

Machine Learning

有向图模型

Bayes nets

华中科技大学计算机学院
王天江

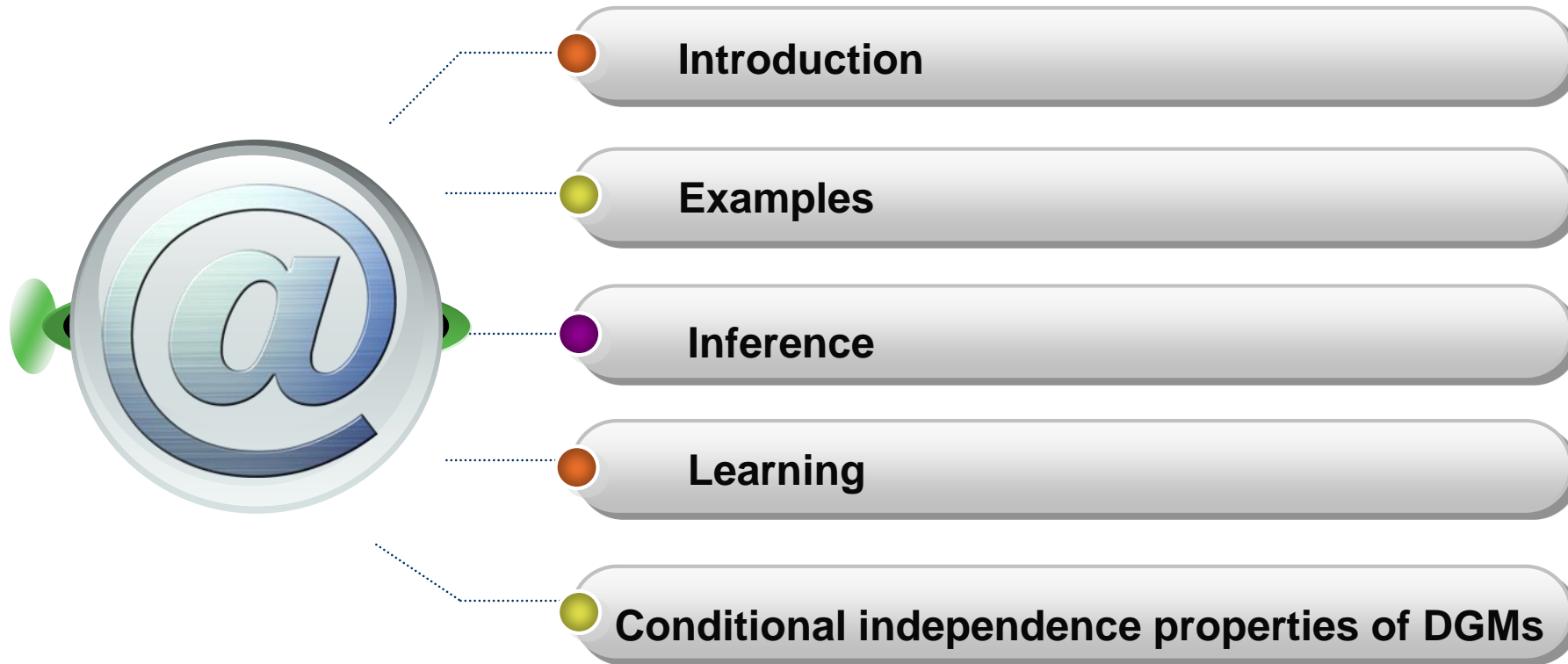


第6章：有向图模型

(*Bayes nets*)



第6章：有向图模型



概述



问题的提出

- 我们需要观察多个相关变量，例如
- 文档中的单词
- 生物芯片中的基因
- 如何方便表示相关变量的联合分布 $p(\mathbf{x}|\theta)$?
- 给定一组变量，如何在合理的计算时间内使用这个分布来推断另一组变量?
- 如何学习这个分布的参数?



概率计算的链式法则

- 链式法则公式：

$$p(x_{1:V}) = p(x_1)p(x_2 | x_1)p(x_3 | x_2, x_1) \cdots p(x_V | x_{1:V-1})$$

- 当 t 变大时， $p(x_t | x_{1:t-1})$ 的表示将变得越来越复杂
- 例如，设所有变量都有 K 个状态.
 - $p(x_1)$ 可表示为一个含 $O(K)$ 个参数的表，
 - $p(x_2 | x_1)$ 可表示为一个含 $O(K^2)$ 个参数的表
 - $p(x_3 | x_1, x_2)$ 可表示为一个含 $O(K^3)$ 个参数的3d表.
 - 以此类推，模型将包含 $O(K^V)$ 个参数.



随机矩阵 (*stochastic matrix*)

➤ 随机矩阵是一个条件概率的2维表:

➤ $p(x_2 = j | x_1 = i) = T_{ij}$;

➤ 满足约束: $\sum_j T_{ij} = 1$ for all rows i , $0 \leq T_{ij} \leq 1$ for all entries

➤ 例如

x	1	...	K
p	0.1	...	0.15

x_2/x_1	1	...	K
1	0.1	...	0.15
/	/	...	/
K	0.2	...	0.3



条件概率表 (CPTs)

➤ 对于多维变量，其联合发布参数就变得很多.

■ 因此，需要用多个表格来表达这些参数.

➤ 这些表格称为条件概率表

x1	x2	x3=	1	...	K		
K	x1	x2	x3=	1	...	K	
/	/	x1	x2	x3=	1	...	K
K	/	1	1		0.1	...	0.14
	/	/	/		/		/
		1	K		0.13	...	0.16



条件独立性 (CI)

- 当存在: $p(X, Y | Z) = p(X|Z)p(Y |Z)$
 - 称 X 、 Y 在给定条件 Z 下, 是条件独立的
 - 记为: $X \perp Y | Z$,
 - 即有, $X \perp Y | Z \Leftrightarrow p(X, Y | Z) = p(X|Z)p(Y |Z)$



Markov 链

➤ 一阶 **Markov** 假设:

➤ $x_{t+1} \perp x_{1:t-1} | x_t$

➤ 这表示 “给定现在, 未来独立于过去”

➤ 基于这个假设, 联合分布可表示成:

$$p(x_{1:v}) = p(x_1) \prod_{1 \leq t \leq v} p(x_t | x_{t-1})$$

➤ 这称为一阶 **Markov** 链



图模型 (GM)

- 定义 $1d$ 序列上的分布时，一阶马尔可夫模型是有用的
- 定义二维图像、三维视频或更高维变量的分布时，使用图模型更方便
- 图模型用图来表示联合分布
 - 节点：随机变量，边：条件依赖
- 图模型类型
 - 有向图模型
 - 无向图模型
 - 有向与无向结合模型



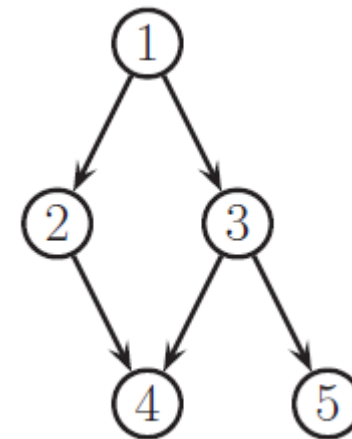
有向图模型(*DGM*)

❖ 用于表示随机变量之间依赖关系的有向无环图模型（DAG），被称为**贝叶斯网络**

- 这些模型也被称为**信念网络**。

❖ 图说明：

- 节点：代表随机变量。
- 边：表示从父节点到子节点的直接依赖关系
- 条件概率表：每个节点关联一个条件概率表，定义了给定其父节点取值时该节点的概率分布。



有向图模型的特点

❖ 局部马尔可夫假设：

- 每个节点仅依赖于它的直接父节点，与其它非后代节点条件独立。这可简化联合概率分布计算
- $p(x_{1:V}|G) = \prod_{t=1}^V p(x_t|x_{pa(t)})$
- 可用三种基本图形模式表达：
 - ⑩ 链式结构：如果存在 $A \rightarrow B \rightarrow C$ ，那么在给定 B 的情况下， A 和 C 条件独立
 - ⑩ 分叉结构：如果存在 $A \rightarrow B$ 和 $A \rightarrow C$ ，那么在给定 A 的情况下， B 和 C 条件独立。
 - ⑩ 碰撞结构：如果存在 $A \rightarrow C \leftarrow B$ ，如果未给定 C ， A 和 B 不独立；如果给定 C ， A 和 B 条件独立。

❖ 全局独立性：

- 从图结构中可以推导出更广泛的条件独立性声明，进一步简化模型复杂度。

❖ 紧凑表示：

- 明确编码变量间的依赖关系，更紧凑表示复杂联合概率分布，避免冗余信息

❖ 推理能力：

- 支持在图上进行高效的推理



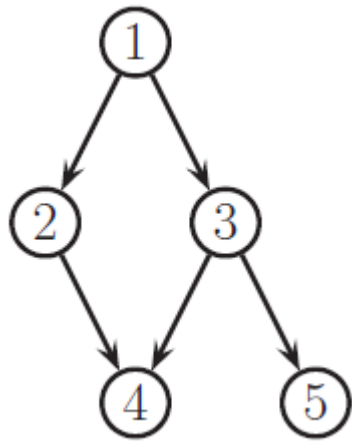
有向图模型举例

❖ 比如有个联合概率分布：

$$p(x_{1:5}) = p(x_1)p(x_2 \mid x_1)p(x_3 \mid x_1, x_2)p(x_4 \mid x_1, x_2, x_3)p(x_5 \mid x_1, x_2, x_3, x_4)$$

$$p(x_{1:5}) = p(x_1)p(x_2 \mid x_1)p(x_3 \mid x_1)p(x_4 \mid x_2, x_3)p(x_5 \mid x_3)$$

❖ 可以表示为：



几个例子



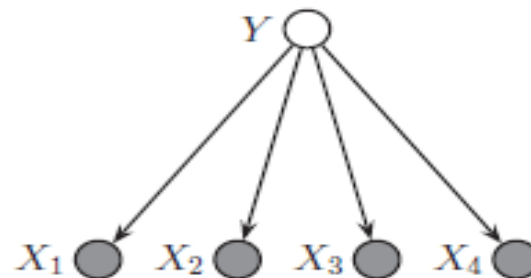
朴素贝叶斯分类器

❖ 朴素贝叶斯分类器假定

- 给定类标签后，特征是条件独立的

❖ 联合分布的表达式与网络图：

$$p(y, x) = p(y) \prod_{j=1}^D p(x_j | y)$$



半朴素贝叶斯分类器

❖ 朴素贝叶斯假设往往很难成立，因此，希望对属性条件独立性假设放松一点

- 产生了“半朴素贝叶斯分类器” (semi-naive Bayes classifiers), 基本思路:

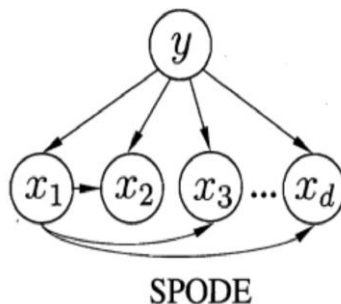
- ⑩ 适当考虑一部分属性间比较强的相互依赖信息

- ⑩ 采用“独依赖估计” (One-Dependent Estimator, ODE): 假设每个属性除类别外，最多只依赖一个其他属性

$$p(C|x) \propto p(C) \prod_{i=1}^d p(x_i|C, pa_i)$$

- 例如，超父结构 (super-parent): 假设所有属性都依赖于同一个属性

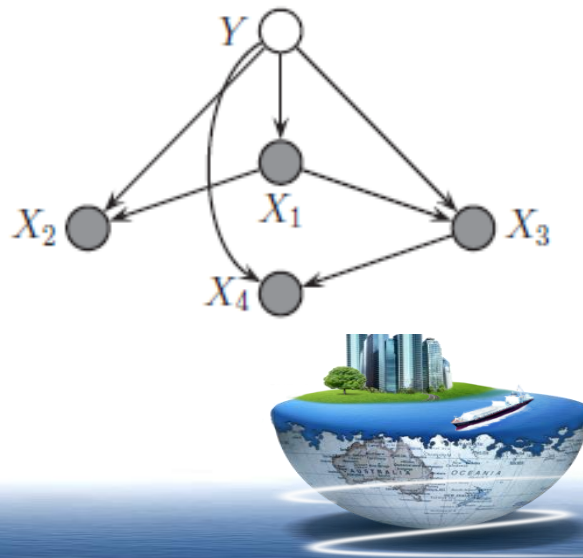
- ⑩ SPODE (Super-Parent ODE)模型:



树增强 (*tree-augmented*) 的朴素贝叶斯分类器

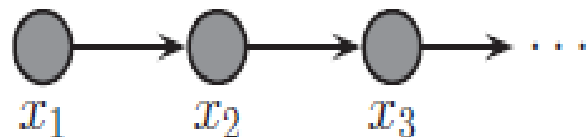
- ❖ 朴素贝叶斯分类器假设特征之间条件独立，这种假设在现实中往往不成立
- ❖ 树增强朴素贝叶斯分类器是对朴素贝叶斯分类器的一种改进
 - 允许特征之间存在一定的依赖关系，从而更好地捕捉特征之间的相关性
 - 这些依赖关系必须符合树结构
- ❖ 它实际上是一个半朴素贝叶斯分类器

⑩ 特征之间可以形成一棵树结构，意味着每个特征最多只有一个父特征

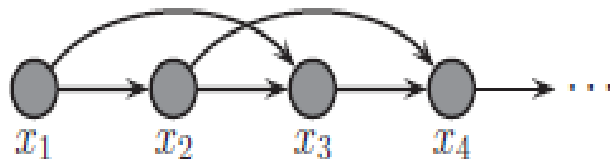


Markov 模型

❖ 一阶和二阶Markov链是有向无环图：



(a)



(b)

$$p(x_{1:T}) = p(x_1)p(x_2 | x_1)p(x_3 | x_2)\dots = p(x_1)\prod_{t=2}^T p(x_t | x_{t-1})$$

$$p(x_{1:T}) = p(x_1)p(x_2 | x_1)p(x_3 | x_1, x_2)p(x_4 | x_2, x_3)\dots = p(x_1)p(x_2 | x_1)\prod_{t=3}^T p(x_t | x_{t-1}, x_{t-2})$$

❖ 类似地可创建高阶马尔可夫模型

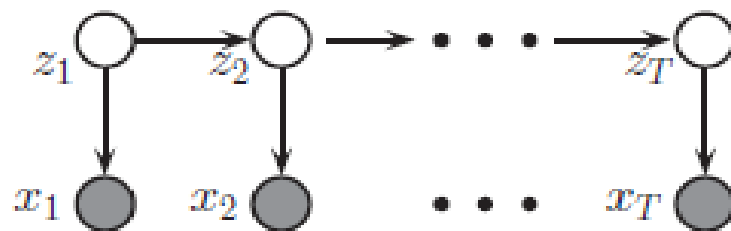
❖ 对于高阶模型，参数的数量将激增。



隐马尔科夫模型(HMM)

❖ 由一阶马尔科夫链定义的隐过程

➤ 其观察过程是带噪声的



➤ z_t 为 t 时刻的隐变量

➤ x_t 为 t 时刻的观察变量

➤ $p(z_t/z_{t-1})$: 转移模型, $p(x_t/z_t)$: 观测模型



医疗诊断

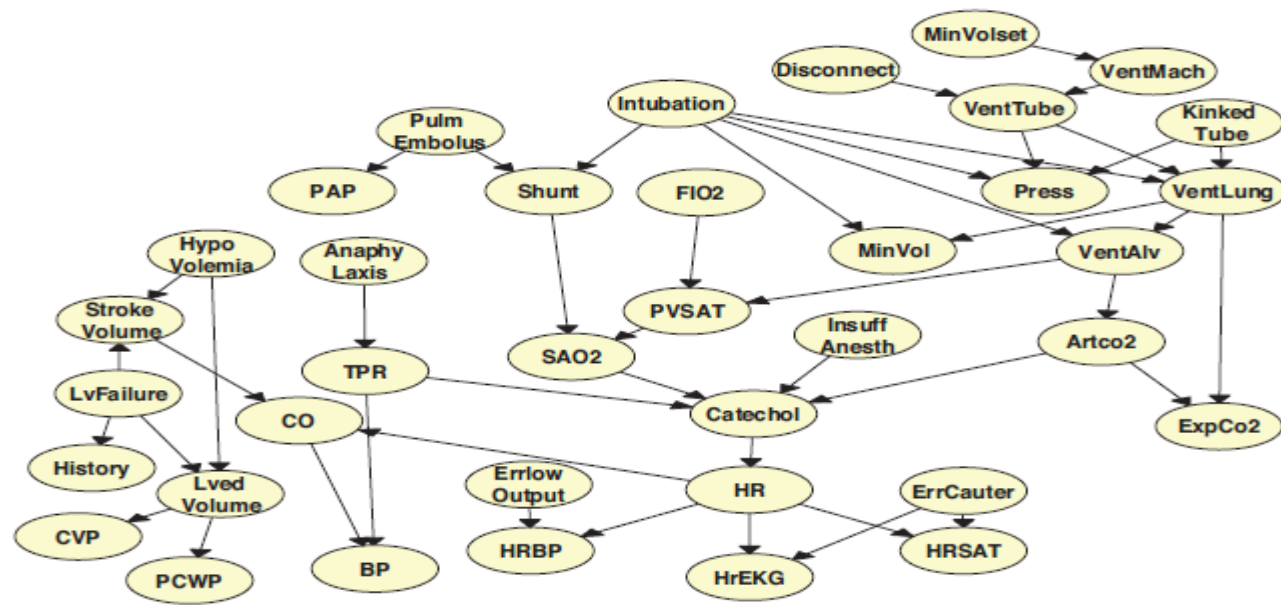
❖ 如果我们希望对重症监护室（ICU）中的病人的疾病进行建模

- 需要测量患者的多种变量，包括呼吸频率、血压等
- 需要描述这些变量之间的关系

❖ Beinlich et al. 1989提出一种“报警网络”

- 它表达了变量间的依赖关系
- 该模型有37个变量和504个参数。

❖ 这样的系统，又称为概率专家系统



“报警网络”说明

❖ “报警网络”是一个复杂的贝叶斯网络，通常用于医疗诊断或监测系统

- 它描述了“临床参数和生理指标”，“病理状态和事件”，“历史和背景信息”及其关系

- 临床参数和生理指标

- ⑩ PAP：肺动脉压，CVP：中心静脉压，PCWP：肺毛细血管楔压，BP：血压，HR：心率，CO：心输出量，等

- 病理状态和事件

- ⑩ Pulm Embolus：肺栓塞，Shunt：分流效应，等

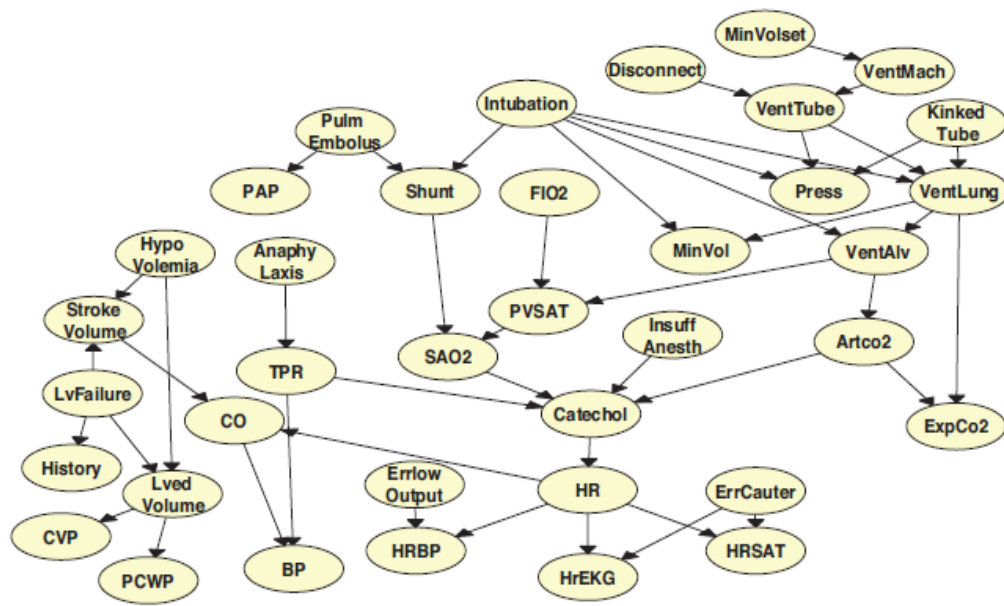
- 历史和背景信息

- ⑩ History：病史，LvFailure：左心衰竭，等

- 关键路径和依赖关系

- ⑩ 从病史到生理指标：

- ✓ 病史影响左心衰竭，进而影响中心静脉压）、肺毛细血管楔压等



另一种医疗诊断网络

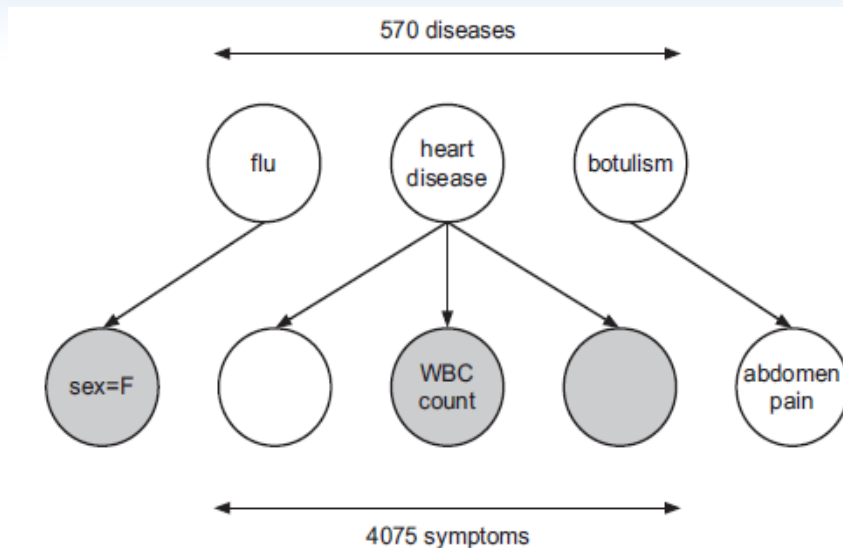
- ❖ Shwe et al. 1991提出，称为快速医学参考（QMR网络）
- ❖ 这是设计的为传染病进行建模
- ❖ QMR模型是二分图结构，上部表示疾病，下部表示症状或发现
 - 所有根节点都服从伯努利分布，代表发生该疾病的先验概率
 - 许多叶节点的父节点数量非常多

⑩ 叶节点（症状）用条件概率表(CPT)表示，需要太多参数

- ❖ 一种替代方案：用逻辑回归表示条件概率分布

$$p(v_t = 1 | h_{pa(t)}) = \text{sigm}(\omega^T h_{pa(t)})$$

- 用有向图表示，称为sigmoid belief net (Neal 1992).



$$p(v, h) = \prod_s p(h_s) \prod_t p(v_t | h_{pa(t)})$$

h_s : 隐藏节点 (疾病diseases)

v_t : 观察节点 (症状symptoms)



推理 (*Inference*)



推理的定义

❖ 概率推理系统的基本任务

- 给定一组证据，计算一组查询变量的后验概率.

❖ 一组完整的变量包括： $\mathbf{X} = \{X\} \cup \mathbf{E} \cup \mathbf{Y}$.

- X 表示查询变量;
- \mathbf{E} 表示一组证据变量 E_1, \dots, E_m ,
- \mathbf{Y} : 即不是证据，也不是查询变量 Y_1, \dots, Y_l (隐变量).

❖ 一个典型的推理任务：计算 $\mathbf{P}(X \mid \mathbf{e})$.

- \mathbf{e} : 特定观测到的证据;

❖ 推理方法分为精确推理和近似推理两类



精确推理方法

- ❖ 贝叶斯网络上的精确推理，从根节点开始计算概率分布，逐步计算到目标变量
- ❖ 枚举推理法
 - 基于联合概率分布的全概率公式，直接计算每个感兴趣的查询变量的所有可能状态的概率值
- ❖ 变量消除法（变量消元法）
 - 逐个消除非查询变量来简化联合概率分布，最终得到查询变量的边际概率。
- ❖ 信念传播：
 - 基于消息传递的思想，在树或聚树上进行高效的推理。
- ❖ 团树（junction tree）算法：
 - 将任意贝叶斯网络转换为一棵团树，然后在其上应用信念传播算法。



近似推理方法

❖ 基于蒙特卡洛采样技术的推理方法：

- 从模型中抽取样本并统计频率来估计概率分布。
- 常用方法
 - ⑩ 前向采样：直接按照网络结构生成样本。
 - ⑩ 拒绝采样：只接受符合证据的样本。
 - ⑩ 重要性采样：加权样本以提高效率。
 - ⑩ 马尔可夫链蒙特卡洛：构建马尔可夫链逐步改进样本质量。

❖ 变分推断：

- 寻找一个简单分布来近似真实的后验分布，使两者之间的差异最小化（如KL散度）



基于枚举的推理

❖ 一个推理 $\mathbf{P}(X \mid \mathbf{e})$ 可以用下式进行计算:

$$p(X \mid \mathbf{e}) = \alpha p(X, \mathbf{e}) = \alpha \sum_y p(X, \mathbf{e}, y)$$

❖ 贝叶斯网络给出了完整的联合概率分布.

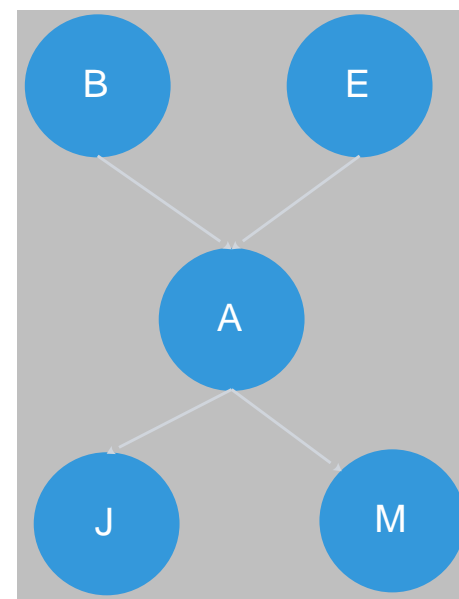
- 推理可以通过贝叶斯网络进行

- ⑩ 计算网络上条件概率乘积的和.



分析一个例子: *Alarm*

- ❖ 场景：当地震发生或者有入室盗窃，报警器会响，若报警器响了， John 和 Mary 会打电话
- ❖ 推理： $P(\text{Burglary} \mid \text{JohnCalls} = \text{true}, \text{MaryCalls} = \text{true})$.
- ❖ 设计网络结构，其中：
 - *B: Burglary*
 - *E: Earthquake*
 - *J: John calls*
 - *M: Mary calls*
 - *A: Alarm*
 - 这个问题里，隐变量是Earthquake和 Alarm



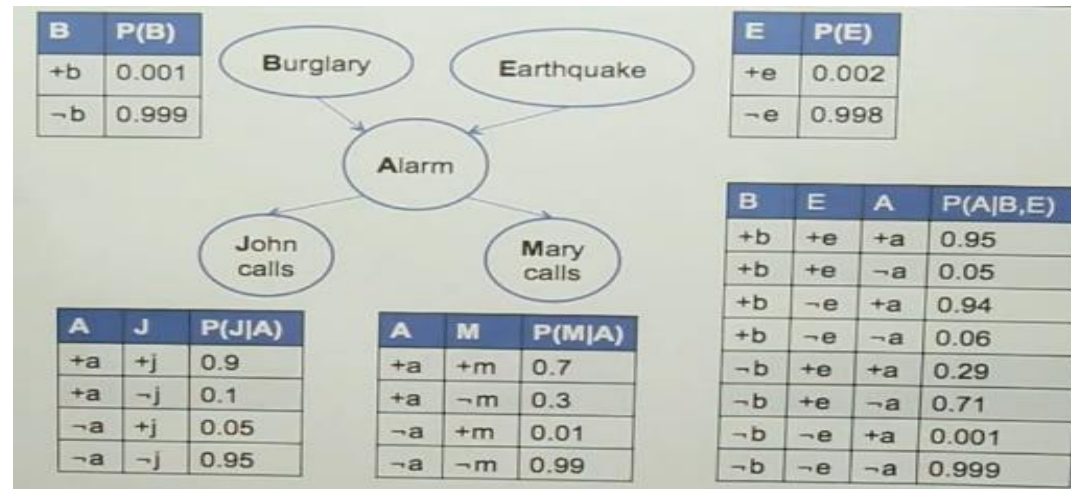
基于枚举直接推理

❖ 根据公式，增加隐变量，有：

$$p(B \mid j, m) = \alpha p(B, j, m) = \alpha \sum_e \sum_a p(B, j, m, e, a)$$

❖ 根据贝叶斯网络，可得到：

$$p(b \mid j, m) = \alpha \sum_e \sum_a p(b)p(e)p(a \mid b, e)p(j \mid a)p(m \mid a)$$



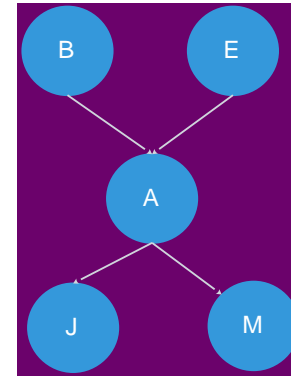
直接枚举推理的复杂度

❖ 贝叶斯网络算法：

$$p(b \mid j, m) = \alpha \sum_e \sum_a p(b)p(e)p(a \mid b, e)p(j \mid a)p(m \mid a)$$

- 仅仅n 个布尔变量的网络，复杂度都是 (2^n)

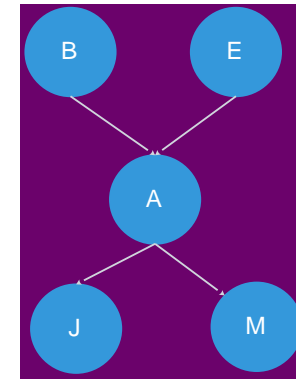
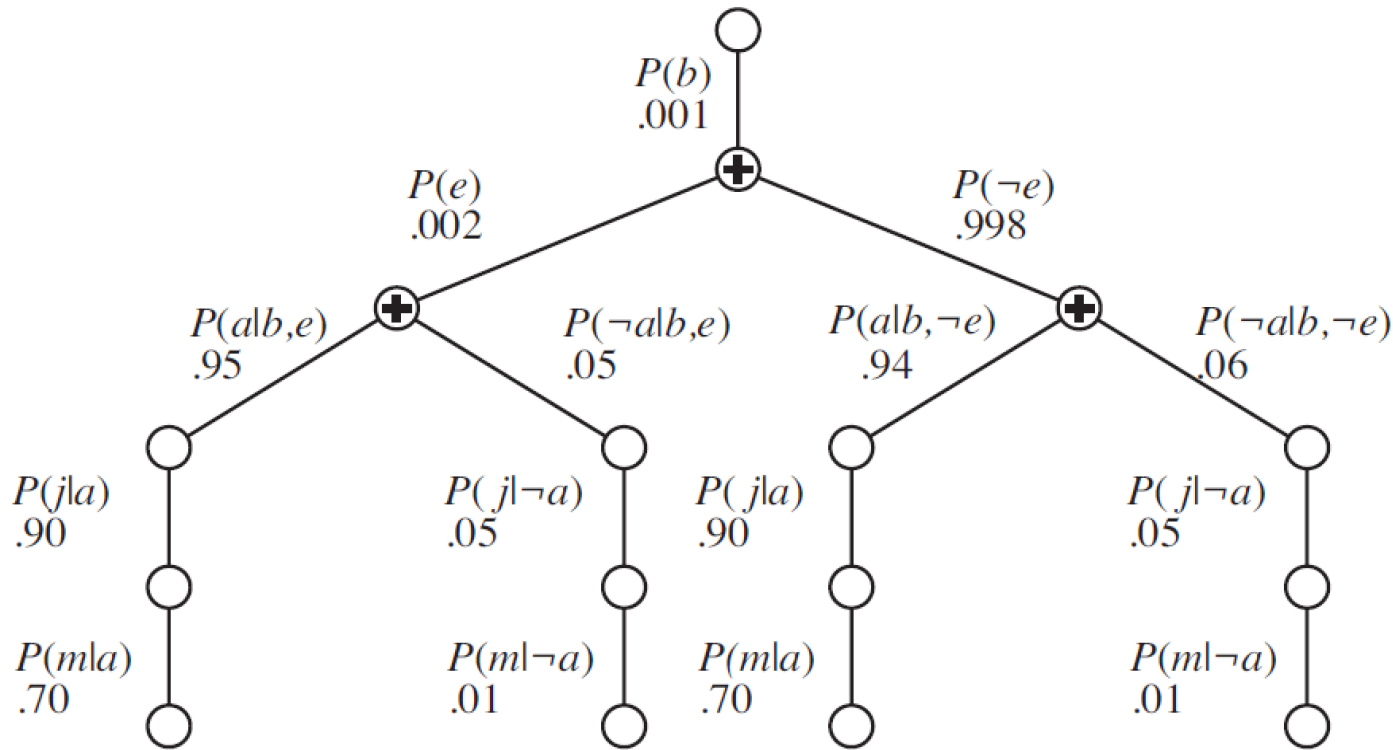
❖ 这种推理方法过于复杂



加速方法：做点简单的改进

- ❖ 一种简单的改进方法：将网络表示成一个树结构

$$p(b \mid j, m) = \alpha p(b) \sum_e p(e) \sum_a p(a \mid b, e) p(j \mid a) p(m \mid a)$$



加速方法: 变量消除法

❖ 变量消除算法: 消除枚举算法中重复变量计算, 改进算法

❖ 想法很简单: 计算一次后保存结果, 后面直接使用。

- 变量消除通过从右到左计算表达式来实现存储中间结果
- 对每个变量的求和仅针对表达式中依赖于该变量的部分。

❖ 例如

$$\begin{aligned} P(d) &= \sum_{a,b,c} P(a, b, c, d) \\ &= \sum_{a,b,c} P(a)P(b|a)P(c|b)P(d|c) = \sum_{b,c} P(c|b)P(d|c) \underbrace{\sum_a P(a)P(b|a)}_{\phi_a(b)} = \sum_c P(d|c) \underbrace{\sum_b P(c|b)\phi_a(b)}_{\phi_b(c)} = \sum_c P(d|c)\phi_b(c) \end{aligned}$$

❖ 存在问题: 不容易找到消解顺序



变量消除法的计算步骤

❖ 初始化:

- 将贝叶斯网络中的每个条件概率表表示为因子（因子分解）
- 对于证据变量 E ，用它们的已知状态，更新相关因子。

❖ 选择消除顺序:

- 变量消除顺序可以影响计算的效率
- 常用的启发式方法:
 - 最小度数（**min-degree**）：选择连接最少其他变量的变量先消除。
 - 最小填充（**min-fill**）：选择消除后引入最少新边的变量先消除。



变量消除法的计算步骤

❖ 消除变量（本质是求边缘概率）：

- 按照选定的顺序逐一消除非查询变量。
- 查找相关因子：找出所有包含要消除的变量的因子
- 乘积运算：将这些因子相乘，生成一个新的因子，这样，相关信息整合到新因子中去了
- 求和运算：对新因子中的要消除变量求和，（这里实际上是在计算边缘概率），消除变量
- 更新因子列表：将含有被消除变量的因子从列表中移除，并添加新的边缘化后的因子到列表中。
- 重复上述步骤，直到只剩下查询变量为止

❖ 归一化：

- 当所有非查询变量都被消除后，剩下的因子只涉及查询变量。
- 对这些因子进行乘积运算，得到查询变量的联合概率分布。
- 对结果归一化，确保其为有效概率分布



因子分解

❖ 如果要计算下列公式:

$$p(b \mid j, m) = \alpha \underbrace{p(B)}_{f_1(B)} \sum_e \underbrace{p(e)}_{f_2(E)} \sum_a \underbrace{p(a \mid b, e)}_{f_3(A,B,E)} \underbrace{p(j \mid a)}_{f_4(A)} \underbrace{p(m \mid a)}_{f_5(A)}$$

- f_i :表示因子, 以矩阵形式存储表达式中某部分的信息

❖ 例如:

- $f_4(A)$ 和 $f_5(A)$ 对应于 $P(j \mid a)$ 和 $P(m \mid a)$, 都依赖于 A

$$f_4(A) = \begin{pmatrix} p(j \mid a) \\ p(j \mid \neg a) \end{pmatrix} = \begin{pmatrix} 0.90 \\ 0.05 \end{pmatrix} \quad f_5(A) = \begin{pmatrix} p(m \mid a) \\ p(m \mid \neg a) \end{pmatrix} = \begin{pmatrix} 0.70 \\ 0.01 \end{pmatrix}$$

- $f_3(A,B,E)$: $2 \times 2 \times 2$ 矩阵, (张量)



因子的逐点乘积

❖ 将表达式：

$$p(b \mid j, m) = \alpha \underbrace{p(B)}_{f_1(B)} \sum_e \underbrace{p(e)}_{f_2(E)} \sum_a \underbrace{p(a \mid b, e)}_{f_3(A, B, E)} \underbrace{p(j \mid a)}_{f_4(A)} \underbrace{p(m \mid a)}_{f_5(A)}$$

❖ 转换成：

$$p(B \mid j, m) = \alpha f_1(B) \times \sum_e f_2(E) \times \sum_a f_3(A, B, E) \times f_4(A) \times f_5(A)$$

- 这里“ \times ”不是普通的矩阵乘法，而是逐点相乘



逐点相乘的定义

- ❖ 因子逐点相乘: $f_1 \times f_2 = f$
 - 假设两个因子有共同变量 Y_1, \dots, Y_k .
 - ✓ $f_1(X_1 \dots X_j, Y_1 \dots Y_k) \times f_2(Y_1 \dots Y_k, Z_1 \dots Z_l) = f(X_1 \dots X_j, Y_1 \dots Y_k, Z_1 \dots Z_l)$
 - ✓ 如果变量都是二值的, 则 f_1 、 f_2 分别有 2^{j+k} 和 2^{k+l} 个项
 - ✓ 逐点乘积有 2^{j+k+l} 个项.
- ❖ 看个两个因子逐点乘积的例子



逐点乘积举例

❖ 给定因子 $f_1(A,B)$ 和 $f_2(B,C)$

- 逐点乘积 $f_1(A,B) \times f_2(B,C) = f_3(A,B,C)$ 有 $2^{1+1+1} = 8$ 项, 如下表

A	B	$f_1(A, B)$	B	C	$f_2(B, C)$	A	B	C	$f_3(A, B, C)$
T	T	0.3	T	T	0.2	T	T	T	$0.3 \times 0.2 = 0.06$
T	F	0.7	T	F	0.8	T	T	F	$0.3 \times 0.8 = 0.24$
F	T	0.9	F	T	0.6	T	F	T	$0.7 \times 0.6 = 0.42$
F	F	0.1	F	F	0.4	T	F	F	$0.7 \times 0.4 = 0.28$
						F	T	T	$0.9 \times 0.2 = 0.18$
						F	T	F	$0.9 \times 0.8 = 0.72$
						F	F	T	$0.1 \times 0.6 = 0.06$
						F	F	F	$0.1 \times 0.4 = 0.04$



变量消除法举例(1)

❖ 对于表达式:

$$p(B \mid j, m) = \alpha f_1(B) \times \sum_e f_2(E) \times \sum_a f_3(A, B, E) \times f_4(A) \times f_5(A)$$

❖ 步骤1:

f_3, f_4, f_5 乘积里对A求和, 得到 2×2 的新因子 $f_6(B, E)$

$$\begin{aligned} f_6(B, E) &= \sum_a f_3(A, B, E) \times f_4(A) \times f_5(A) \\ &= f_3(a, B, E) \times f_4(a) \times f_5(a) + f_3(\neg a, B, E) \times f_4(\neg a) \times f_5(\neg a) \end{aligned}$$

表达式变成:

$$p(B \mid j, m) = \alpha f_1(B) \times \sum_e f_2(E) \times f_6(B, E)$$



变量消除法举例(2)

❖ 对于表达式:

$$p(B \mid j, m) = \alpha f_1(B) \times \sum_e f_2(E) \times f_6(B, E)$$

❖ 步骤2:

f_2, f_6 乘积里对 E 求和, 得到 2×2 的新因子 $f_7(B)$

$$\begin{aligned} f_7(B) &= \sum_e f_2(E) \times f_6(B, E) \\ &= f_2(e) \times f_6(B, e) + f_2(\neg e) \times f_6(B, \neg e) \end{aligned}$$

表达式变成为:

$$p(B \mid j, m) = \alpha f_1(B) \times f_7(B)$$



信念传播法

- ❖ 信念传播是一种在图模型上进行概率推理的算法（特别是在贝叶斯网络）
 - 用于计算网络中各个变量的边缘概率分布，或者最大化后验概率推理。
 - **消息：**消息是从一个节点传递到另一个节点的概率信息
- ❖ 算法步骤
 - **初始化：**为每个节点分配一个初始信念（设置为1）。
 - **消息传递：**节点之间相互传递消息，这些消息代表了节点对其邻居的信念的影响。
 - **消息更新：**节点根据收到的消息更新自己的信念。
 - **收敛：**重复消息传递和更新步骤，直到所有节点的信念收敛（即不再发生变化）。



消息传递

❖ 消息传递过程

- 假设有贝叶斯网络 $A \rightarrow B \rightarrow C$ ，并且已经观测到了 C 的值。假定变量都是二元的：
- 初始化：非观测节点 A 和 B ，初始信念 $P(A)$ 和 $P(B|A)$ ；观测节点 C 发生，信念为1否则为0
- 从 C 向 B 传递消息：
 - ✓ 如果 C 发生，计算 $P(C=\text{true}|B)$ ： $\text{message}(C \rightarrow B) = P(C=\text{True}|B)$
- 从 B 向 A 传递消息：
 - ✓ 整合来自 C 的消息和 B 的条件概率 $P(B|A)$ ： $\text{message}(B \rightarrow A) = \sum_B [P(B|A) * \text{message}(C \rightarrow B)]$



消息更新

- ❖ 当一个节点接收到所有邻居节点的消息后，它可以使用这些消息来更新自己的信念
 - 信念是所有传入消息的乘积，再归一化以确保总和为 1
 - 更新 B 的信念: $\text{belief}(B) \propto P(B | A) \times P(C=\text{True} | B)$
 - 更新 A 的信念: $\text{belief}(A) \propto P(A) \times \sum B [P(B | A) \times P(C=\text{True} | B)]$



学习



学习

- ❖ 推理是指计算 $p(\mathbf{x}_h/\mathbf{x}_v, \boldsymbol{\theta})$,
 - $\boldsymbol{\theta}$ 是模型的参数, 假定已知.

- ❖ 学习通常是指
 - 给定数据后, 计算 **MAP** 来估计参数

$$\hat{\boldsymbol{\theta}} = \arg \max_{\boldsymbol{\theta}} \sum_{i=1}^N \log p(x_{i,v} | \boldsymbol{\theta}) + \log p(\boldsymbol{\theta})$$

- ❖ 如果是均匀先验, $p(\boldsymbol{\theta}) \propto 1$, 后验估计就退化为最大似然估计



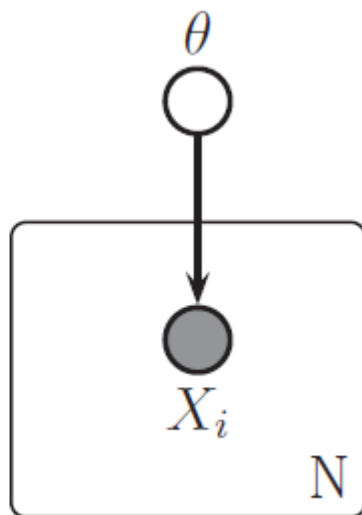
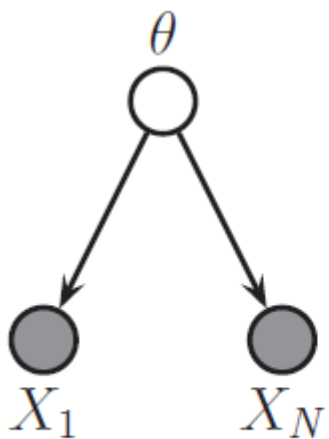
贝叶斯观点

- ❖ 将参数也看成是未知变量进行推理。
- ❖ 推理和学习之间没有区别：
 - 将参数作为节点添加到图中，条件为D推断所有节点的值。
- ❖ 隐变量和参数之间的主要区别
 - 隐变量的数量随着训练数据的数量而增长，
 - 参数的数量通常是固定的



盘标记

- ❖ 当从数据中推断参数时，通常假设数据是独立同分布的（iid）
- ❖ 将这个假设表示为
 - 给定 θ ，数据点 x_i 是条件独立的
 - 这个假定可用盘标记



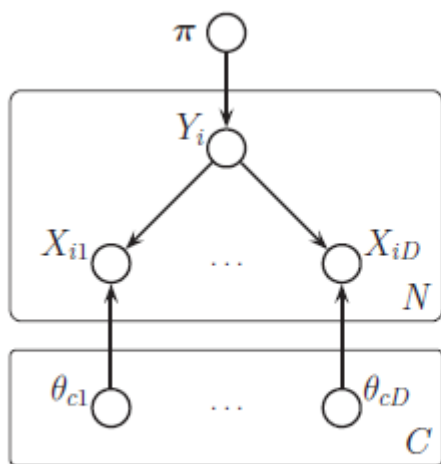
$$p(\theta, D) = p(\theta) \prod_{i=1}^N p(x_i | \theta)$$



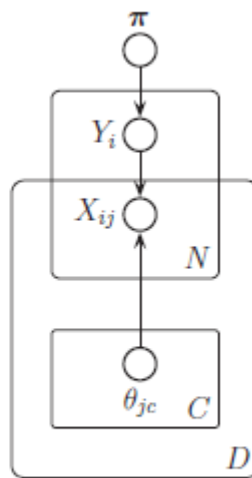
嵌套盘标记

❖ 一个稍微复杂一点的例子：朴素贝叶斯分类器

- (a) 表示已将D个特征“展开”
- (b) 用嵌套盘标记相同的模型



(a)



(b)

$$p(\mathbf{x} | y = c, \boldsymbol{\theta}) = \prod_{j=1}^D p(x_j | y = c, \theta_{jc})$$



完全数据 (*Complete data*)

- ❖ 如果在每种情况下所有的变量都被完全观察到
 - 没有丢失数据，也没有隐藏变量
- ❖ 我们称这数据是完全的



从完全数据中学习

- 当所有变量都已经被观测到时，参数学习就变成了简单的频率计数问题。
- 对于每个条件概率 $P(X|Pa)$ ，计算给定 Pa 的值下 X 发生的频率来估计概率
- 这种情况，最大似然估计是常用的参数估计方法，对于完全数据的有向图模型，其似然函数：

$$p(D | \theta) = \prod_{i=1}^N p(x_i | \theta) = \prod_{i=1}^N \prod_{t=1}^V p(x_{it} | x_{i,pa(t)}, \theta_t) = \prod_{t=1}^V p(D_t | \theta_t)$$

这里 D_t 是与节点 t 相关联的数据以及它们的父节点，

■ 这就是根据图的似然分解.

- 假设先验为：
$$p(\theta) = \prod_{t=1}^V p(\theta_t)$$
- 后验也因子分解为：
$$p(\theta) \propto p(D | \theta)p(\theta) = \prod_{t=1}^V p(D_t | \theta_t)p(\theta_t)$$



DGMs条件独立的特性



图模型的关键

- 任何图模型的关键都是一组条件独立性假设
- $x_A \perp_G x_B | x_C$: A is independent of B given C in the graph G
 - I-map (independence map) : the set of all such CI statements
 - $I(G)$: the set of all such CI statements encoded by the graph
 - $I(p)$ is the set of all CI *statements that hold for distribution p*.
 - *G is an I-map for p, iff $I(G) \subseteq I(p)$*
 - ✓ There are many I-map for p,
 - *$I(G) = I(P)$, Graph G represent the distribution P*
 - ✓ The $I(G)$ is also called P-map (Perfect-map)。



d-separation

- Conditional independence of two variables in graph G
 - determined by using a very important conception called **d-separation**
 - **d-separation : Directed Separation**
- We say path P is **d-separated by a set of nodes E**
 - E contains the evidence
 - iff at least one of the following conditions hold:
 1. P contains a chain, $s \rightarrow m \rightarrow t$ or $s \leftarrow m \leftarrow t$, ($m \in E$)
 2. P contains a tent or fork, $s \swarrow m \searrow t$, where $m \in E$
 3. P contains a **v-structure**, $s \searrow m \swarrow t$, m is not in E and nor is any descendant of m .



d-separated two sets of nodes

- *a set of nodes A is d-separated from a different set of nodes B*
 - *given a third observed set E*
 - *iff each path from every node $a \in A$ to every node $b \in B$ is d-separated by E .*



Why to need d-separate?

- *if the observation variable (evidence node) set E is known*
 - whether node set A and node set B are conditionally independent with respect to E ?
 - ✓ *we can follow all path from A to B . If these path all are d-separated by E*
 - ✓ *it shows that A and B are conditionally independent with respect to E*



global Markov properties

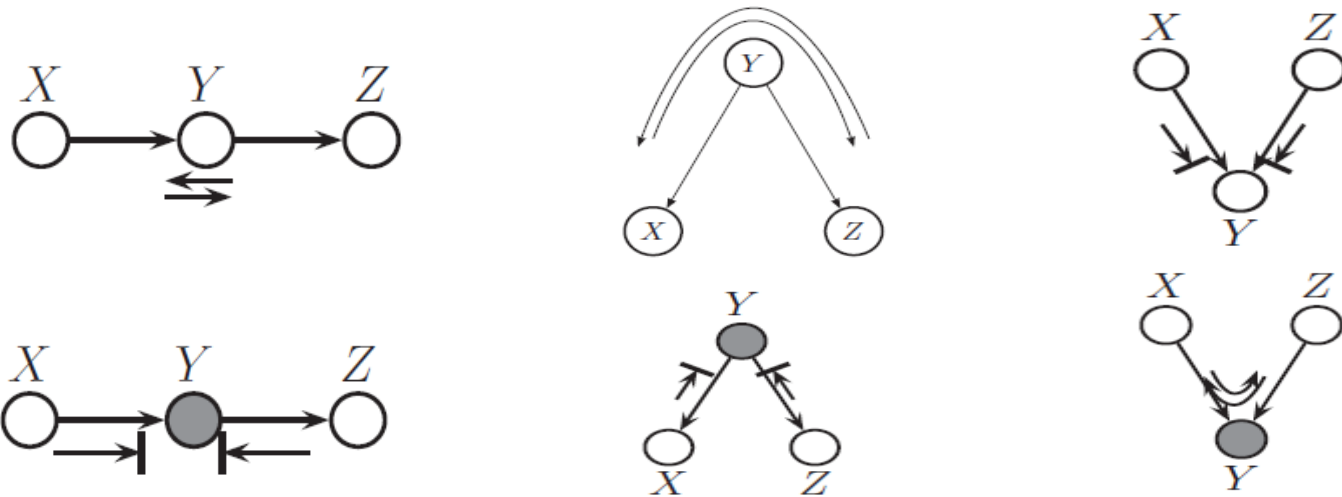
➤ Conditional independence properties of a DGM

■ $\mathbf{x}_A \perp_G \mathbf{x}_B \mid \mathbf{x}_E \iff A \text{ is } d\text{-separated from } B \text{ given } E.$



Bayes ball algorithm

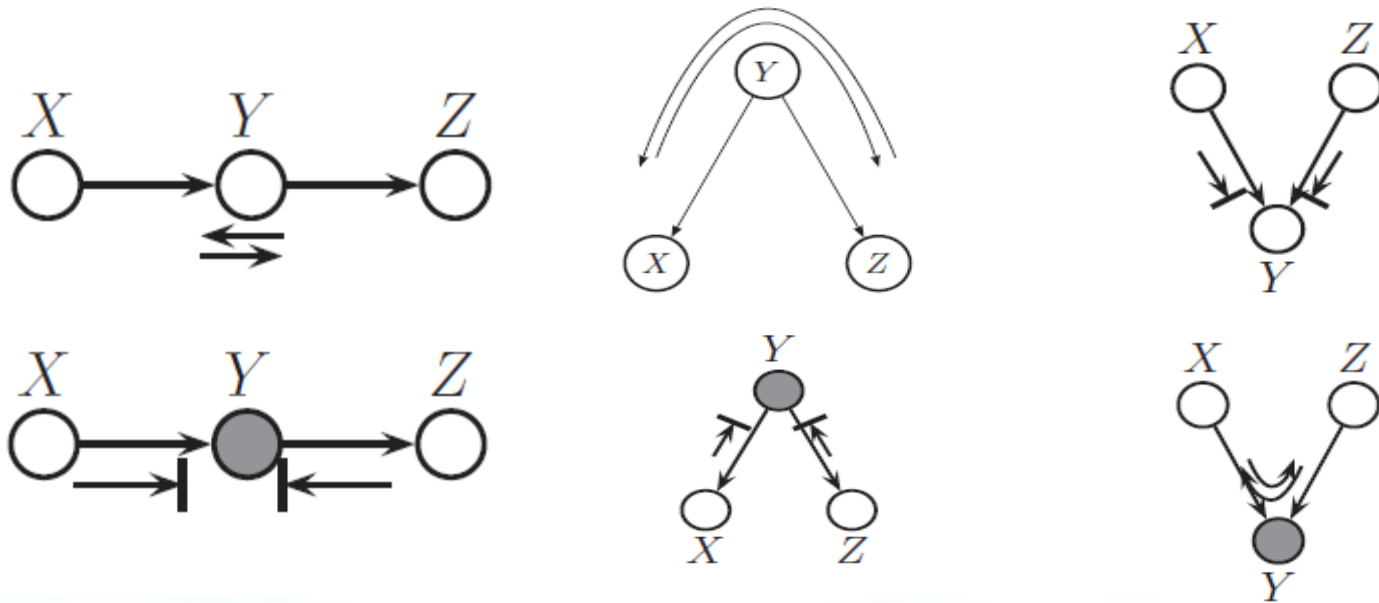
- a simple way to see if A is d -separated from B given E
- The idea of the algorithm:
 - “shade” all nodes in E means they are observed.
 - place “balls” at each node in A , let them “bounce around” according to some rules, ask if any of the balls reach any of the nodes in B .



Rules of Bayes ball

➤ The three main rules

- a ball can pass through a chain, but not if it is shaded in the middle.
- a ball can pass through a fork, but not if it is shaded in the middle.
- a ball cannot pass through a v-structure, unless it is shaded in the middle

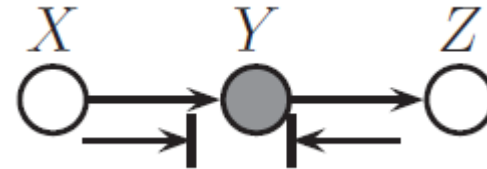
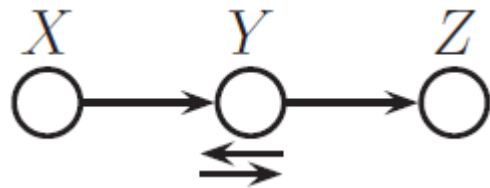


justify Rule 1 of Bayes ball

❖ consider a chain structure $X \rightarrow Y \rightarrow Z$

$$p(x, y, z) = p(x)p(y \mid x)p(z \mid y)$$

$$p(x, z \mid y) = \frac{p(x, z \mid y)p(y)}{p(y)} = \frac{p(x, y, z)}{p(y)} = \frac{p(x)p(y \mid x)p(z \mid y)}{p(y)} = p(x \mid y)p(z \mid y)$$

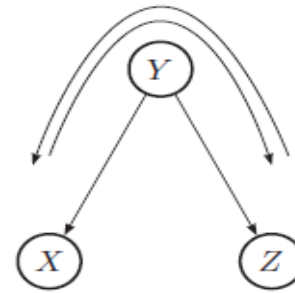
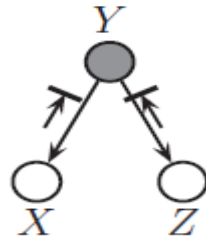


justify Rule 2 of Bayes ball

❖ consider a tent or fork, $X \swarrow Y \searrow Z$

$$p(x, y, z) = p(y)p(x \mid y)p(z \mid y)$$

$$p(x, z \mid y) = \frac{p(x, z \mid y)p(y)}{p(y)} = \frac{p(x, y, z)}{p(y)} = \frac{p(y)p(x \mid y)p(z \mid y)}{p(y)} = p(x \mid y)p(z \mid y)$$



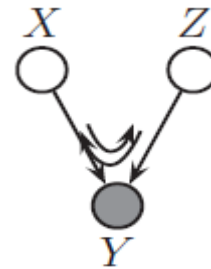
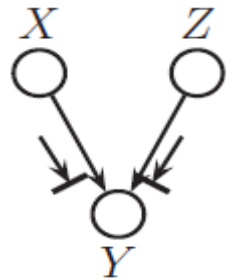
justify Rule 3 of Bayes ball

❖ consider a v-structure, $X \searrow Y \swarrow Z$

$$p(x, y, z) = p(x)p(z)p(y \mid x, z)$$

$$p(x, z \mid y) = \frac{p(x, z \mid y)p(y)}{p(y)} = \frac{p(x, y, z)}{p(y)} = \frac{p(x)p(z)p(y \mid x, z)}{p(y)}$$

❖ So, $X \not\perp Z \mid Y$, *but*, in the unconditional distribution, $X \perp Z$



explaining away

❖ in conditioning on a common child at the bottom of a v-structure

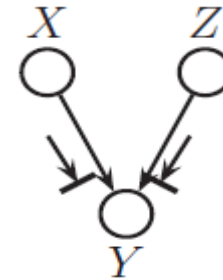
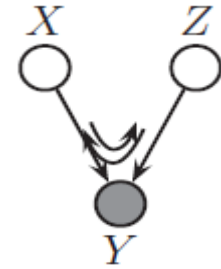
- we see that

- ⑩ x and z are marginally independent.

- ⑩ but it makes its parents become dependent.

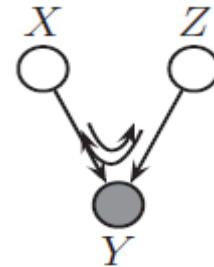
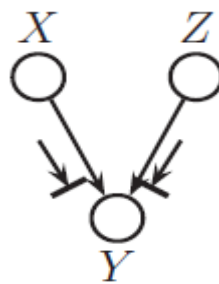
❖ This important effect is called

- **explaining away**
- **inter-causal reasoning**
- **Berkson's paradox**



Example of explaining away

- ❖ suppose we toss two coins (0,1), the coins are independent
 - we observe the “sum” of their values.
 - but once we observe their sum, they become coupled, e.g.
 - ⑩ if the sum is 1, and the first coin is 0,
 - ⑩ then we know the second coin is 1

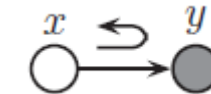
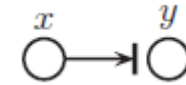


Boundary conditions of Bayes ball

➤ For a v-structure

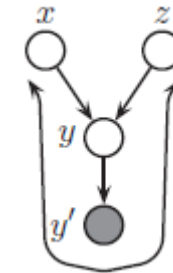
- we need the “boundary conditions of Bayes Ball
- To explain

⑩ besides y , the descendant of Y play the same role as y



➤ Let us analyse it

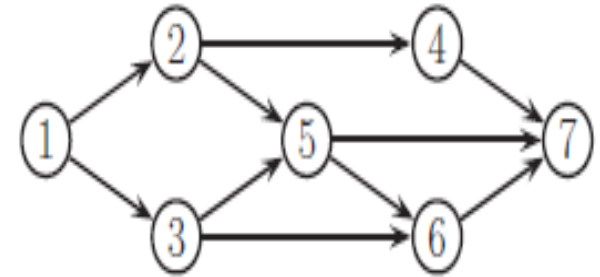
- The Suppose Y' is a noise-free copy of Y .
- if we observe Y' , we effectively observe Y as well,
- so the parents X and Z have to compete to explain this.
- So if we send a ball down $X \rightarrow Y \rightarrow Y'$, it should “bounce back” up along $Y' \rightarrow Y \rightarrow Z$.
- if Y and all its children are hidden, the ball does not bounce back.



directed local Markov property

➤ From the d-separation criterion, one can conclude

- $t \perp \text{nd}(t) \setminus \text{pa}(t) / \text{pa}(t)$
- $\text{nd}(t)$: **non-descendants** of a node t
 - ✓ all the nodes except for its descendants,
 - ✓ $\text{nd}(t) = V \setminus \{t \cup \text{desc}(t)\}$.



➤ For example

- Suppose $t = 3$, then $\text{nd}(3) = \{1, 2, 4\}$, and $\text{pa}(3) = 1$,
- So we have $3 \perp 2, 4 / 1$.



ordered Markov property

❖ A special case of directed local Markov property

- $t \perp \text{pred}(t) \setminus \text{pa}(t) \mid \text{pa}(t)$

- ✓ since $\text{pred}(t) \subseteq \text{nd}(t)$.

- ✓ $\text{pred}(t)$: **predecessors** of a node t

- we only look at predecessors of a node

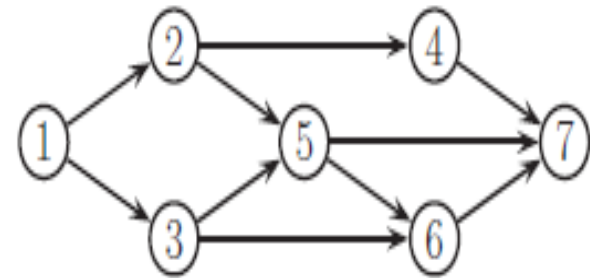
- ✓ according to some topological ordering.

- For example

- ✓ suppose topological ordering $1, 2, \dots, 7$.

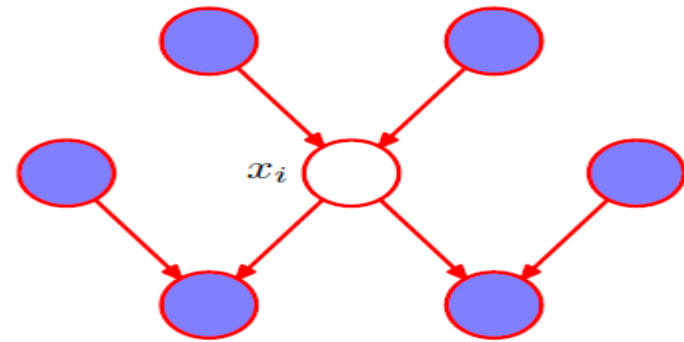
- ✎ $\text{pred}(3) = \{1, 2\}$ and $\text{pa}(3) = 1$

- ✎ So, we have $3 \perp 2/1$



Markov blanket

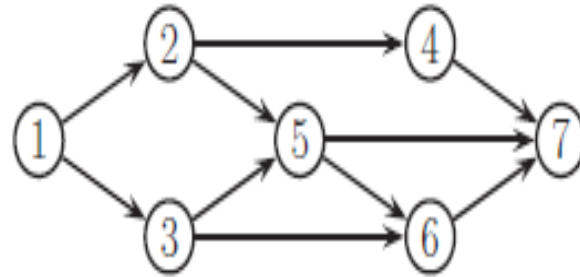
- Markov blanket is the set of nodes
 - it renders t conditionally independent of all the other nodes in the graph
 - It is called **Markov blanket** of t : $mb(t)$
- Markov blanket is equal to
 - the parents + the children + the co-parents
 - $mb(t) = ch(t) \cup pa(t) \cup copa(t)$
- the co-parents:
 - these are also parents of its children



Example of Markov blanket

➤ Given a graph, we have

➤ $mb(5) = \{6, 7\} \cup \{2, 3\} \cup \{4\} = \{2, 3, 4, 6, 7\}$



full conditional

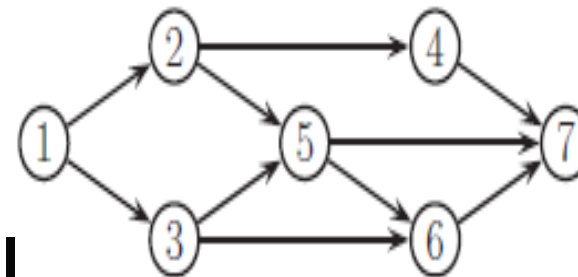
➤ why the co-parents are in the Markov blanket ?

■ Let us see:

$$P(x_t \mid x_{-t}) = \frac{P(x_t, x_{-t})}{P(x_{-t})} = \frac{P(x_t \mid x_{pa(t)})P(x_{ch(t)} \mid x_{pa(ch(t))})P(x_B \mid x_{pa(B)})}{\int P(x_t \mid x_{pa(t)})P(x_{ch(t)} \mid x_{pa(ch(t))})P(x_B \mid x_{pa(B)})dx_t}$$

$$P(x_t \mid x_{-t}) \propto P(x_t \mid x_{pa(t)}) \prod_{s \in ch(t)} P(x_s \mid x_{pa(s)})$$

Where x_{-t} : all the terms that do not involve x_t



➤ The expression is called t's **full conditional**

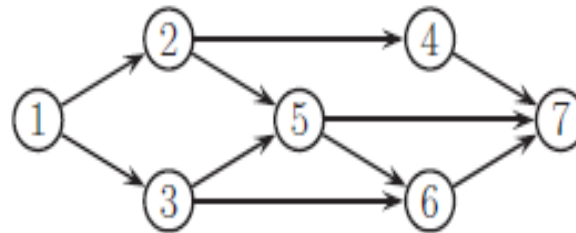
➤ For example

$$p(x_5/x_{-5}) \propto p(x_5/x_2, x_3)p(x_6/x_3, x_5)p(x_7/x_4, x_5, x_6)$$



Example of condition independence

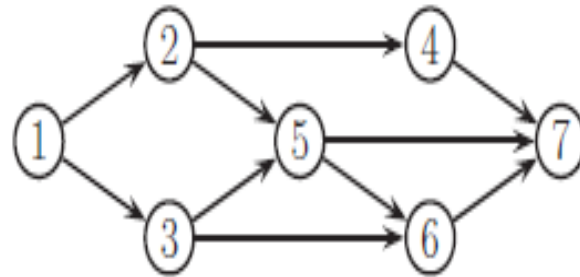
- we see that $x_2 \perp x_6 | x_5$, since
 - the $2 \rightarrow 5 \rightarrow 6$ path is blocked by x_5 (which is observed),
 - the $2 \rightarrow 4 \rightarrow 7 \rightarrow 6$ path is blocked by x_7 (which is hidden),
 - the $2 \rightarrow 1 \rightarrow 3 \rightarrow 6$ path is blocked by x_1 (which is hidden).
- we also see that $x_2 \not\perp x_6 | x_5, x_7$, since
 - the $2 \rightarrow 4 \rightarrow 7 \rightarrow 6$ path is no longer blocked by x_7 (which is observed).



Example of full conditional

➤ Given the graph, we have

$$P(x_5 \mid x_{-5}) \propto P(x_5 \mid x_2, x_3)P(x_6 \mid x_3, x_5)P(x_7 \mid x_4, x_5, x_6)$$





Thank You !