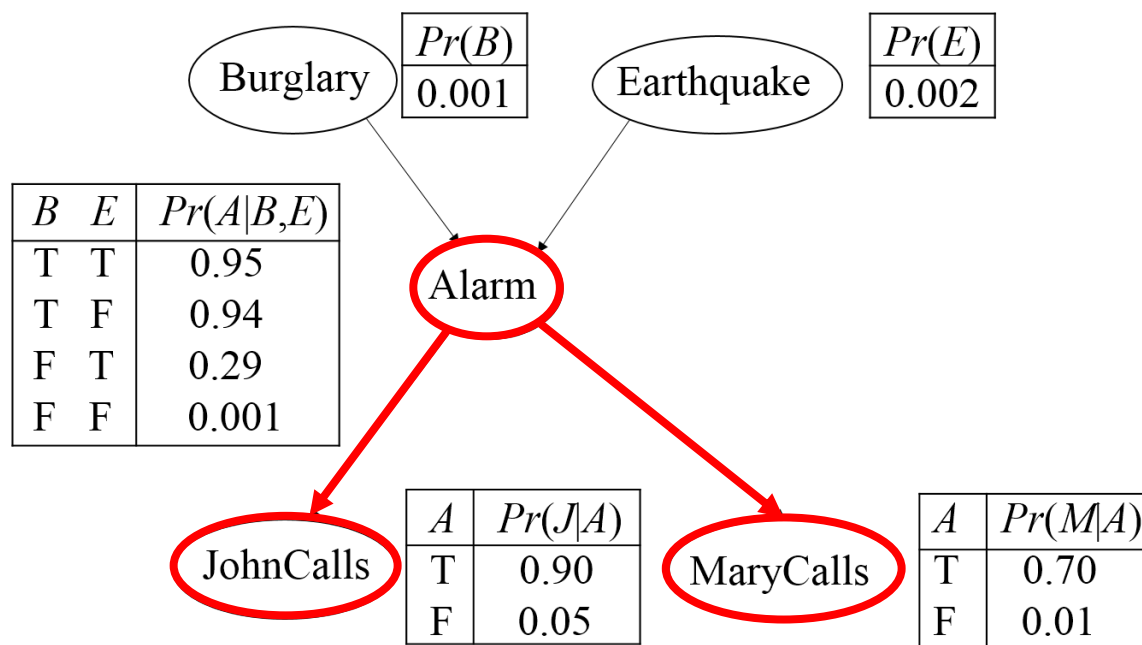
贝叶斯网络中的D-分离

□ 共同的原因：



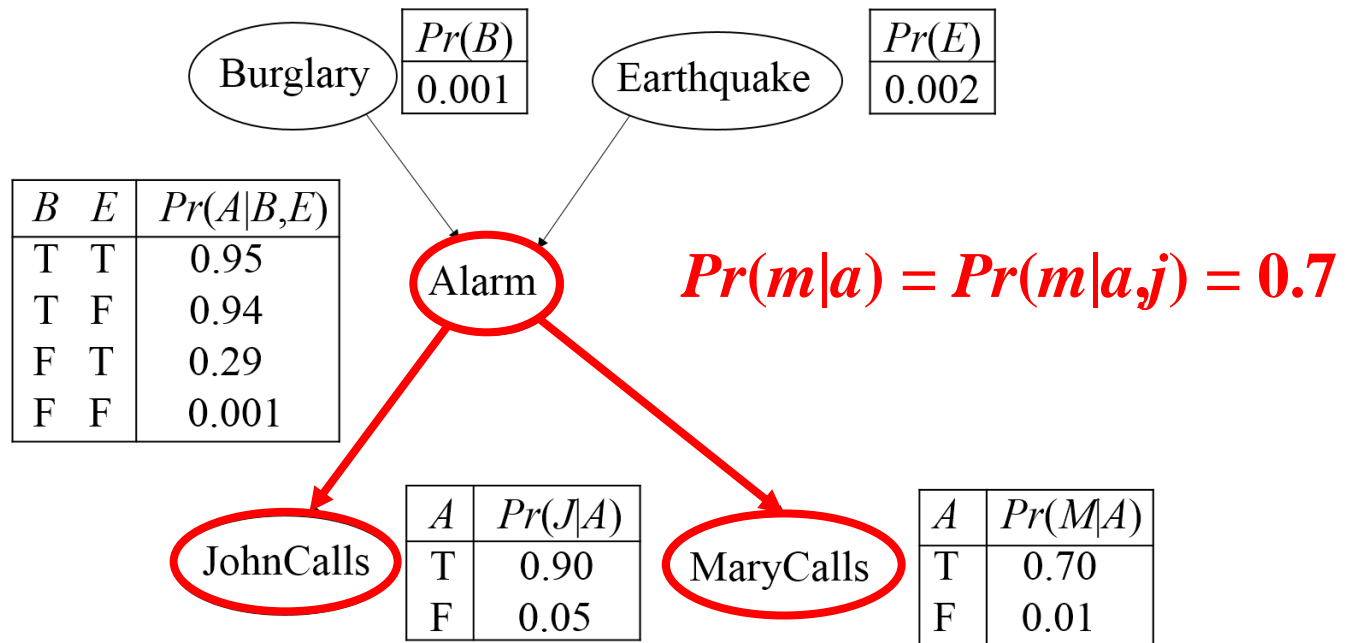
贝叶斯网络中的D-分离

□ 共同的原因：



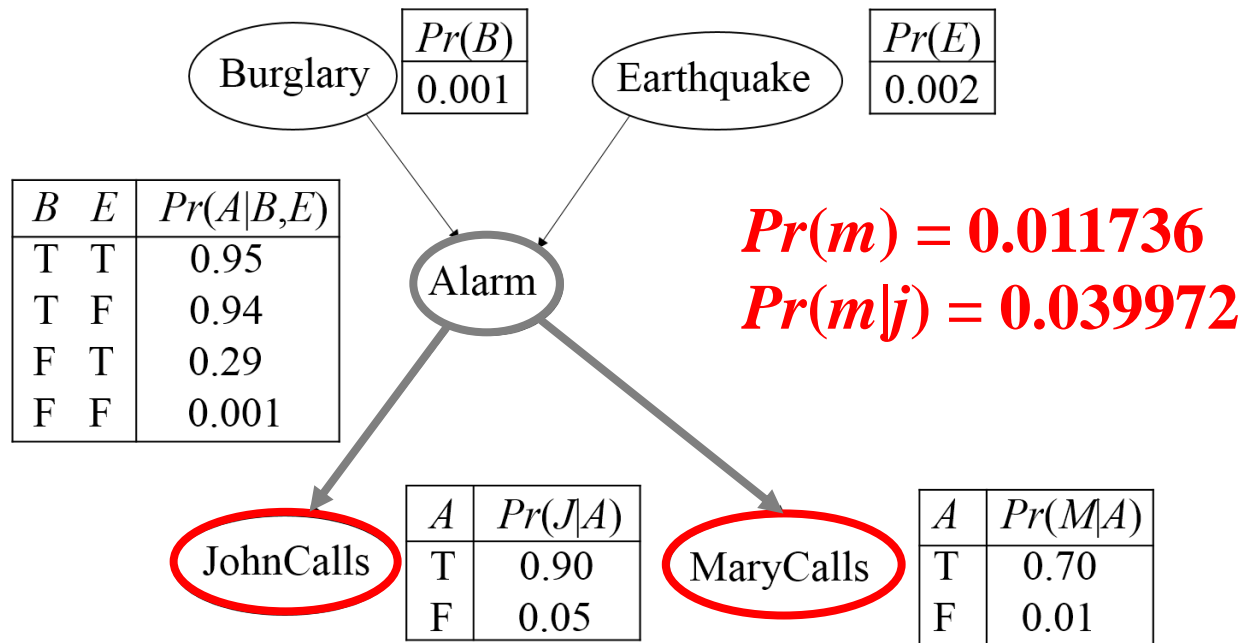
贝叶斯网络中的D-分离

- 共同的原因(Fork): 给定A的前提下, J与M条件独立。



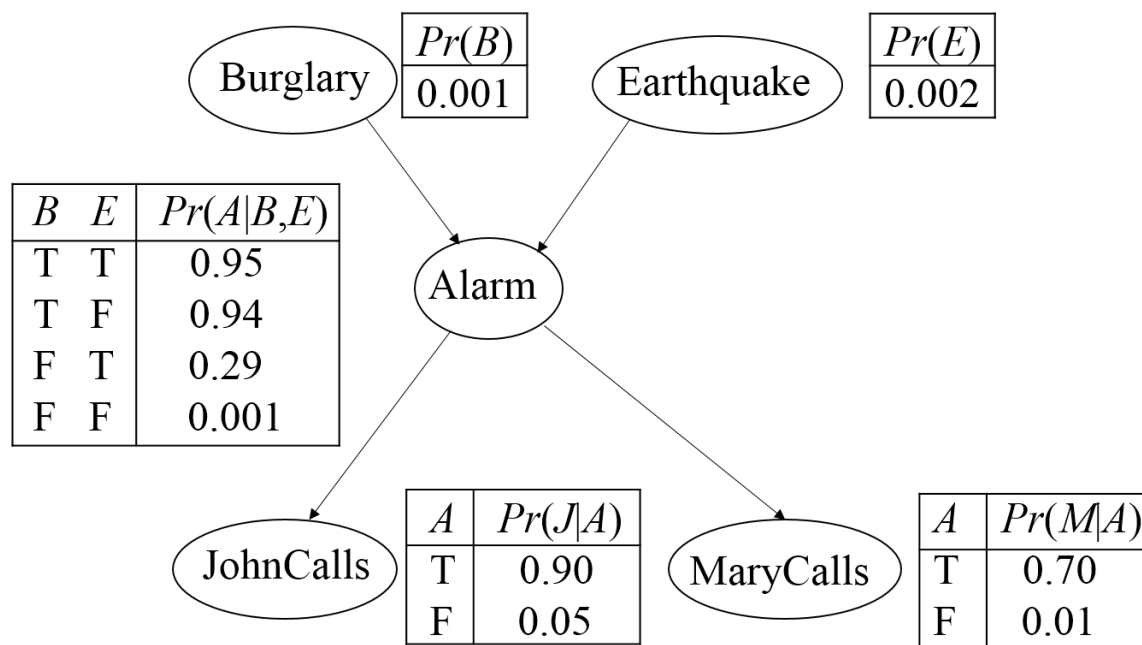
贝叶斯网络中的D-分离

- 共同的原因(Fork) : 未给定A的前提下, J与M不独立。



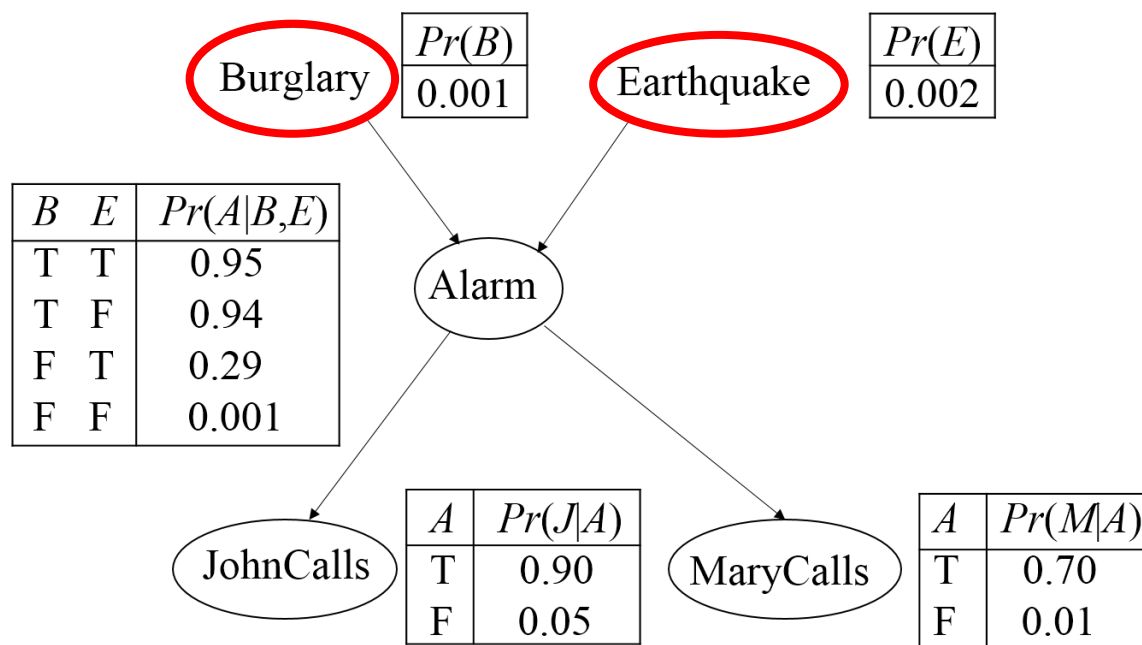
贝叶斯网络中的D-分离

□ 共同的作用：



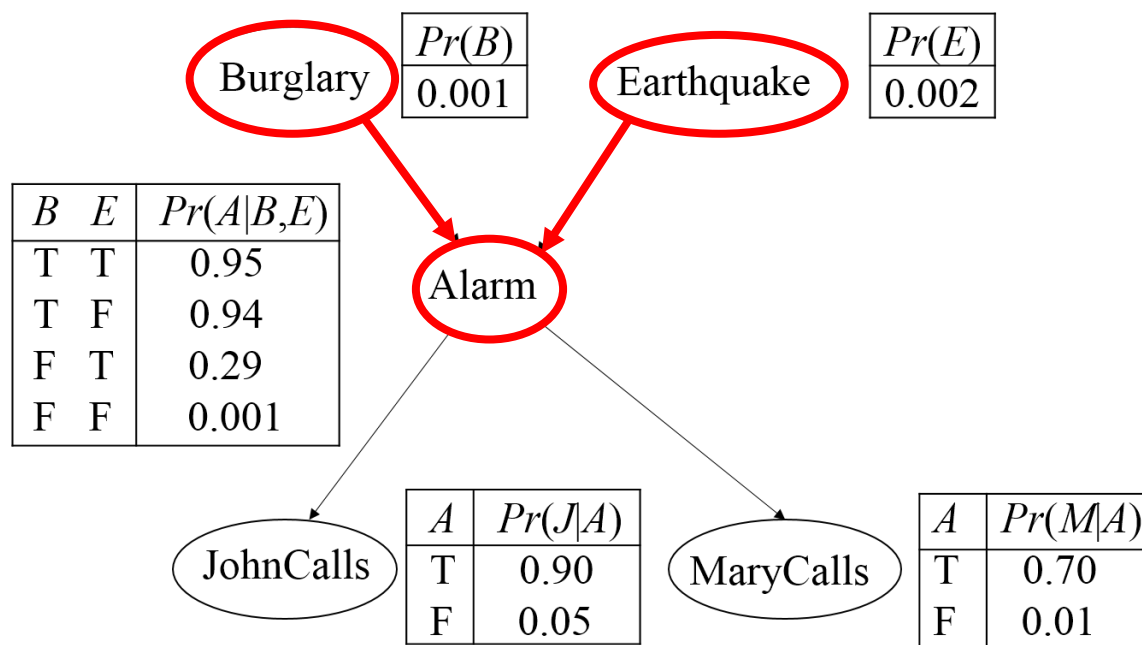
贝叶斯网络中的D-分离

□ 共同的作用：



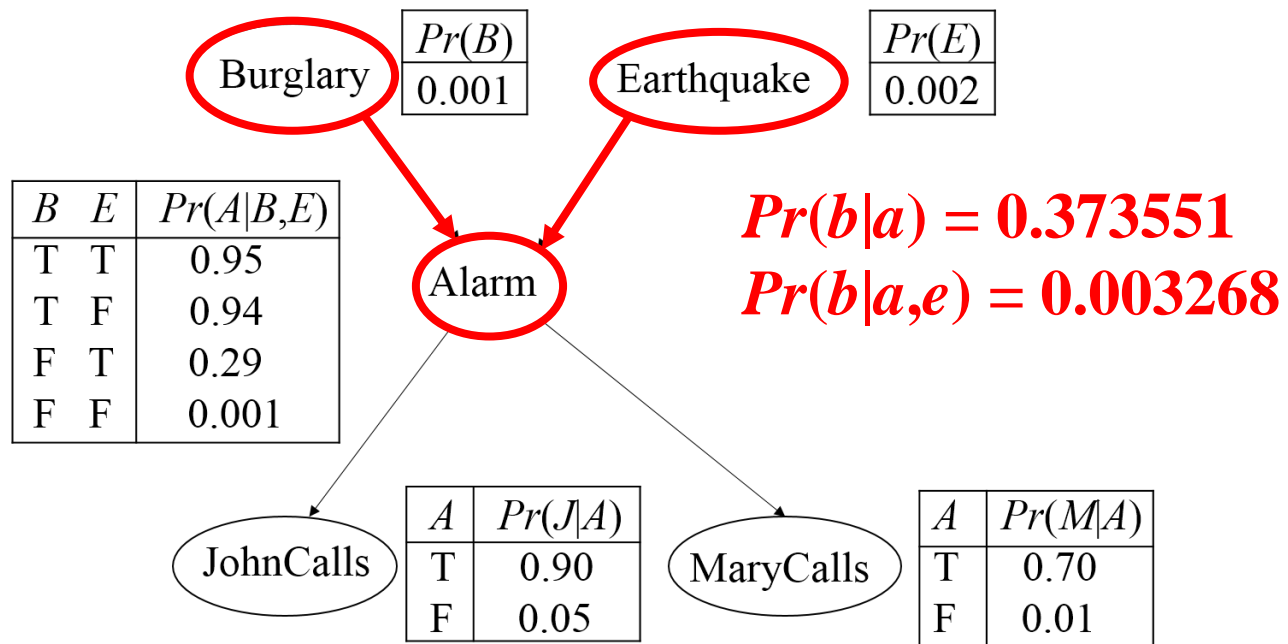
贝叶斯网络中的D-分离

□ 共同的作用：



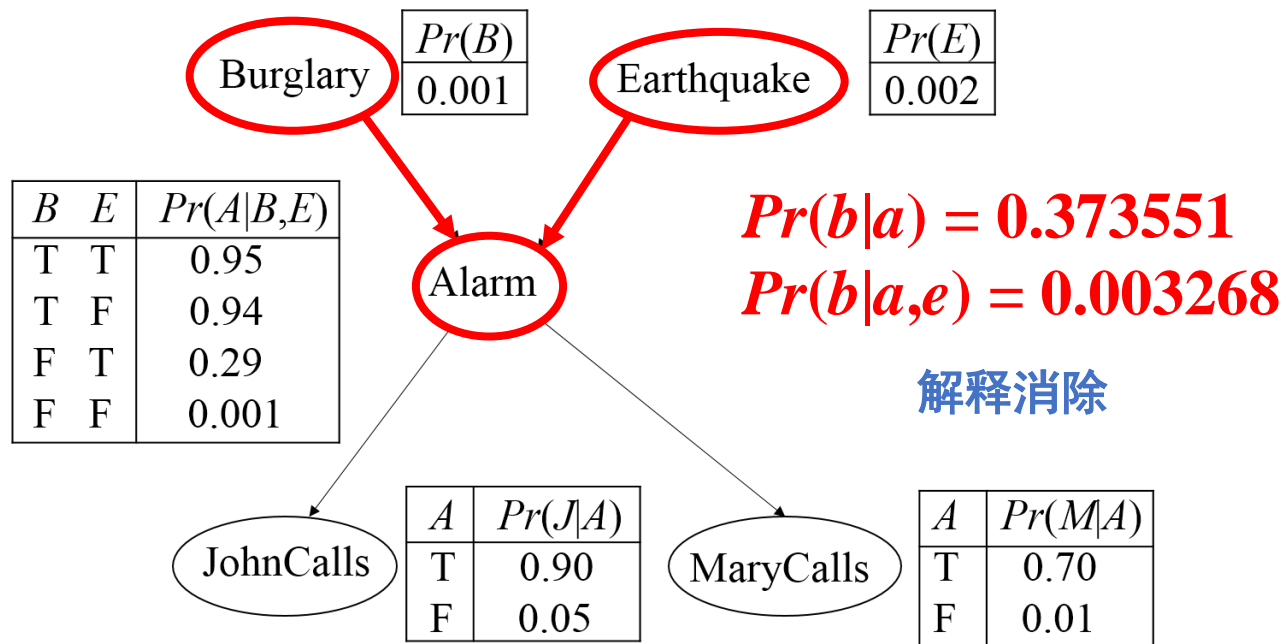
贝叶斯网络中的D-分离

- 共同的作用 (Collider): 给定A的前提下, B与E不是条件独立的。



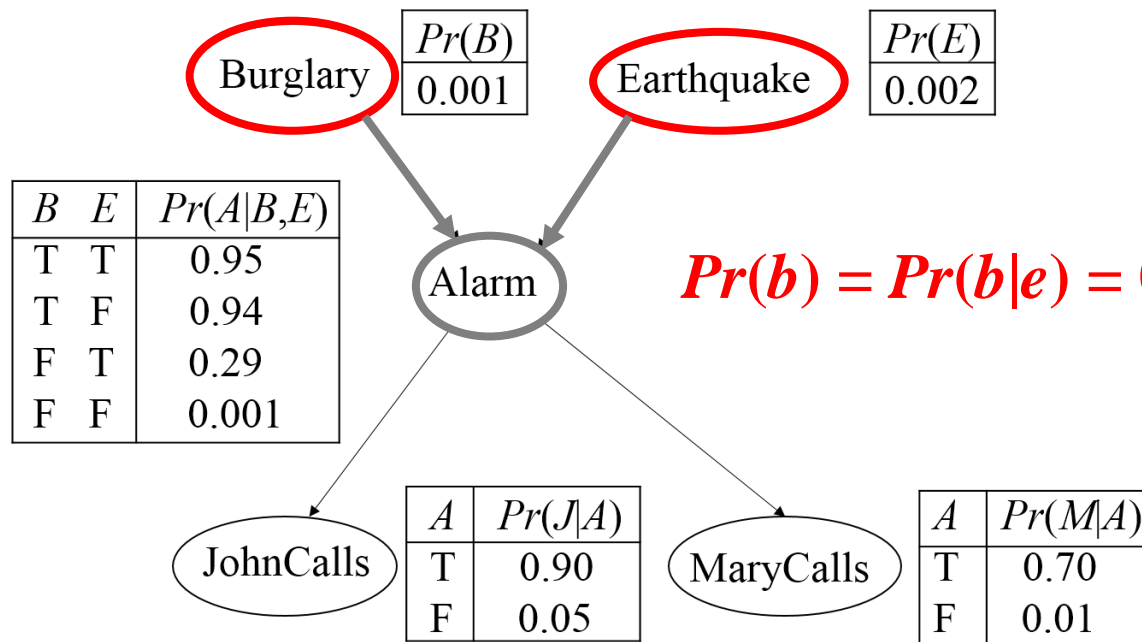
贝叶斯网络中的D-分离

- 共同的作用 (Collider): 给定A的前提下, B与E不是条件独立的。



贝叶斯网络中的D-分离

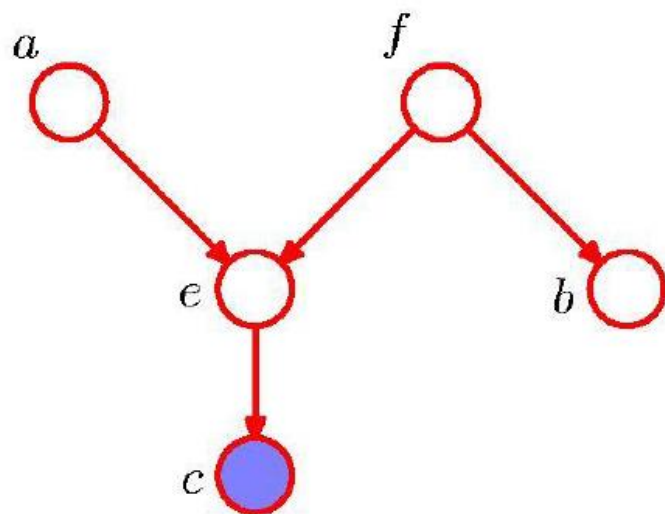
- 共同的作用 (Collider) : 未给定A的前提下, B与E是独立的。



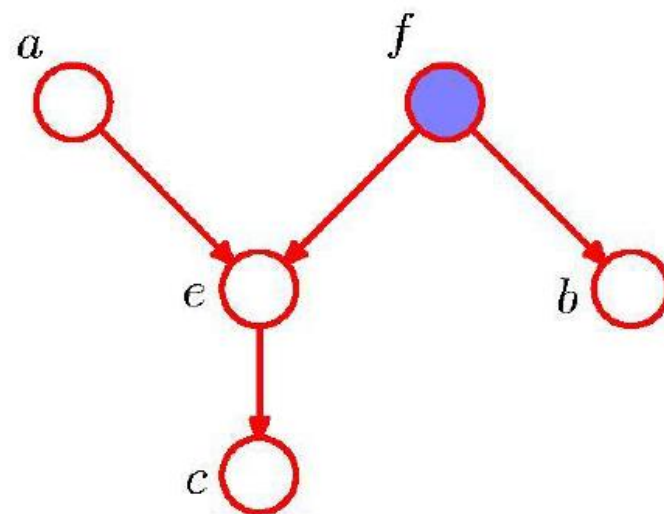
贝叶斯网络中的D-分离

- D-分离 (Directional separation, D-separation) 可用于判断任意两个节点的相关性和独立性。若存在一条路径将这两个节点(直接)连通, 则称这两个节点是有向连接(d-connected)的, 即这两个节点是相关的; 若不存在这样的路径将这两个节点连通, 则这两个节点不是有向连接的, 则称这两个节点是有向分离的(d-separated), 即这两个节点相互独立。
- 定义: 路径 p 被限定集 Z 阻塞(block)当且仅当:
 - 路径 p 含有链结构 $A \rightarrow B \rightarrow C$ 或分连结构 $A \leftarrow B \rightarrow C$ 且中间节点 B 在 Z 中, 或者
 - 路径 p 含有汇连结构 $A \rightarrow B \leftarrow C$ 且汇连节点 B 及其后代都不在 Z 中。
- 定义: 若 Z 阻塞了节点 X 和节点 Y 之间的每一条路径, 则称给定 Z 时, X 和 Y 是D-分离, 即给定 Z 时, X 和 Y 条件独立。

贝叶斯网络中的D-分离

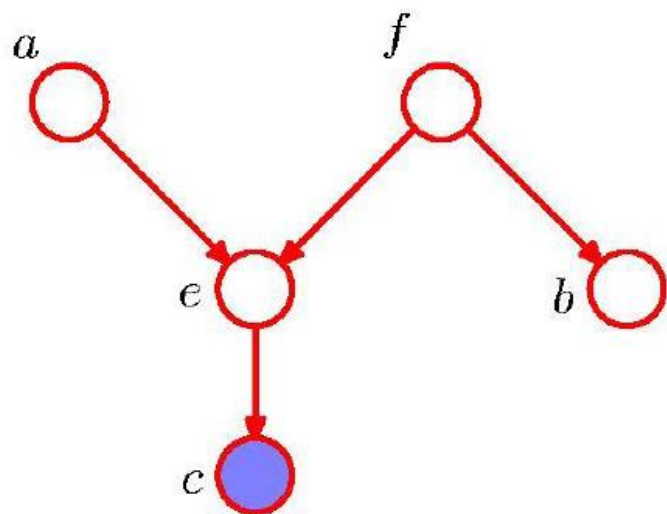


$a \perp\!\!\!\perp b \mid c$?



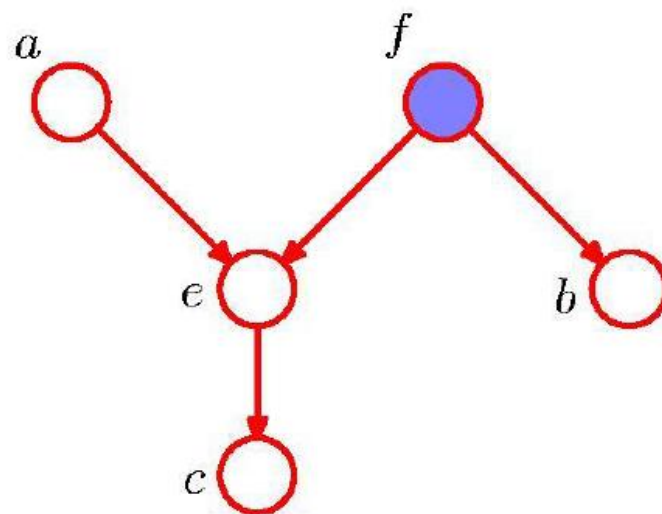
$a \perp\!\!\!\perp b \mid f$

贝叶斯网络中的D-分离



$a \perp\!\!\!\perp b \mid c$?

X

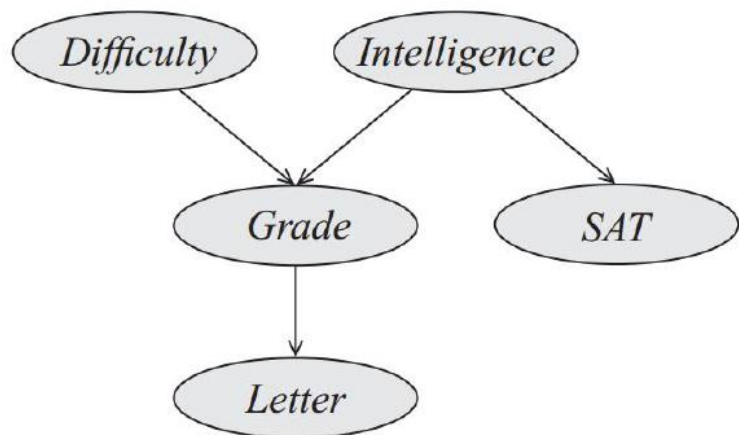


$a \perp\!\!\!\perp b \mid f$

✓

贝叶斯网络中的独立性

- **引理**: 父节点已知时, 该节点与其所有非后代的节点 (non-descendants) 满足 D-separated.
- **定理**: 父节点已知时, 该节点与其所有非后代的节点 (non-descendants) 条件独立。



$$\begin{aligned} p(I, D, G, S, L) \\ &= P(I)P(D|I)P(G|I, D)P(L|I, D, G)P(S|I, D, G, L) \\ &= P(I)P(D)P(G|I, D)P(L|G)P(S|I) \end{aligned}$$

Blue arrows indicate the simplifications: $P(D|I) \rightarrow P(D)$, $P(L|I, D, G) \rightarrow P(L|G)$, and $P(S|I, D, G, L) \rightarrow P(S|I)$.

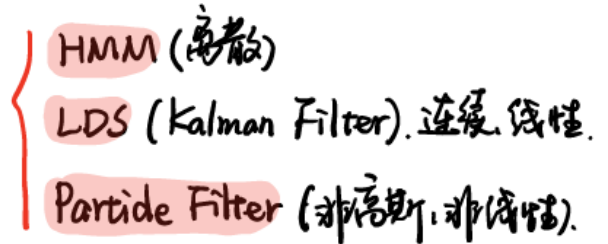
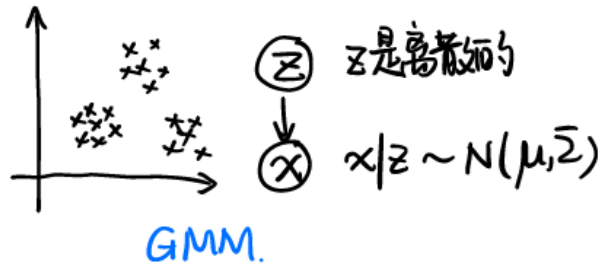
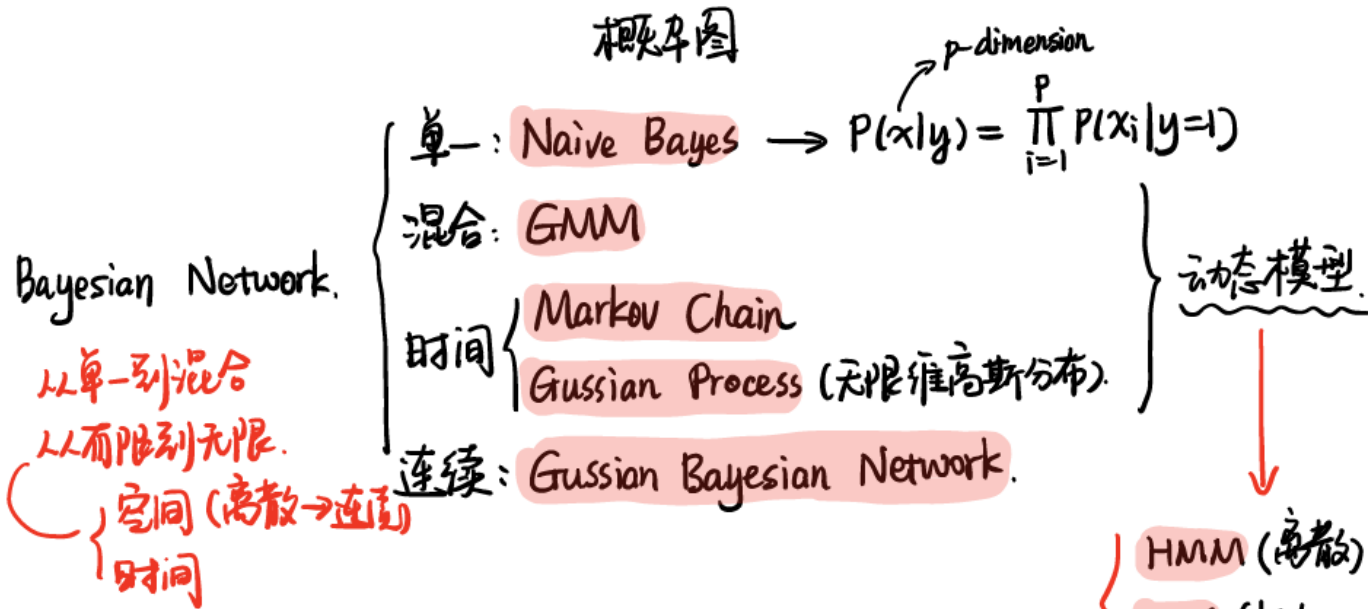
条件独立

$$\left\{ \begin{array}{l} P(D|I) = P(D) \\ P(L|G) = P(L|I, D, G) \\ P(S|I) = P(S|I, D, G, L) \end{array} \right.$$

贝叶斯网络具体例子

- 单一: Naive Bayes (朴素贝叶斯) $P(x|y) = \prod_{i=1}^p P(x_i|y=1)$
- 混合: GMM (混合高斯模型)
- 加入时间维度
 - Markov Chain、Gaussian Process (无限维高斯分布)
- 连续: Gaussian Bayesian Network
- 混合模型与时间结合起来: 动态模型
 - HMM (离散)、Kalman Filter (连续[高斯]、线性)、Particle Filter (非高斯、非线性)
- 总的来说, 用两句话总结这些模型: 从单一到混合、从有限到无限
- 两个角度: 空间 (随机变量的取值从离散到连续)、时间 (加入时间维度, 从某一时刻延长到无限时间)

贝叶斯网络具体例子



不确定知识表示和推理

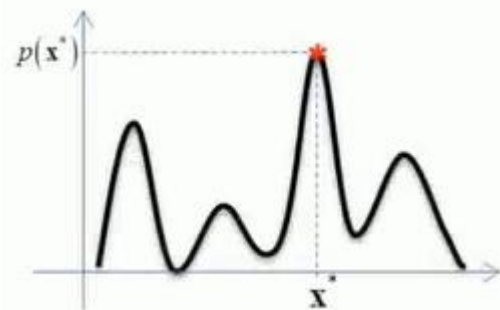
- 背景
- 概率论与图论基础
- 概率图模型的表示
- 概率图模型的推理
- 概率图模型的学习

推理(Inference)

- 已知联合概率分布: $p(\mathbf{x}) = p(x_1, x_2, \dots, x_N) = \frac{1}{Z} \prod_c \phi_c(\mathbf{x}_c)$

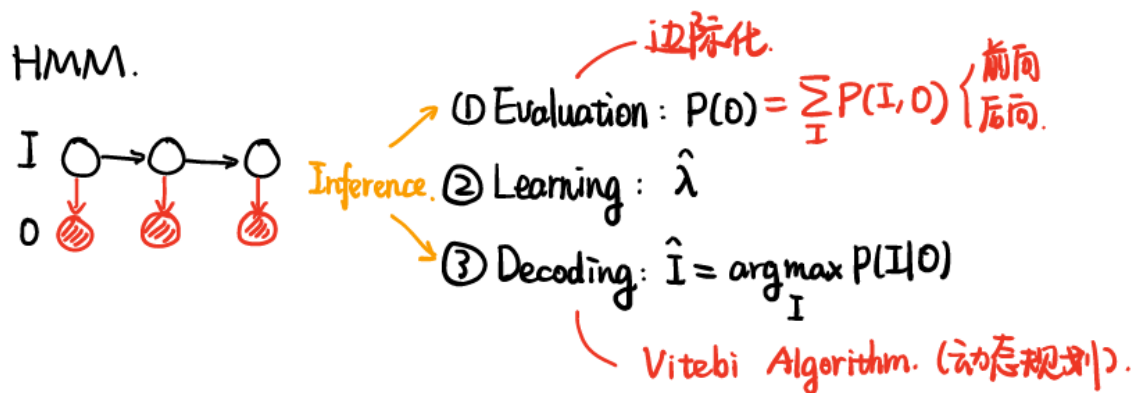
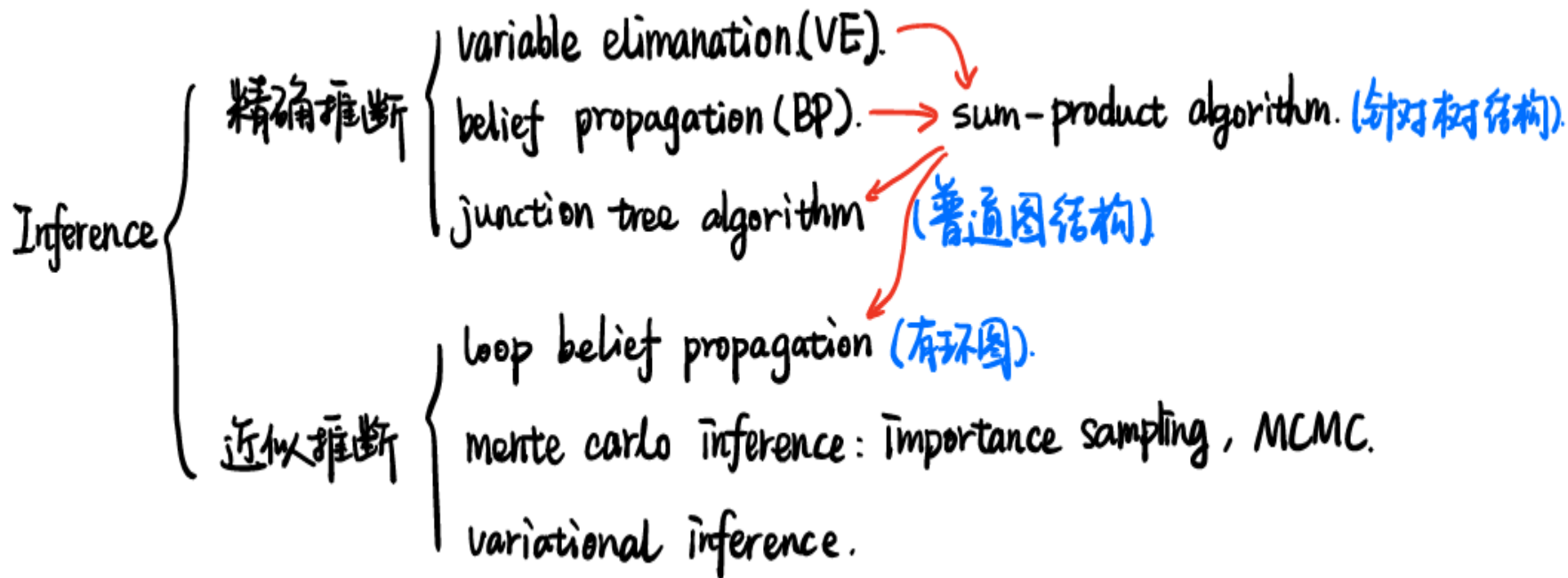
三类 概率 推理

- 边缘概率: $p(\mathbf{x}_\alpha) = \sum_{\mathbf{x} \setminus \mathbf{x}_\alpha} p(\mathbf{x})$
- 最大后验概率状态: $\mathbf{x}^* = \arg \max_{\mathbf{x} \in \mathcal{X}} p(\mathbf{x})$
- 求归一化因子: $Z = \sum_{\mathbf{x}} \prod_c \phi_c(\mathbf{x}_c)$



- 推理问题是概率图模型的核心问题
 - 推理相当于模型求解
 - 学习过程也蕴含推理过程
 - 推理复杂度高, 精确推理是NP难问题

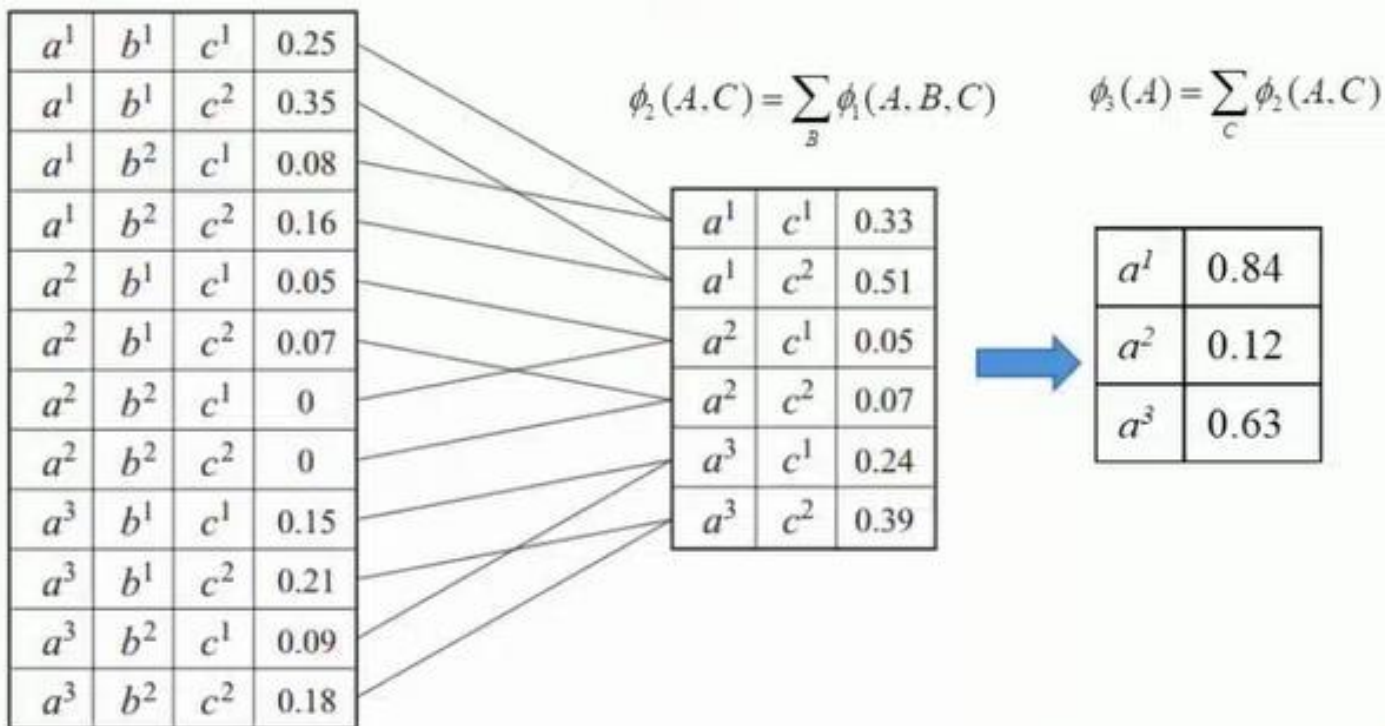
推理 (Inference)



HMM: dynamic bayesian network.

变量消元算法(VE)

- **变量消除** (Variable Elimination, **VE**) 基本思想: 通过从联合概率分别逐步消除变量, 来求解边缘概率



变量消元算法(VE)

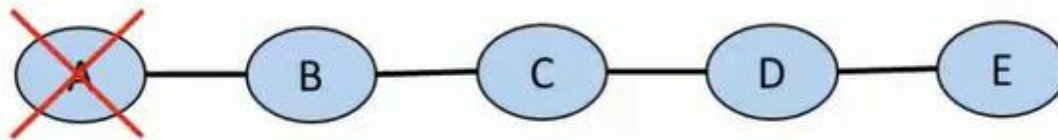
□ RMF(链状图模型): 求解节点边缘概率



$$\begin{aligned} P(E) &\propto \sum_D \sum_C \sum_B \sum_A \tilde{P}(A, B, C, D, E) \\ &= \sum_D \sum_C \sum_B \sum_A \phi_1(A, B) \phi_2(B, C) \phi_3(C, D) \phi_4(D, E) \\ &= \sum_D \sum_C \sum_B \phi_2(B, C) \phi_3(C, D) \phi_4(D, E) \sum_A \phi_1(A, B) \\ &= \sum_D \sum_C \sum_B \phi_2(B, C) \phi_3(C, D) \phi_4(D, E) \tau_1(B) \end{aligned}$$

变量消元算法(VE)

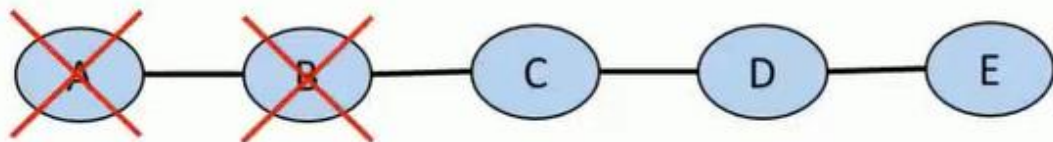
□ RMF(链状图模型): 求解节点边缘概率



$$\begin{aligned} P(E) &\propto \sum_D \sum_C \sum_B \sum_A \tilde{P}(A, B, C, D, E) \\ &= \sum_D \sum_C \sum_B \sum_A \phi_1(A, B) \phi_2(B, C) \phi_3(C, D) \phi_4(D, E) \\ &= \sum_D \sum_C \sum_B \phi_2(B, C) \phi_3(C, D) \phi_4(D, E) \sum_A \phi_1(A, B) \\ &= \sum_D \sum_C \sum_B \phi_2(B, C) \phi_3(C, D) \phi_4(D, E) \tau_1(B) \end{aligned}$$

变量消元算法(VE)

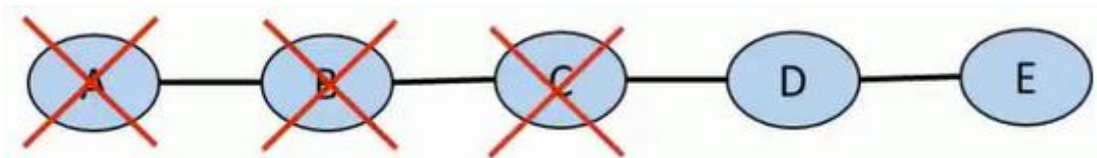
□ RMF(链状图模型): 求解节点边缘概率



$$\begin{aligned} P(E) &\propto \sum_D \sum_C \sum_B \phi_2(B, C) \phi_3(C, D) \phi_4(D, E) \tau_1(B) \\ &= \sum_D \sum_C \phi_3(C, D) \phi_4(D, E) \sum_B \phi_2(B, C) \tau_1(B) \\ &= \sum_D \sum_C \phi_3(C, D) \phi_4(D, E) \tau_2(C) \end{aligned}$$

变量消元算法(VE)

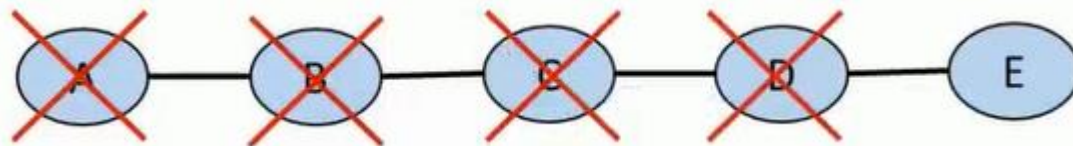
□ RMF(链状图模型): 求解节点边缘概率



$$\begin{aligned} P(E) &\propto \sum_D \sum_C \phi_3(C, D) \phi_4(D, E) \tau_2(C) \\ &= \sum_D \phi_4(D, E) \sum_C \phi_3(C, D) \tau_2(C) \\ &= \sum_D \phi_4(D, E) \tau_3(D) \end{aligned}$$

变量消元算法(VE)

□ RMF(链状图模型): 求解节点边缘概率



$$P(E) \propto \sum_D \phi_4(D, E) \tau_3(D)$$

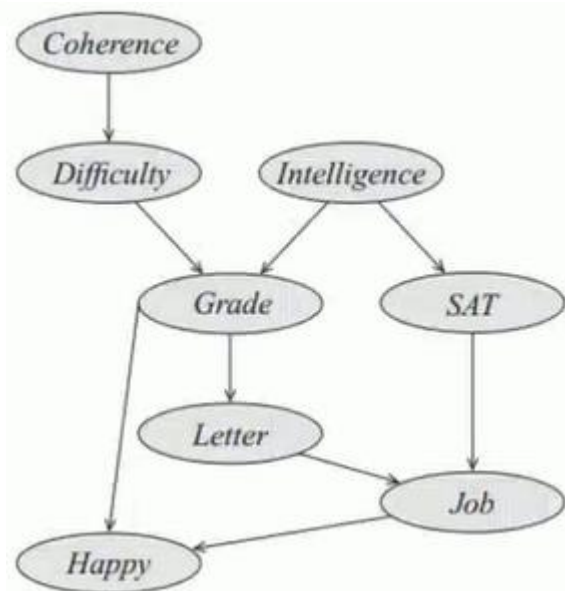
变量消元算法(VE)

□ 贝叶斯网络: 求解节点J的边缘概率

$$P(C, D, I, G, S, L, J, H)$$

$$= P(C)P(D|C)P(I)P(G|I, D)P(S|I)P(L|G)P(J|L, S)P(H|G, J)$$

$$= \phi_C(C)\phi_D(D, C)\phi_I(I)\phi_G(G, I, D)\phi_S(S, I)\phi_L(L, G)\phi_J(J, L, S)\phi_H(H, G, J)$$



变量消元算法(VE)

□ 贝叶斯网络: 求解节点J的边缘概率

$$P(C, D, I, G, S, L, J, H)$$

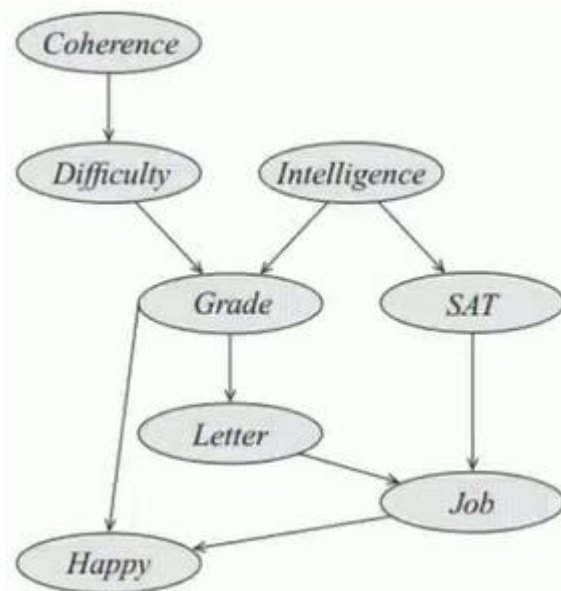
$$= P(C)P(D|C)P(I)P(G|I, D)P(S|I)P(L|G)P(J|L, S)P(H|G, J)$$

$$= \phi_C(C)\phi_D(D, C)\phi_I(I)\phi_G(G, I, D)\phi_S(S, I)\phi_L(L, G)\phi_J(J, L, S)\phi_H(H, G, J)$$

$$= \sum_{L, S, G, H, I, D} \phi_I(I)\phi_G(G, I, D)\phi_S(S, I)\phi_L(L, G)\phi_J(J, L, S)\phi_H(H, G, J)\tau_1(D)$$

1: 消除变量 C

$$\psi_1(C, D) = \phi_C(C)\phi_D(D, C)$$
$$\tau_1(D) = \sum_C \psi_1$$



变量消元算法(VE)

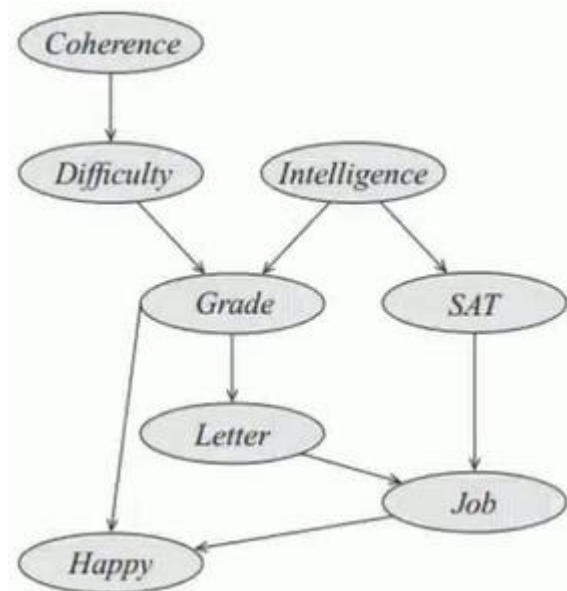
□ 贝叶斯网络: 求解节点J的边缘概率

$$= \sum_{L,S,G,H,I,D} \phi_I(I) \phi_G(G,I,D) \phi_S(S,I) \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_1(D)$$

$$= \sum_{L,S,G,H,I} \phi_I(I) \phi_S(S,I) \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_2(G,I)$$

2: 消除变量 D

$$\begin{aligned} \psi_2(G,I,D) &= \phi_G(G,I,D) \tau_1(D) \\ \tau_2(G,I) &= \sum_D \psi_2(G,I,D) \end{aligned}$$



变量消元算法(VE)

□ 贝叶斯网络: 求解节点J的边缘概率

$$= \sum_{L,S,G,H,I,D} \phi_I(I) \phi_G(G,I,D) \phi_S(S,I) \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_1(D)$$

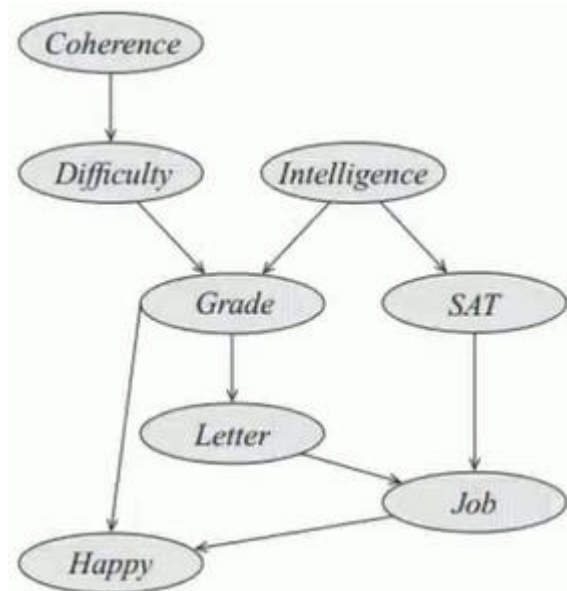
$$= \sum_{L,S,G,H,I} \phi_I(I) \phi_S(S,I) \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_2(G,I)$$

$$= \sum_{L,S,G,H} \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_3(G,S)$$

3: 消除变量 I

$$\psi_3(G,I,S) = \phi_I(I) \phi_S(S,I) \tau_2(G,I)$$

$$\tau_3(G,S) = \sum_I \psi_3(G,I,S)$$



变量消元算法(VE)

□ 贝叶斯网络: 求解节点J的边缘概率

$$= \sum_{L,S,G,H,I,D} \phi_I(I) \phi_G(G,I,D) \phi_S(S,I) \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_1(D)$$

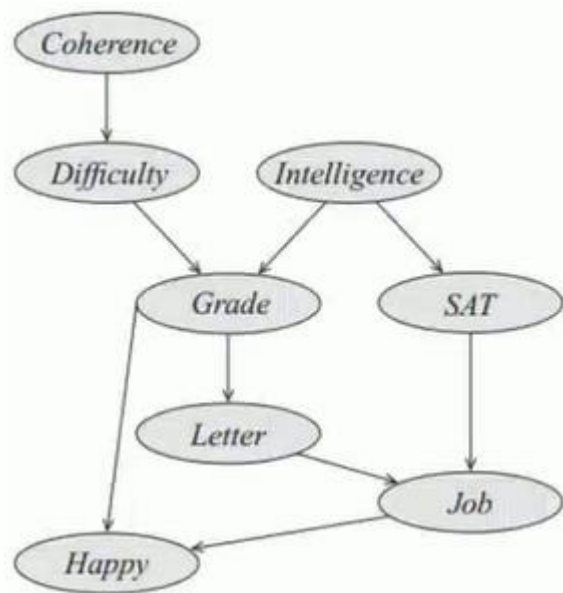
$$= \sum_{L,S,G,H,I} \phi_I(I) \phi_S(S,I) \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_2(G,I)$$

$$= \sum_{L,S,G,H} \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_3(G,S)$$

$$= \sum_{L,S,G} \phi_L(L,G) \phi_J(J,L,S) \tau_3(G,S) \tau_4(G,J)$$

4: 消除变量 H

$$\begin{aligned} \psi_4(G,J,H) &= \phi_H(H,G,J) \\ \tau_4(G,J) &= \sum_H \psi_4(G,J,H) \end{aligned}$$



变量消元算法(VE)

□ 贝叶斯网络: 求解节点J的边缘概率

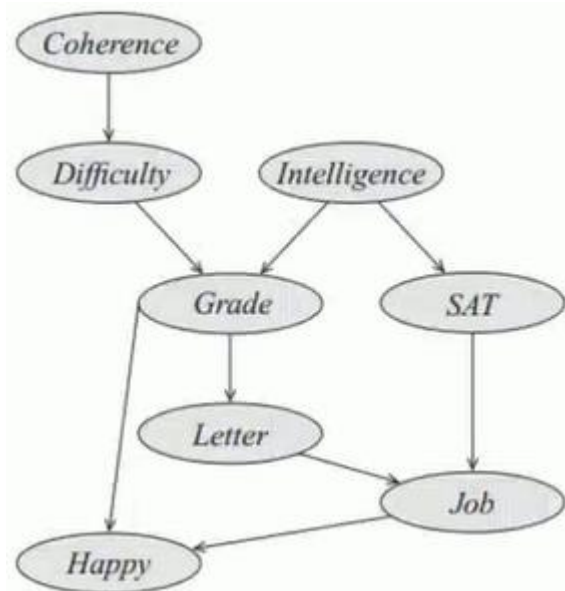
$$= \sum_{L,S,G,H,I,D} \phi_I(I) \phi_G(G,I,D) \phi_S(S,I) \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_1(D)$$

$$= \sum_{L,S,G,H,I} \phi_I(I) \phi_S(S,I) \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_2(G,I)$$

$$= \sum_{L,S,G,H} \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_3(G,S)$$

$$= \sum_{L,S,G} \phi_L(L,G) \phi_J(J,L,S) \tau_3(G,S) \tau_4(G,J)$$

$$= \sum_{L,S} \phi_J(J,L,S) \tau_5(L,S,J)$$



5: 消除变量 G

$$\psi_5(G,J,L,S) = \phi_L(L,G) \tau_3(G,S) \tau_4(G,J)$$
$$\tau_5(L,S,J) = \sum_G \psi_5(G,J,L,S)$$

变量消元算法(VE)

□ 贝叶斯网络: 求解节点J的边缘概率

$$= \sum_{L,S,G,H,I,D} \phi_I(I) \phi_G(G,I,D) \phi_S(S,I) \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_1(D)$$

$$= \sum_{L,S,G,H,I} \phi_I(I) \phi_S(S,I) \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_2(G,I)$$

$$= \sum_{L,S,G,H} \phi_L(L,G) \phi_J(J,L,S) \phi_H(H,G,J) \tau_3(G,S)$$

$$= \sum_{L,S,G} \phi_L(L,G) \phi_J(J,L,S) \tau_3(G,S) \tau_4(G,J)$$

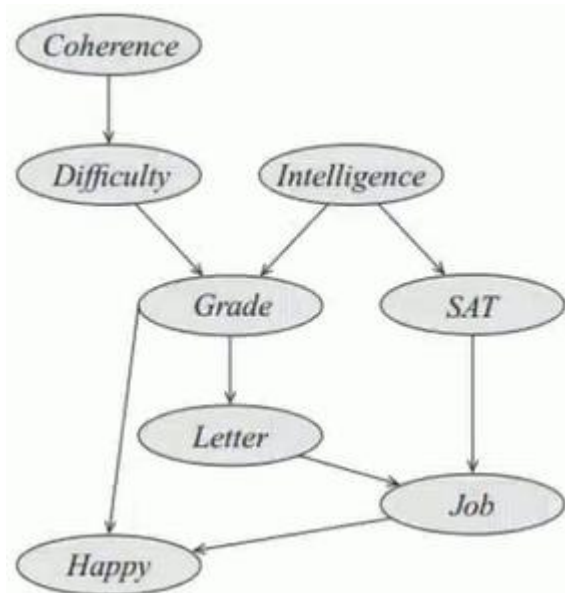
$$= \sum_{L,S} \phi_J(J,L,S) \tau_5(L,S,J)$$

$$= \sum_L \tau_6(L,J)$$

$$= \tau_7(J)$$

6: 消除变量 S

$$\psi_6(J,L,S) = \phi_J(J,L,S) \tau_5(L,S,J)$$
$$\tau_6(L,J) = \sum_S \psi_6(J,L,S)$$



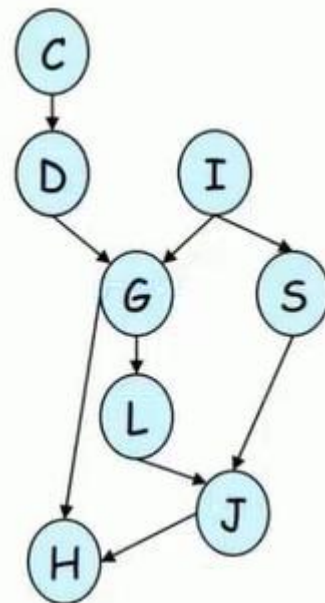
7: 消除变量 L

$$\psi_7(J,L) = \tau_6(L,J)$$
$$\tau_7(J) = \sum_L \psi_7(J,L)$$

变量消元算法(VE)

□ 贝叶斯网络: 求解节点J的边缘概率

消元变量	涉及因子	涉及变量	新因子
C	$\phi_C(C), \phi_D(D, C)$	C, D	$\tau_1(D)$
D	$\phi_G(G, I, D), \tau_1(D)$	G, I, D	$\tau_2(G, I)$
I	$\phi_I(I), \phi_S(S, I), \tau_2(G, I)$	G, S, I	$\tau_3(G, S)$
H	$\phi_H(H, G, J)$	H, G, J	$\tau_4(G, J)$
G	$\tau_4(G, J), \tau_3(G, S), \phi_L(L, G)$	G, J, L, S	$\tau_5(J, L, S)$
S	$\tau_5(J, L, S), \phi_J(J, L, S)$	J, L, S	$\tau_6(J, L)$
L	$\tau_6(J, L)$	J, L	$\tau_7(J)$



□ 其他消除顺序

消元变量	涉及因子	涉及变量	新因子
G	$\phi_G(G, I, D), \phi_L(L, G), \phi_H(H, G, J)$	G, I, D, L, J, H	$\tau_1(I, D, L, J, H)$
I	$\phi_I(I), \phi_S(S, I), \tau_1(I, D, L, S, J, H)$	S, I, D, L, J, H	$\tau_2(D, L, S, J, H)$
S	$\phi_J(J, L, S), \tau_2(D, L, S, J, H)$	D, L, S, J, H	$\tau_3(D, L, J, H)$
L	$\tau_3(D, L, J, H)$	D, L, J, H	$\tau_4(D, J, H)$
H	$\tau_4(D, J, H)$	D, J, H	$\tau_5(D, J)$
C	$\tau_5(D, J), \phi_C(C), \phi_D(D, C)$	D, J, C	$\tau_6(D, J)$
D	$\tau_6(D, J)$	D, J	$\tau_7(J)$

变量消元算法(VE)

Sum-Product-VE(Φ, \mathbf{Z})

// Φ 为因子集, \mathbf{Z} 为消元变量集

1: 令变量消元顺序为 Z_1, \dots, Z_k

2: for $i = 1, \dots, k$

3: $\Phi \leftarrow \text{Sum-Product-Eliminate-Var}(\Phi, Z_i)$

4: $\phi^* \leftarrow \prod_{\phi \in \Phi} \phi$

Sum-Product-Eliminate-Var(Φ, Z)

1: $\Phi' \leftarrow \{\phi \in \Phi : Z \in \text{Scope}[\phi]\}$

2: $\Phi'' \leftarrow \Phi - \Phi'$

3: $\psi \leftarrow \prod_{\phi \in \Phi'} \phi$

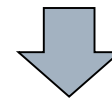
4: $\tau \leftarrow \sum_Z \psi$

5: return $\Phi'' \cup \{\tau\}$

• 求条件概率: $p(J|I=i, H=h) = \frac{p(J, I=i, H=h)}{p(I=i, H=h)}$

• 变量消元顺序: C, D, G, S, L

$$P(C)P(D|C)P(I)P(G|I,D)P(S|I)P(L|G)P(J|L,S)P(H|G,J) \\ = \phi_C(C)\phi_D(D,C)\phi_I(I)\phi_G(G,I,D)\phi_S(S,I)\phi_L(L,G)\phi_J(J,L,S)\phi_H(H,G,J)$$



$$\sum_{L,S,G,D,C} \phi_C(C)\phi_D(D,C)\phi'_G(G,D)\phi'_S(S)\phi_L(L,G)\phi_J(J,L,S)\phi'_H(H,G,J)$$

贝叶斯网络的推理

□ 变量消元 (Variable Elimination, VE) 算法

- 给定一个贝叶斯网络的一系列条件概率表 F , 查询变量 Q , 证据变量的取值 $E=e$, 剩余变量 Z , 需要计算 $Pr(Q | E)$, 则VE算法的流程如下:
- 对于任意因子 $f \in F$, 如果该因子涉及 E 中的一个或多个变量, 则将其替换为限制后的因子 $f_{E=e}$;
- 给定一个消元顺序, 对于变量 $Z_j \in Z$:
 - 令 f_1, f_2, \dots, f_k 为 F 中包含 Z_j 的所有因子;
 - 将这些因子相乘并在 Z_j 上求和后得到新的因子:
$$g_j = \sum_{Z_j} f_1 \times f_2 \times \dots \times f_k;$$
 - 从 F 中删掉 f_1, f_2, \dots, f_k , 并将 g_j 加入 F 中。
- 剩下的因子将只涉及查询变量 Q , 对这些因子相乘并归一化后得到 $Pr(Q | E)$ 。

变量消元算法(VE)

□ VE过程理解

$$\textcircled{a} \rightarrow \textcircled{b} \rightarrow \textcircled{c} \rightarrow \textcircled{d}^? \quad (\text{假设 } a, b, c, d \text{ 均为离散的二值 random variable})$$

$a, b, c, d \in \{0, 1\}$

$$P(d) = \sum_{a, b, c} P(a, b, c, d)$$

$$= \sum_{a, b, c} P(a) \cdot P(b|a) \cdot P(c|b) \cdot P(d|c)$$

$$= P(a=0) \cdot P(b=0|a=0) \cdot P(c=0|b=0) \cdot P(d=0|c=0)$$

$$+ P(a=1) \cdot P(b=0|a=1) \cdot P(c=0|b=0) \cdot P(d=0|c=0)$$

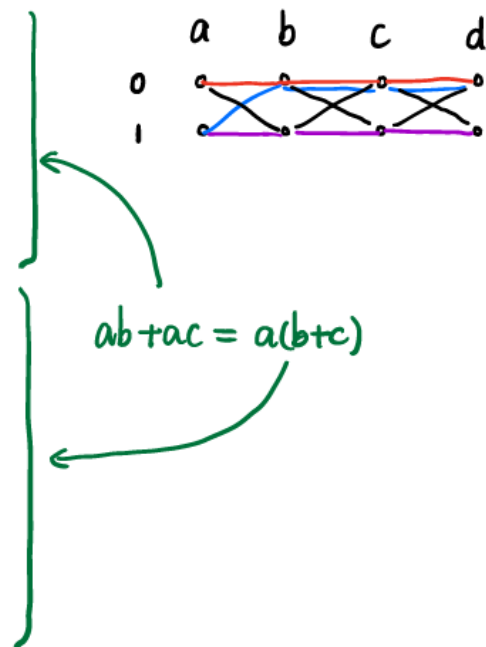
$$+ \dots + P(a=1) \cdot P(b=1|a=1) \cdot P(c=1|b=1) \cdot P(d=1|c=1)$$

$$= \sum_{b, c} P(c|b) \cdot P(d|c) \cdot \underbrace{\sum_a \frac{P(a) \cdot P(b|a)}{\phi_a(b)}}_{\phi_a(b)}$$

$$= \sum_c P(d|c) \cdot \underbrace{\sum_b P(c|b) \cdot \phi_a(b)}_{\phi_b(c)}$$

$$= \phi_c(d)$$

(乘法对加法的分配律)



变量消元算法(VE)

□ VE过程理解



$$P(a,b,c,d,e) = P(a) \cdot P(b|a) \cdot P(c|b) \cdot P(d|c) \cdot P(e|d).$$

$$P(e) = \sum_{a,b,c,d} P(a,b,c,d,e)$$

$$= \sum_d P(e|d) \cdot \sum_c P(d|c) \cdot \sum_b P(c|b) \cdot \underbrace{\sum_a P(b|a) \cdot P(a)}_{m_{a \rightarrow b}(b)}$$
$$\quad \quad \quad m_{b \rightarrow c}(c).$$

forward algorithm

$$P(c) = \sum_{a,b,d,e} P(a,b,c,d,e).$$

$$= \underbrace{\left(\sum_b P(c|b) \cdot P(b|a) \cdot P(a) \right)}_{m_{b \rightarrow c}(c)} \cdot \left(\sum_d P(d|c) \sum_e P(e|d) \right)$$

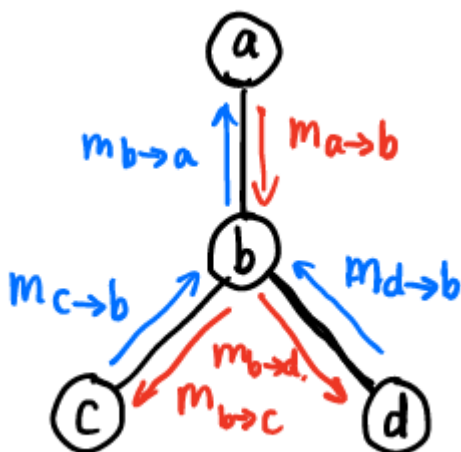
forward-backward algorithm

Chain \rightarrow Tree

有向 \rightarrow 无向

信念传播算法(BP)

- 信念传播算法 (Belief Propagation, BP): 在树状图模型上能收敛, 且能达到精确推理。但在一般图模型上, BP为近似推理算法。



$$P(a,b,c,d) = \frac{1}{Z} \varphi_a(a) \cdot \varphi_b(b) \cdot \varphi_c(c) \cdot \varphi_d(d) \cdot \varphi_{ab}(a,b) \cdot \varphi_{bc}(b,c) \cdot \varphi_{bd}(b,d).$$

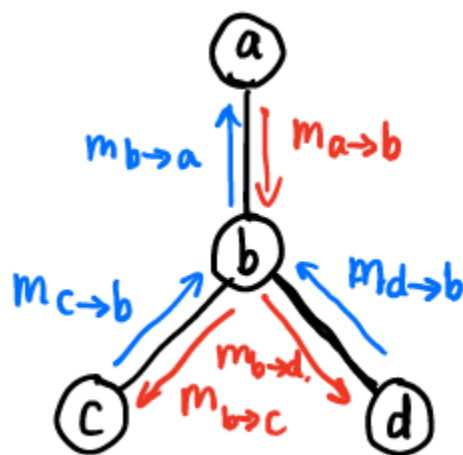
$$P(a) = \sum_{b,c,d} P(a,b,c,d).$$

Generalize

$$\left\{ \begin{array}{l} m_{b \rightarrow a}(x_a) = \sum_{x_b} \varphi_{ab} \varphi_b m_{c \rightarrow b}(x_b) \cdot m_{d \rightarrow b}(x_b) \\ P(a) = \varphi_a m_{b \rightarrow a}(x_a) \\ m_{j \rightarrow i}(x_i) = \sum_{x_j} \varphi_{ij} \varphi_j \prod_{k \in \text{NB}(j)-i} m_{k \rightarrow j}(x_j) \\ P(x_i) = \varphi_i \prod_{k \in \text{NB}(i)} m_{k \rightarrow i}(x_i) \end{array} \right.$$

★ 不要直接去求边缘概率 $P(a), P(b), P(c), P(d)$
只需求 $m_{i \rightarrow j}$

信念传播算法(BP)



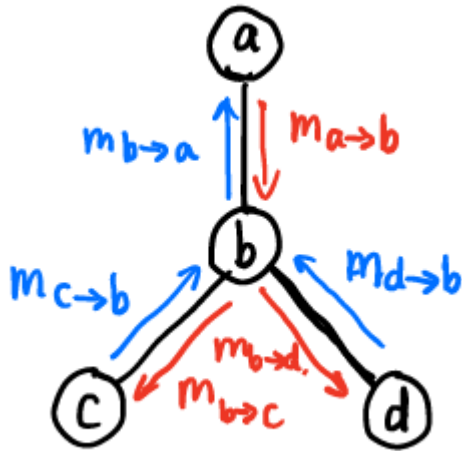
$$m_{b \rightarrow a} = \sum_b \varphi_{ab} \cdot \underbrace{\varphi_b}_{\text{self}} \cdot \underbrace{m_{c \rightarrow b} \cdot m_{d \rightarrow b}}_{\text{children}} \quad (\text{belief})$$

$$\begin{cases} \text{belief}(b) = \varphi_b \cdot \text{children} \\ m_{b \rightarrow a} = \sum_b \varphi_{ab} \cdot \text{belief}(b). \end{cases}$$

BP = VE + Caching.

↳ (直接求 $m_{ij} \Rightarrow P(x_i)$)
图的遍历

信念传播算法(BP)



$$m_{j \rightarrow i} = \sum_b \varphi_{ab} \cdot \underbrace{\varphi_b}_{\text{self}} \cdot \underbrace{m_{c \rightarrow b} \cdot m_{d \rightarrow b}}_{\text{children}} \quad (\text{belief})$$

$$\begin{cases} \text{belief}(b) = \varphi_b \cdot \text{children} \\ m_{b \rightarrow a} = \sum_b \varphi_{ab} \cdot \text{belief}(b) \end{cases}$$

BP = VE + Caching.

↳ (直接求 $m_{ij} \Rightarrow P(x_i)$)
图的遍历

BP (Sequential Implementation)

① get root, assume a is root

② collect message. for x_i in NB(root): collectmsg(x_i)

③ distribute message. for x_j in NB(root): distribute(x_j)

可得 m_{ij} for all $i, j \in V$

求 $P(x_k)$ KEV.

BP (Parallel Implementation).

不确定知识表示和推理

- 背景
- 概率论与图论基础
- 概率图模型的表示
- 概率图模型的推理
- 概率图模型的学习

贝叶斯网络的参数学习

- 条件概率表的学习方法:
 - 极大似然
 - 期望-最大化

贝叶斯网络的参数学习

□ 条件概率表的学习方法：

- 极大似然
- 期望-最大化

• 示例1：

- 某超市销售的糖果有2种口味 (*cherry*和*lime*)
- 假设尝到*c*颗*cherry*口味的和*l*颗*lime*口味的

$Pr(F = \textit{cherry})$
θ



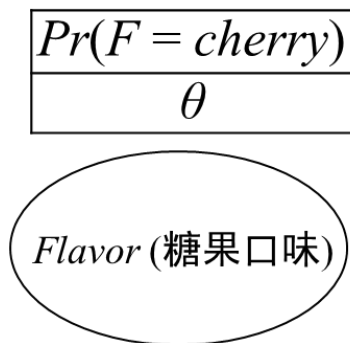
贝叶斯网络的参数学习

□ 条件概率表的学习方法：

- 极大似然
- 期望-最大化

• 示例1：

- 某超市销售的糖果有2种口味 (*cherry*和*lime*)
- 假设尝到*c*颗*cherry*口味的和*l*颗*lime*口味的
记上述观察到的数据为*d*



$$Pr(d | h_{\theta}) = \theta^c (1 - \theta)^l$$

$$\log Pr(d | h_{\theta}) = c \log \theta + l \log(1 - \theta)$$

$$d(\log Pr(d | h_{\theta})) / d\theta = c/\theta - l/(1 - \theta)$$

$$c/\theta - l/(1 - \theta) = 0 \Rightarrow \theta = c/(c + l)$$

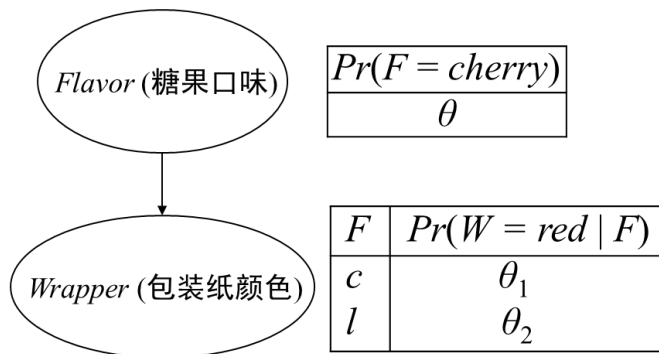
贝叶斯网络的参数学习

□ 条件概率表的学习方法：

- 极大似然
- 期望-最大化

• 示例2：

- 某超市销售的糖果有2种口味 (*cherry*和*lime*) , 包装纸的颜色分为绿色和红色
- 糖果的口味决定了包装纸的颜色
- 假设尝到*c*颗*cherry*口味的 (其中*g_c*颗为绿色包装纸, *r_c*颗为红色包装纸) , 以及*l*颗*lime*口味的 (其中*g_l*颗为绿色包装纸, *r_l*颗为红色包装纸)



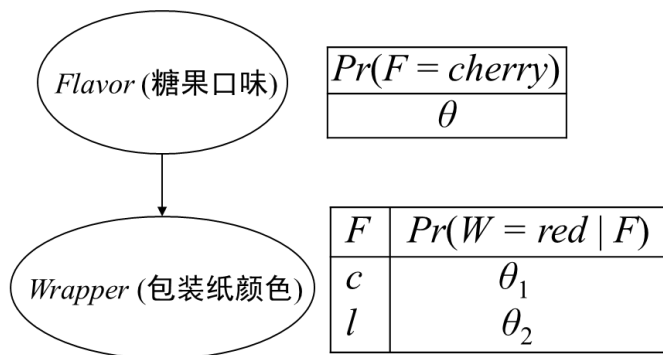
贝叶斯网络的参数学习

□ 条件概率表的学习方法：

- 极大似然
- 期望-最大化

• 示例2：

- 某超市销售的糖果有2种口味 (*cherry*和*lime*) , 包装纸的颜色分为绿色和红色
- 糖果的口味决定了包装纸的颜色
- 假设尝到*c*颗*cherry*口味的 (其中*g_c*颗为绿色包装纸, *r_c*颗为红色包装纸) , 以及*l*颗*lime*口味的 (其中*g_l*颗为绿色包装纸, *r_l*颗为红色包装纸)



$$Pr(d | h_{\theta, \theta_1, \theta_2}) = \theta^c \theta_1^{r_c} (1 - \theta_1)^{g_c} (1 - \theta)^l \theta_2^{r_l} (1 - \theta_2)^{g_l}$$

$$c/\theta - l/(1 - \theta) = 0 \Rightarrow \theta = c/(c + l)$$

$$r_c/\theta_1 - g_c/(1 - \theta_1) = 0 \Rightarrow \theta_1 = r_c/(r_c + g_c)$$

$$r_l/\theta_2 - g_l/(1 - \theta_2) = 0 \Rightarrow \theta_2 = r_l/(r_l + g_l)$$

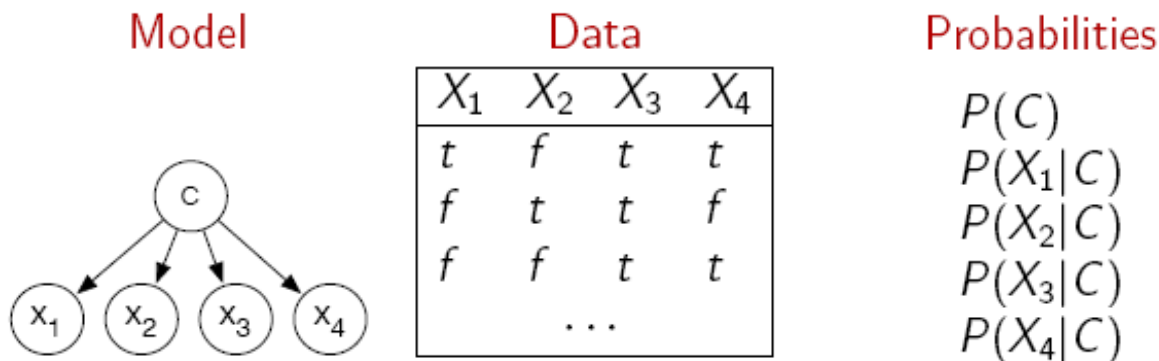
贝叶斯网络的参数学习

□ 条件概率表的学习方法：

- 极大似然
- 期望-最大化

• 示例3：

- 不可观测的 k 值随机变量 C
- 可观测的二值属性 X_1 至 X_4



贝叶斯网络的参数学习

□ 条件概率表的学习方法：

- 极大似然
- 期望-最大化

假设 $k = 3$ 。**期望步**：随机初始化条件概率表，通过贝叶斯定理，可以计算每行数据的后验概率，例如 $Pr(C = 1 \mid X_1 = t, X_2 = f, X_3 = t, X_4 = t)$ 、 $Pr(C = 2 \mid X_1 = t, X_2 = f, X_3 = t, X_4 = t)$ 、 $Pr(C = 3 \mid X_1 = t, X_2 = f, X_3 = t, X_4 = t)$ ，假设分别为0.407、0.121、0.472，由此可以得到完备化的数据实例。

• 示例3：

- 不可观测的 k 值随机变量 C
- 可观测的二值属性 X_1 至 X_4

$A[X_1, \dots, X_4, C]$					
X_1	X_2	X_3	X_4	C	Count
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
t	f	t	t	1	40.7
t	f	t	t	2	12.1
t	f	t	t	3	47.2
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots

贝叶斯网络的参数学习

□ 条件概率表的学习方法：

- 极大似然
- 期望-最大化

• 示例3：

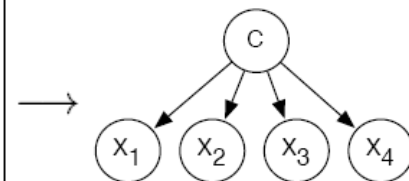
- 不可观测的 k 值随机变量 C
- 可观测的二值属性 X_1 至 X_4

假设 $k = 3$ 。**最大化步**：通过期望步得到的完备化数据实例，采用极大似然法，重新计算条件概率表。

$$P(C=v_i) = \frac{\sum_{t \models C=v_i} \text{Count}(t)}{\sum_t \text{Count}(t)}$$

$$P(X_k = v_j | C=v_i) = \frac{\sum_{t \models C=v_i \wedge X_k=v_j} \text{Count}(t)}{\sum_{t \models C=v_i} \text{Count}(t)}$$

X_1	X_2	X_3	X_4	C	Count
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
t	f	t	t	1	40.7
t	f	t	t	2	12.1
t	f	t	t	3	47.2
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots



贝叶斯网络的学习

- 在给定一个数据样本集合的前提下，寻找一个与训练样本集匹配最好的网络结构，被称为贝叶斯网络的结构学习。
- 搜索最优的网络结构是一个NP难问题，从数据中学习贝叶斯网络的结构一直是研究的热点。

- [1] C. K. Chow, C. N. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3): 462-467, 1968.
- [2] N. Wermuth, S. L. Lauritzen. Graphical and recursive models for contingency tables. *Biometrika*, 72: 537-552, 1983.
- [3] G. Rebane, J. Pearl. The recovery of causal polytrees from statistical data. *UAI*, 222-228, 1987.
- [4] G. F. Cooper, E. Herskovits. A bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9: 309-347, 1992.
- [5] X. Zheng, et al. DAGs with no tears: Continuous optimization for structure learning. *NeurIPS*, 9492-9503, 2018.
- [6] S. Lachapelle, et al. Gradient-based neural DAG learning. *ICLR*, 2020.
- [7] Y. Bengio, et al. A meta-transfer objective for learning to disentangle causal mechanisms. *ICLR*, 2020.
- [8] Y. Luo, J. Peng, J. Ma. When causal inference meets deep learning. *Nature Machine Intelligence*, 2: 426-427, 2020.

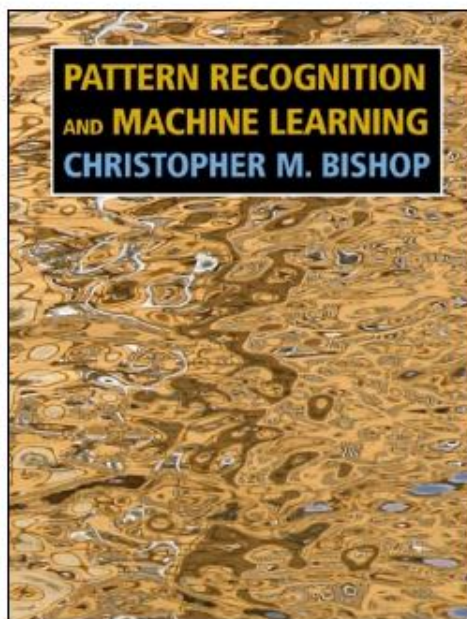
贝叶斯网络的学习

- 贝叶斯网络结构学习常用方法：
 - **基于约束的结构学习** (constraint-based structure learning) : 将贝叶斯网络看作是一种独立关系的表示。尝试对数据中的条件依赖和条件独立关系进行检验, 找到能够对这些依赖和独立关系给出最好解释的某个网络。
 - **基于得分的结构学习** (score-based structure learning) : 定义模型对观测数据拟合程度的得分函数, 通过最大化得分函数完成结构学习。

概率图模型参考资料

□ Textbooks:

- Daphne Koller and Nir Friedman, Probabilistic Graphical Models
- M. I. Jordan, An Introduction to Probabilistic Graphical Models



<http://research.microsoft.com/~cmbishop>