贝叶斯网络

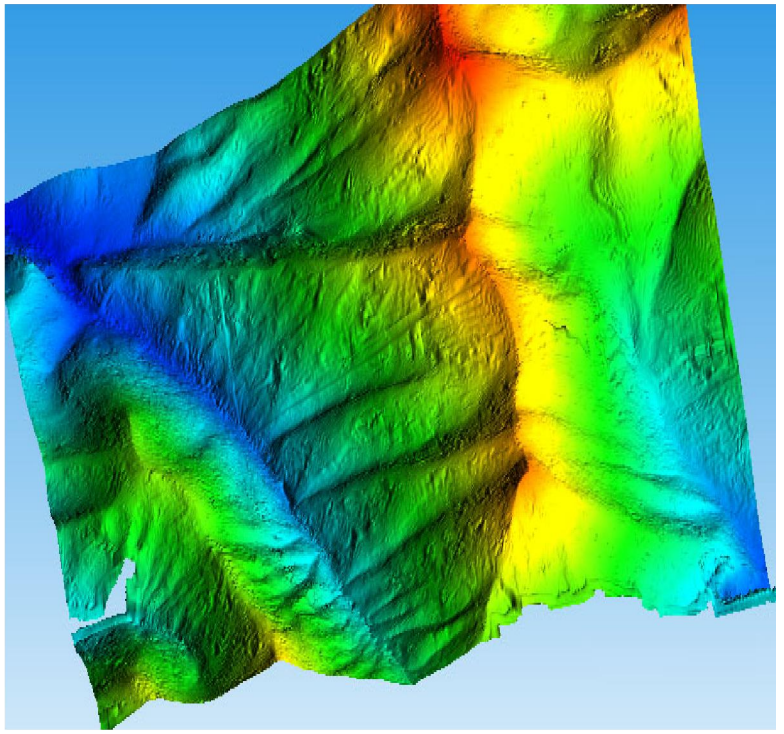
七月算法 邹博

2015年4月12日

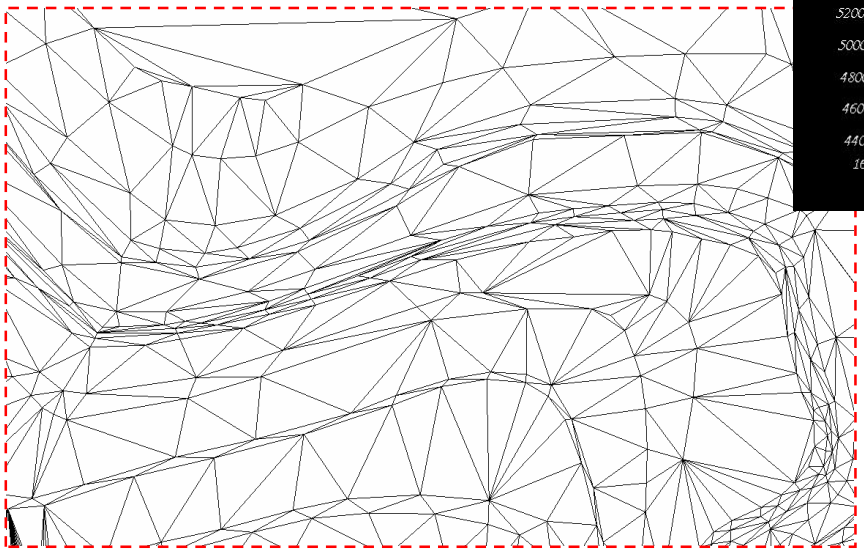
复习：换个角度看对偶

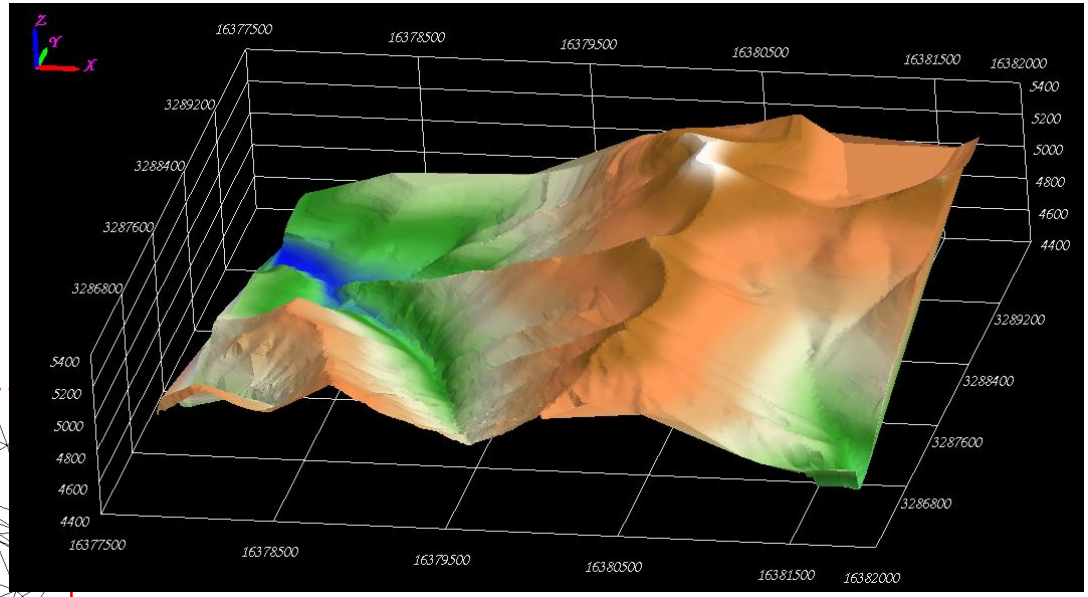
- 给定M个整数和某定值s，要求从M个数中选择若干个数(同一个整数不能多次选择)，使得被选中的数的和为s。输出满足条件的选择数目。
  - 如：从1、2、3、4、5、6、7、8、9中选择若干数，使得它们的和为40。

对偶图：Voronoi图和Delaunay剖分



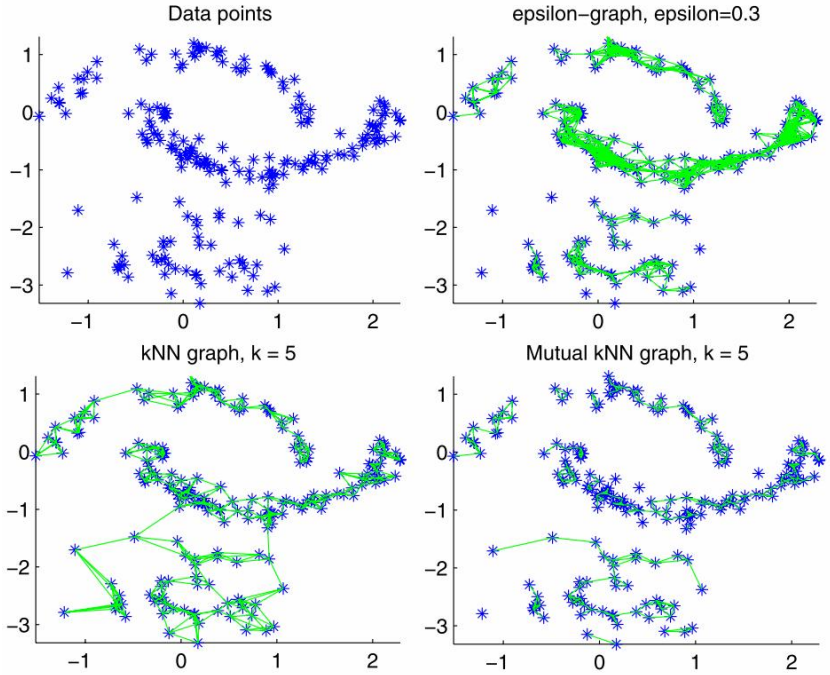
Delaunay三角剖分





K近邻图的有趣结论

- K近邻图中，结点的度至少是K
- K互近邻图中，结点的度至多是K



相对熵

- 相对熵，又称互熵，交叉熵，鉴别信息，Kullback熵，Kullback-Leible散度等
- 设p(x)、q(x)是X中取值的两个概率分布，则p对q的相对熵是

$$D(p\|q)=\sum_x p(x)\log\frac{p(x)}{q(x)}=E_{p(x)}\log\frac{p(x)}{q(x)}$$

- 说明：
  - 相对熵可以度量两个随机变量的“距离”
  - 一般的，D(p||q) ≠D(q||p)
  - D(p||q)≥0、 D(q||p) ≥0
    - 提示：凸函数中的Jensen不等式

相对熵的应用思考

- 假定已知随机变量P，求相对简单的随机变量Q，使得Q尽量接近P
  - 方法：使用P和Q的K-L距离。
  - 难点：K-L距离是非对称的，两个随机变量应该谁在前谁在后呢？
- 假定使用KL(Q||P)，为了让距离最小，则要求在P为0的地方，Q尽量为0。会得到比较“窄”的分布曲线；
- 假定使用KL(P||Q)，为了让距离最小，则要求在P不为0的地方，Q也尽量不为0。会得到比较“宽”的分布曲线；

复习：互信息

- 两个随机变量X，Y的互信息，定义为X，Y的联合分布和独立分布乘积的相对熵。
- I(X,Y)=D(P(X,Y) || P(X)P(Y))

$$I(X,Y)=\sum_{x,y}p(x,y)\log\frac{p(x,y)}{p(x)p(y)}$$

复习：信息增益

- 信息增益表示得知特征A的信息而使得类X的信息的不确定性减少的程度。
- 定义：特征A对训练数据集D的信息增益g(D,A)，定义为集合D的经验熵H(D)与特征A给定条件下D的经验条件熵H(D|A)之差，即：
  - $g(D,A)=H(D) - H(D|A)$
  - 显然，这即为训练数据集D和特征A的互信息。

概率

- 条件概率：

$$P(A|B)=\frac{P(AB)}{P(B)}$$

- 全概率公式：

$$P(A)=\sum_i P(A|B_i)P(B_i)$$

- 贝叶斯(Bayes)公式：

$$P(B_i|A)=\frac{P(A|B_i)P(B_i)}{\sum_j P(A|B_j)P(B_j)}$$

贝叶斯公式的应用

- 8支步枪中有5支已校准过，3支未校准。一名射手用校准过的枪射击，中靶概率为0.8；用未校准的枪射击，中靶概率为0.3；现从8支枪中随机取一支射击，结果中靶。求该枪是已校准过的概率。

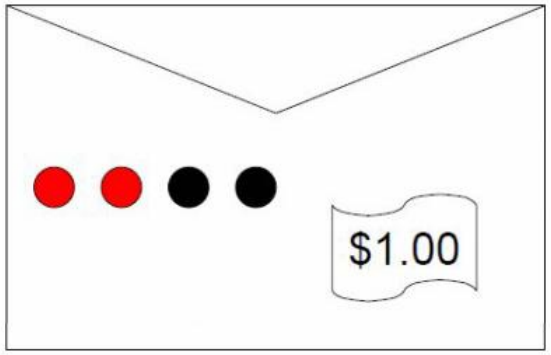
$$P(G=1)=\frac{5}{8} \qquad P(G=0)=\frac{3}{8}$$

- 解：

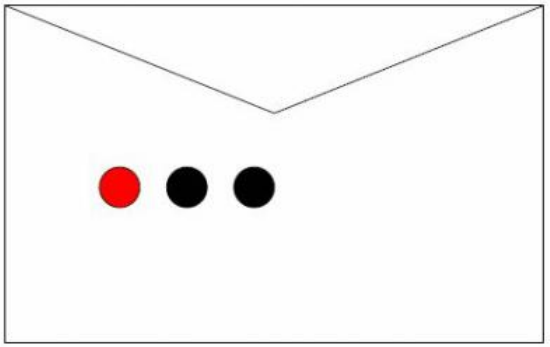
$$P(A=1|G=1)=0.8 \quad P(A=0|G=1)=0.2$$
$$P(A=1|G=0)=0.3 \quad P(A=0|G=0)=0.7$$
$$P(G=1|A=1)=?$$

$$P(G=1|A=1)=\frac{P(A=1|G=1)P(G=1)}{\sum_{i\in G}P(A=1|G=i)P(G=i)}=\frac{0.8\times\frac{5}{8}}{0.8\times\frac{5}{8}+0.3\times\frac{3}{8}}=0.8163$$

一个实例



The “Win” envelope has a dollar and four beads in it



The “Lose” envelope has three beads and no money

Interesting question: before deciding, you are allowed to see one bead drawn from the envelope.

Suppose it’s black: How much should you pay?

Suppose it’s red: How much should you pay?

后验概率

- c1、c2表示左右两个信封。
- P(R)，P(B)表示摸到红球、黑球的概率。
- P(R)=P(R|c1)*P(c1) + P(R|c2)*P(c2)：全概率公式
- P(c1|R)=P(R|c1)*P(c1)/P(R)
  - P(R|c1)=2/4
  - P(R|c2)=1/3
  - P(c1)=P(c2)=1/2
- 如果摸到一个红球，那么，这个信封有1美元的概率是0.6
- 如果摸到一个黑球，那么，这个信封有1美元的概率是3/7

朴素贝叶斯的假设

- 一个特征出现的概率，与其他特征(条件)独立(特征独立性)
  - 其实是：对于给定分类的条件下，特征独立
- 每个特征同等重要(特征均衡性)

以文本分类为例

- 样本：1000封邮件，每个邮件被标记为垃圾邮件或者非垃圾邮件
- 分类目标：给定第1001封邮件，确定它是垃圾邮件还是非垃圾邮件
- 方法：朴素贝叶斯

分析

- 类别c：垃圾邮件c_1，非垃圾邮件c_2
- 词汇表，两种建立方法：
  - 使用现成的单词词典；
  - 将所有邮件中出现的单词都统计出来，得到词典。
  - 记单词数目为N
- 将每个邮件m映射成维度为N的向量**x**
  - 若单词w_i在邮件m中出现过，则$x_i=1$，否则，$x_i=0$。即邮件的向量化：$m \rightarrow (x_1,x_2.....x_N)$
- 贝叶斯公式：$P(c|\mathbf{x})=P(\mathbf{x}|c)*P(c) / P(\mathbf{x})$
  - $P(c_1|\mathbf{x})=P(\mathbf{x}|c_1)*P(c_1) / P(\mathbf{x})$
  - $P(c_2|\mathbf{x})=P(\mathbf{x}|c_2)*P(c_2) / P(\mathbf{x})$
    - 注意这里**x**是向量

分解

- $P(c|\mathbf{x})=P(\mathbf{x}|c)*P(c) / P(\mathbf{x})$
- $P(\mathbf{x}|c)=P(x_1,x_2\ldots x_N|c)=P(x_1|c)*P(x_2|c)\ldots P(x_N|c)$
  - 特征条件独立假设
- $P(\mathbf{x})=P(x_1,x_2\ldots x_N)=P(x_1)*P(x_2)\ldots P(x_N)$
  - 特征独立假设
- 带入公式： $P(c|\mathbf{x})=P(\mathbf{x}|c)*P(c) / P(\mathbf{x})$

- 等式右侧各项的含义：
  - $P(x_i|c_j)$：在c_j(此题目，cj要么为垃圾邮件1，要么为非垃圾邮件2)的前提下，第i个单词x_i出现的概率
  - $P(x_i)$：在所有样本中，单词x_i出现的概率
  - $P(c_j)$：在所有样本中，邮件类别c_j出现的概率

拉普拉斯平滑

- $p(x_1|c_1)$是指的:在垃圾邮件c_1这个类别中，单词x1出现的概率。
  - x_1是待考察的邮件中的某个单词
- 定义符号
  - n_1：在所有垃圾邮件中单词x_1出现的次数。如果x_1没有出现过，则$n_1=0$。
  - n：属于c_1类的所有文档的出现过的单词总数目。
- 得到公式：$$p(x_1|c_1)=\frac{n_1}{n}$$
- 拉普拉斯平滑：$$p(x_1|c_1)=\frac{n_1+1}{n+N}$$
  - 其中，N是所有单词的数目。修正分母是为了保证概率和为1
- 同理，以同样的平滑方案处理$p(x_1)$

对朴素贝叶斯的思考

- 拉普拉斯平滑能够避免0/0带来的算法异常
- 因为要比较的是$P(c_1|\mathbf{x})$和$P(c_2|\mathbf{x})$的相对大小，而根据公式$P(c|\mathbf{x}) = P(\mathbf{x}|c)*P(c) / P(\mathbf{x})$，二者的分母都是除以$P(\mathbf{x})$，实践时可以不计算该系数。
- 编程的限制：小数乘积下溢出怎么办？
- 问题：一个词在样本中出现多次，和一个词在样本中出现一次，形成的词向量相同
  - 由0/1改成计数
- 如何判断两个文档的距离
  - 夹角余弦
- 如何判定该分类器的正确率
  - 样本中：K个生成分类器，1000-K个作为测试集
  - 交叉验证

贝叶斯网络

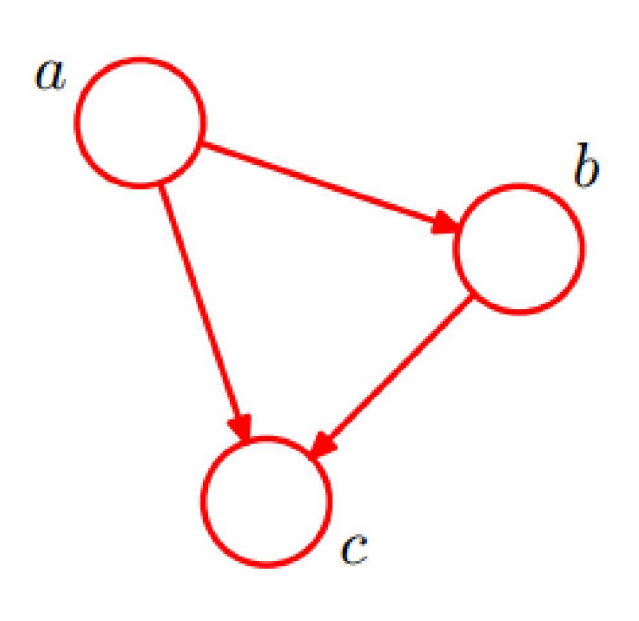
- 把某个研究系统中涉及的随机变量，根据是否条件独立绘制在一个有向图中，就形成了贝叶斯网络。
- 贝叶斯网络(Bayesian Network)，又称有向无环图模型(directed acyclic graphical model ,DAG)，是一种概率图模型，根据概率图的拓扑结构，考察一组随机变量$\{X_1,X_2...X_n\}$及其n组条件概率分布(Conditional Probability Distributions, CPD)的性质。

贝叶斯网络

- 一般而言，贝叶斯网络的有向无环图中的节点表示随机变量，它们可以是可观察到的变量，或隐变量、未知参数等。连接两个节点的箭头代表此两个随机变量是具有因果关系(或非条件独立)。若两个节点间以一个单箭头连接在一起，表示其中一个节点是“因(parents)”，另一个是“果(children)”，两节点就会产生一个条件概率值。
- 每个结点在给定其直接前驱时，条件独立于其非后继。
  - 稍后详细解释此结论

一个简单的贝叶斯网络

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



全连接贝叶斯网络

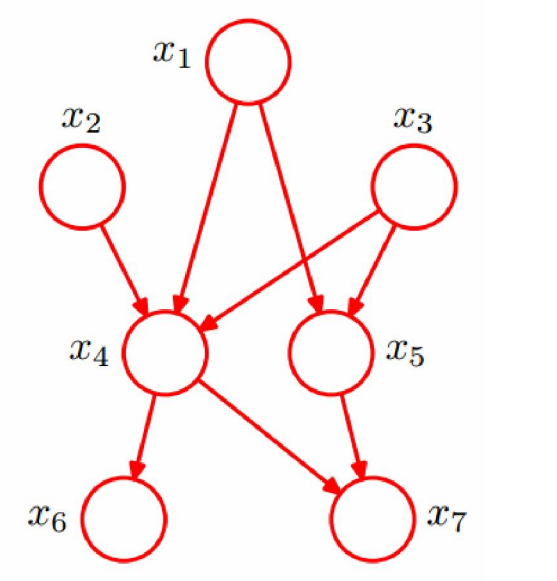
- 每一对结点之间都有边连接

$$p(x_1,\ldots,x_K) = p(x_K|x_1,\ldots,x_{K-1})\ldots p(x_2|x_1)p(x_1)$$

$$\mathrm{P}(X_1 = x_1,\ldots,X_n = x_n) = \prod_{i=1}^{n} \mathrm{P}(X_i = x_i \mid X_{i+1} = x_{i+1},\ldots,X_n = x_n)$$

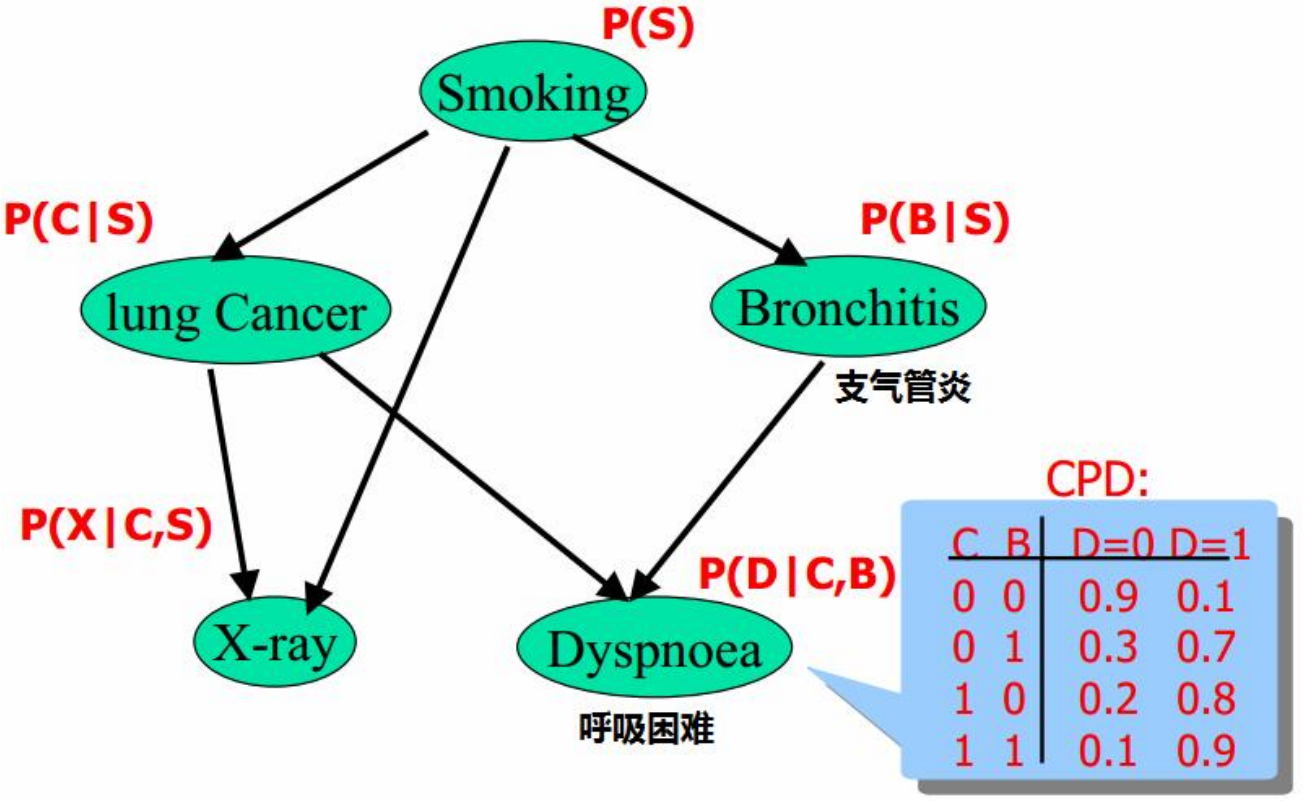
一个“正常”的贝叶斯网络

- 有些边缺失
- 直观上：
  - x1和x2独立
  - x6和x7在x4给定的条件下独立
- x1,x2,...x7的联合分布：



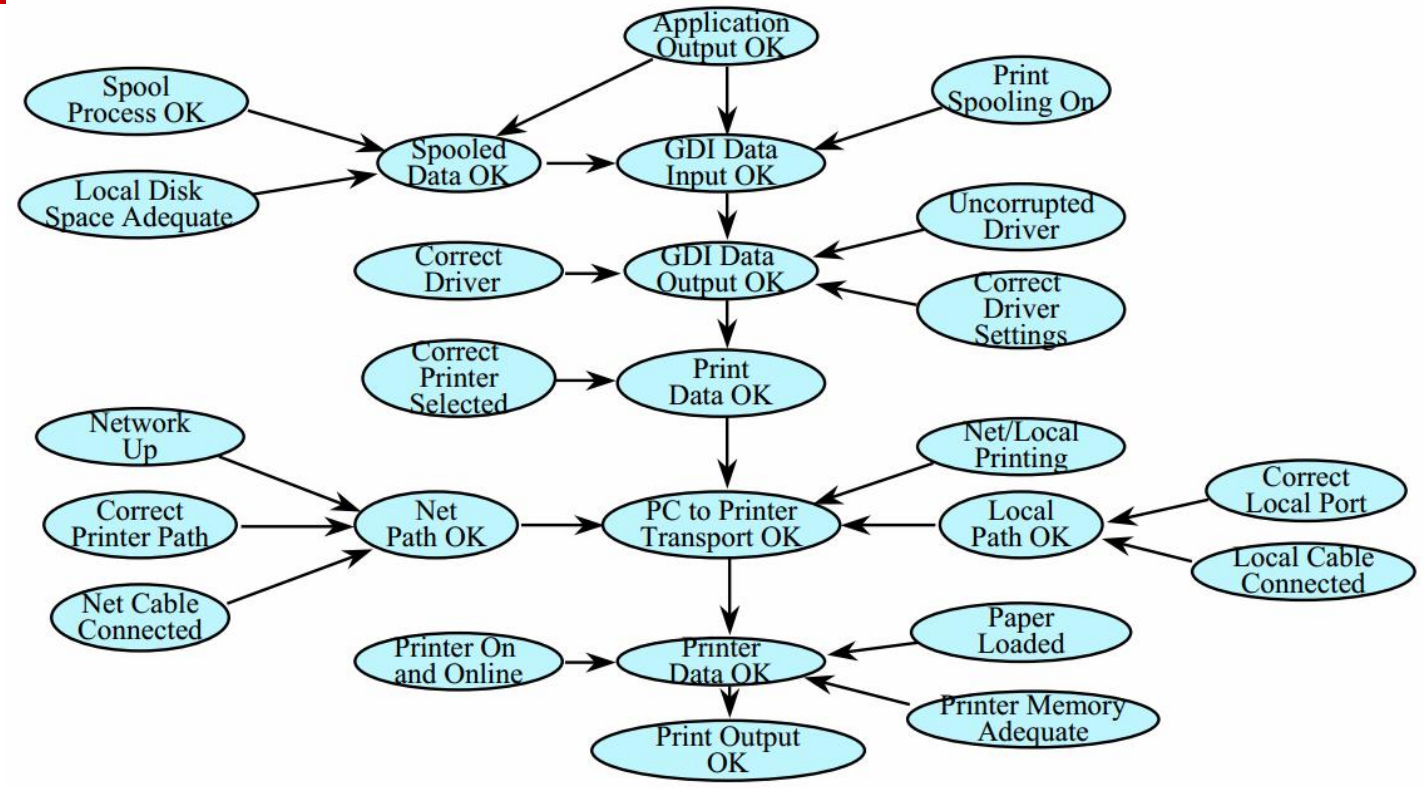
$$p(x_1)p(x_2)p(x_3)p(x_4|x_1,x_2,x_3)p(x_5|x_1,x_3)p(x_6|x_4)p(x_7|x_4,x_5)$$

对一个实际贝叶斯网络的分析



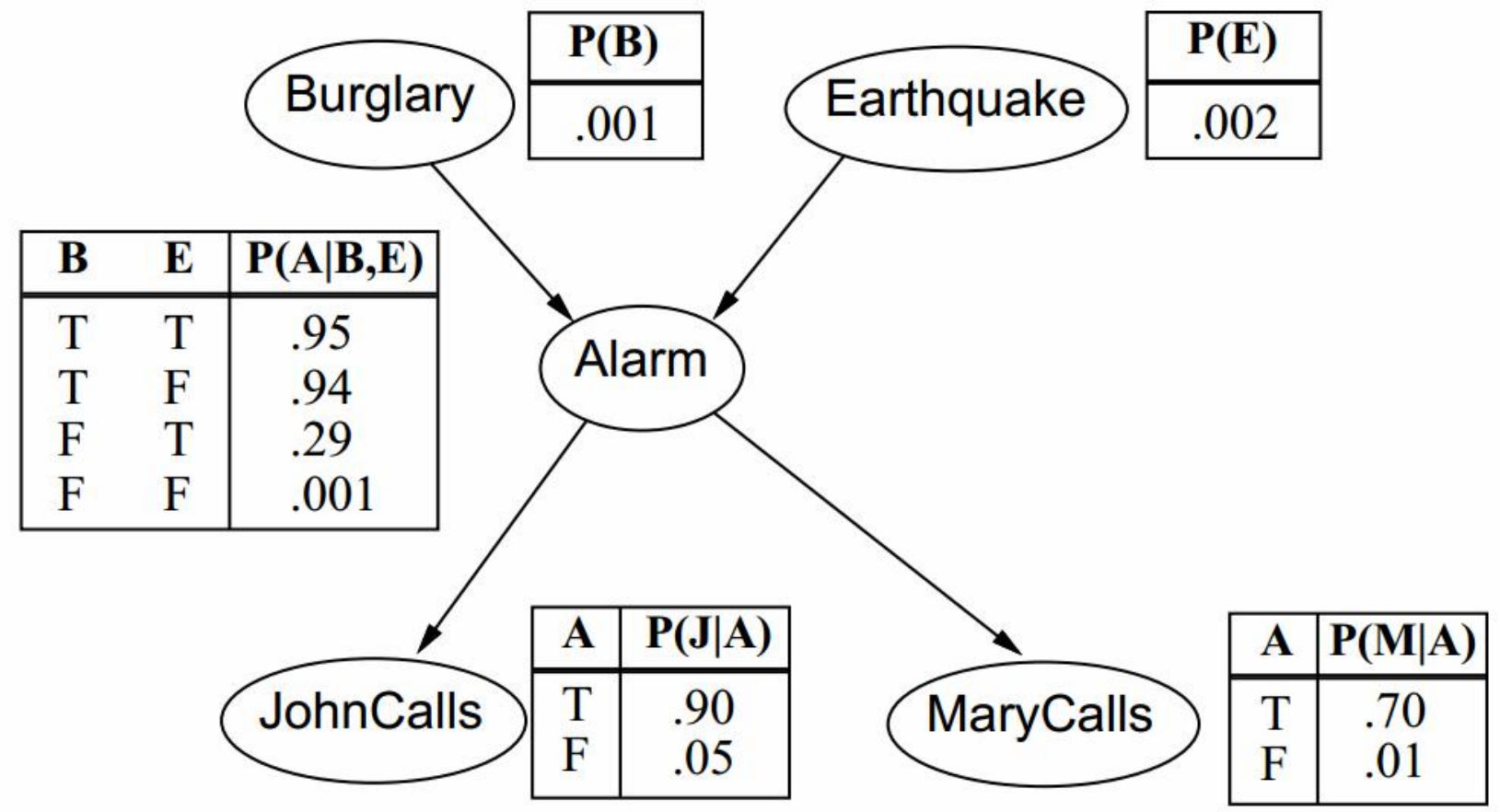
$1+2+2+4+4=13$ vs 2^5

贝叶斯网络：打印机故障诊断



- $17*1 + 1*2 + 2*2^2 + 3*2^3 + 3*2^4 = 99$
- $2^{26} = 67108864$

贝叶斯网络：警报


Burglary
P(B)
.001
Earthquake
P(E)
.002
B E P(A|B,E)
T T .95
T F .94
F T .29
F F .001
Alarm
JohnCalls
A P(J|A)
T .90
F .05
MaryCalls
A P(M|A)
T .70
F .01

贝叶斯网络：警报

- 全部随机变量的联合分布

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

$$P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$$

$$= P(j|a)P(m|a)P(a|\neg b, \neg e)P(\neg b)P(\neg e)$$
$$= 0.9 \times 0.7 \times 0.001 \times 0.999 \times 0.998$$
$$\approx 0.00063$$

贝叶斯网络的形式化定义

- BN(G, Θ)
  - G:有向无环图
  - G的结点：随机变量
  - G的边：结点间的有向依赖
  - Θ：所有条件概率分布的参数集合
  - 结点X的条件概率：P(X|parent(X))

$$P(S, C, B, X, D) = P(S)\,P(C|S)\,P(B|S)\,P(X|C,S)\,P(D|C,B)$$

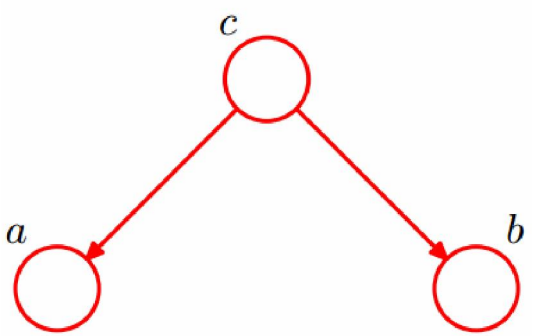
- 思考：需要多少参数才能确定上述网络呢？
  - 每个结点所需参数的个数：结点的parent数目是M，结点和parent的可取值数目都是K：$K^M*(K-1)$
  - 为什么？
  - 考察结点的parent对该结点形成了多少种情况(条件分布)

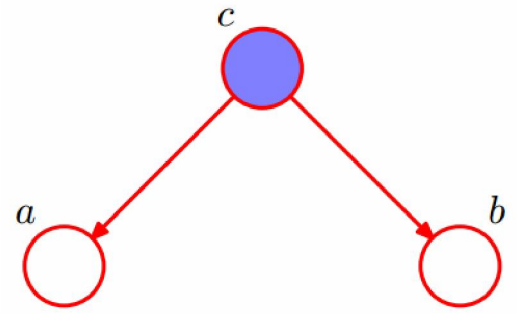
特殊的贝叶斯网络

- M个离散结点形成一条链，每一个结点有K个状态，则需要K-1+(M-1)K(K-1)个参数。这是关于长度M的线性函数。
  - 别忘了，如果是全连接，需要K^M-1个参数，是关于M的指数函数。
- 这个网络被称作马尔科夫模型。

通过贝叶斯网络判定条件独立—1

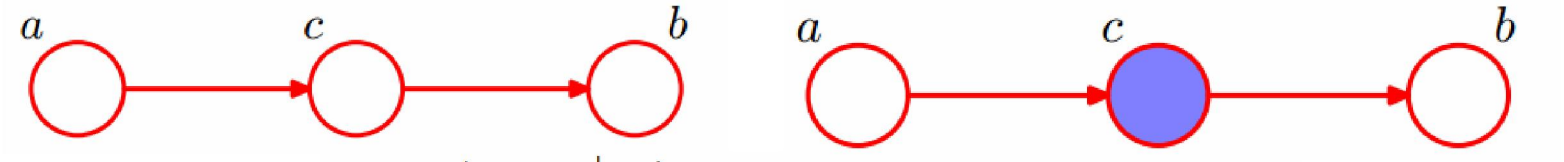
- P(a,b,c)=P(c)*P(a|c)*P(b|c)
- 则：P(a,b|c)=P(a,b,c)/P(c)
- 带入，得到：
- P(a,b|c)=P(a|c)*P(b|c)
- 即：在c给定的条件下，a，b被阻断(blocked)，是独立的。
  - 条件独立：tail-to-tail





通过贝叶斯网络判定条件独立—2

- P(a,b,c)=P(a)*P(c|a)*P(b|c)



$$
\begin{aligned}
& P(a, b \mid c) \\
= \ & P(a, b, c) / P(c) \\
= \ & P(a) * P(c \mid a) * P(b \mid c) \ / \ P(c) \\
= \ & P(a, c) * \ P(b \mid c) \ / \ P(c) \\
= \ & P(a \mid c) \ * \ P(b \mid c)
\end{aligned}
$$

- 即：在c给定的条件下，a，b被阻断(blocked)，是独立的。
  - 条件独立：head-to-tail

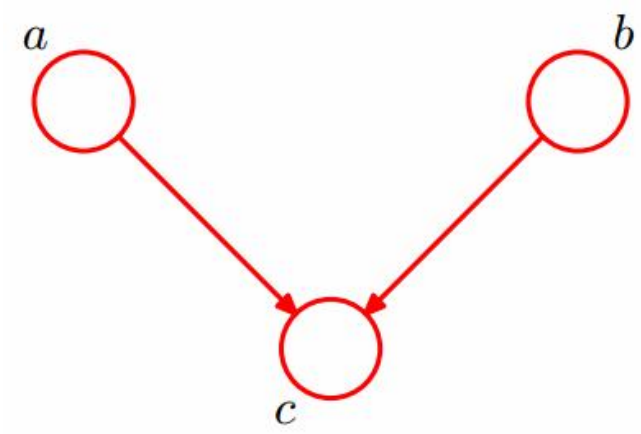
通过贝叶斯网络判定条件独立—3

- $P(a,b,c) = P(a)*P(b)*P(c|a,b)$

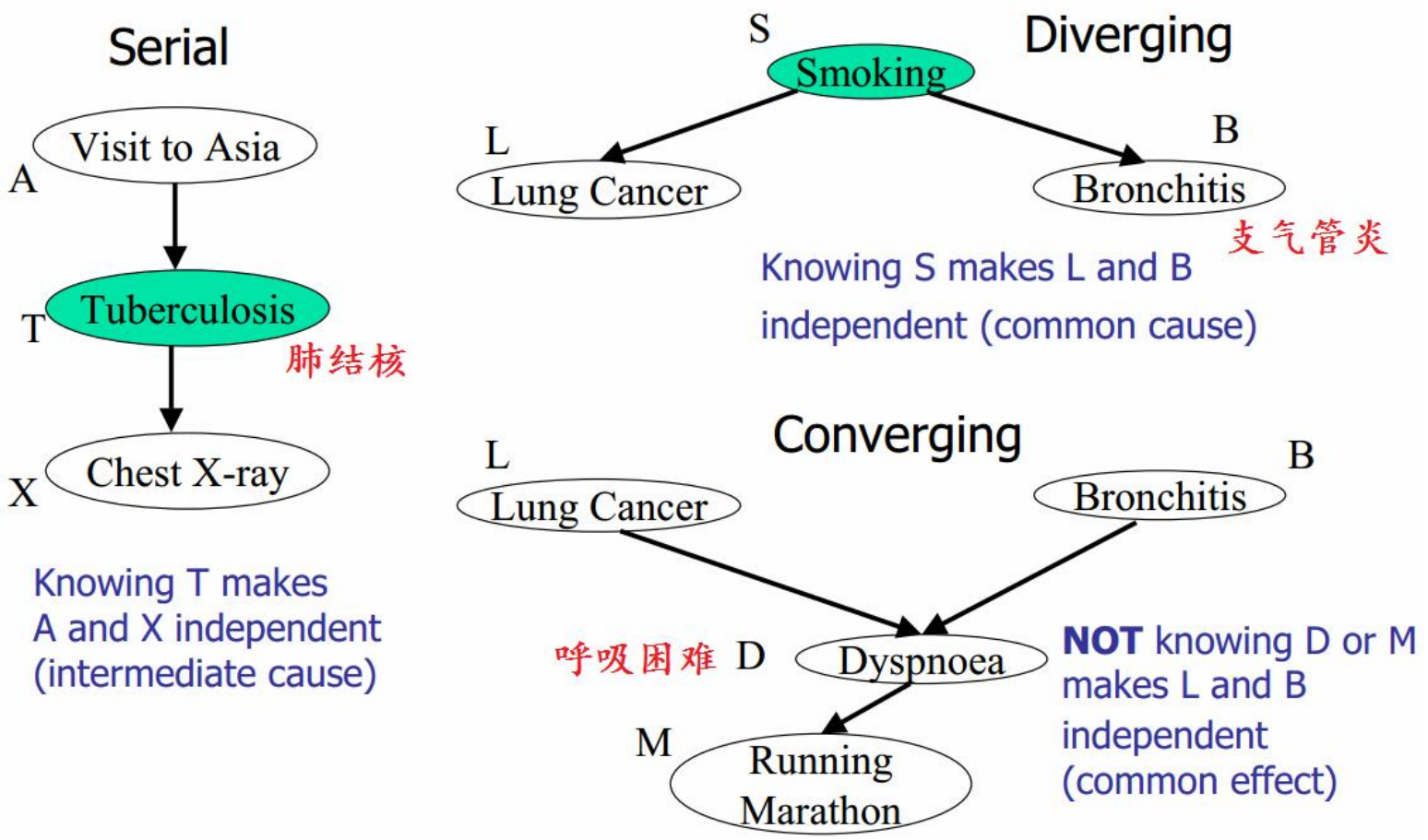
$$\sum_c P(a,b,c) = \sum_c P(a)*P(b)*P(c \mid a,b)$$

$$\Rightarrow P(a,b) = P(a)*P(b)$$

- 在c未知的条件下，a，b被阻断(blocked)，是独立的： head-to-head



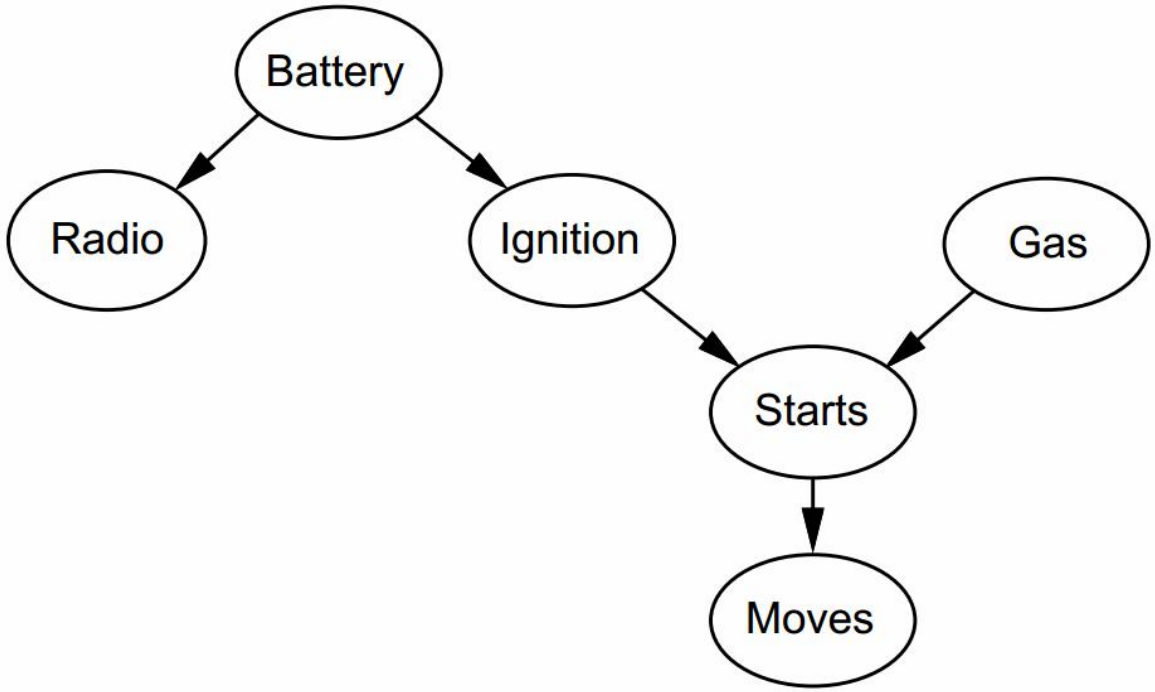
举例说明这三种情况


Serial
A
Visit to Asia
T
Tuberculosis
肺结核
X
Chest X-ray
Knowing T makes
A and X independent
(intermediate cause)
S
Smoking
Diverging
L
Lung Cancer
B
Bronchitis
支气管炎
Knowing S makes L and B
independent (common cause)
Converging
L
Lung Cancer
B
Bronchitis
呼吸困难 D
Dyspnoea
M
Running
Marathon
NOT knowing D or M
makes L and B
independent
(common effect)

将上述结点推广到结点集

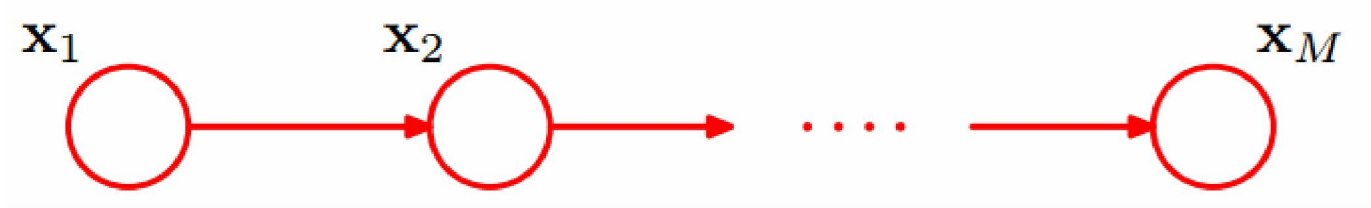
- D-separation：有向分离
- 对于任意的结点集A，B，C，考察所有通过A中任意结点到B中任意结点的路径，若要求A，B条件独立，则需要所有的路径都被阻断(blocked)，即满足下列两个前提之一：
  - A和B的“head-to-tail型”和“tail-to-tail型”路径都通过C；
  - A和B的“head-to-head型”路径不通过C以及C的子孙；
- 如果A,B不满足D-separation，A,B有时被称为D-connected.

有向分离的举例



- Gas和Radio是独立的吗？给定Battery呢？Ignition呢？Starts呢？Moves呢？(答：IIDD)

再次分析链式网络



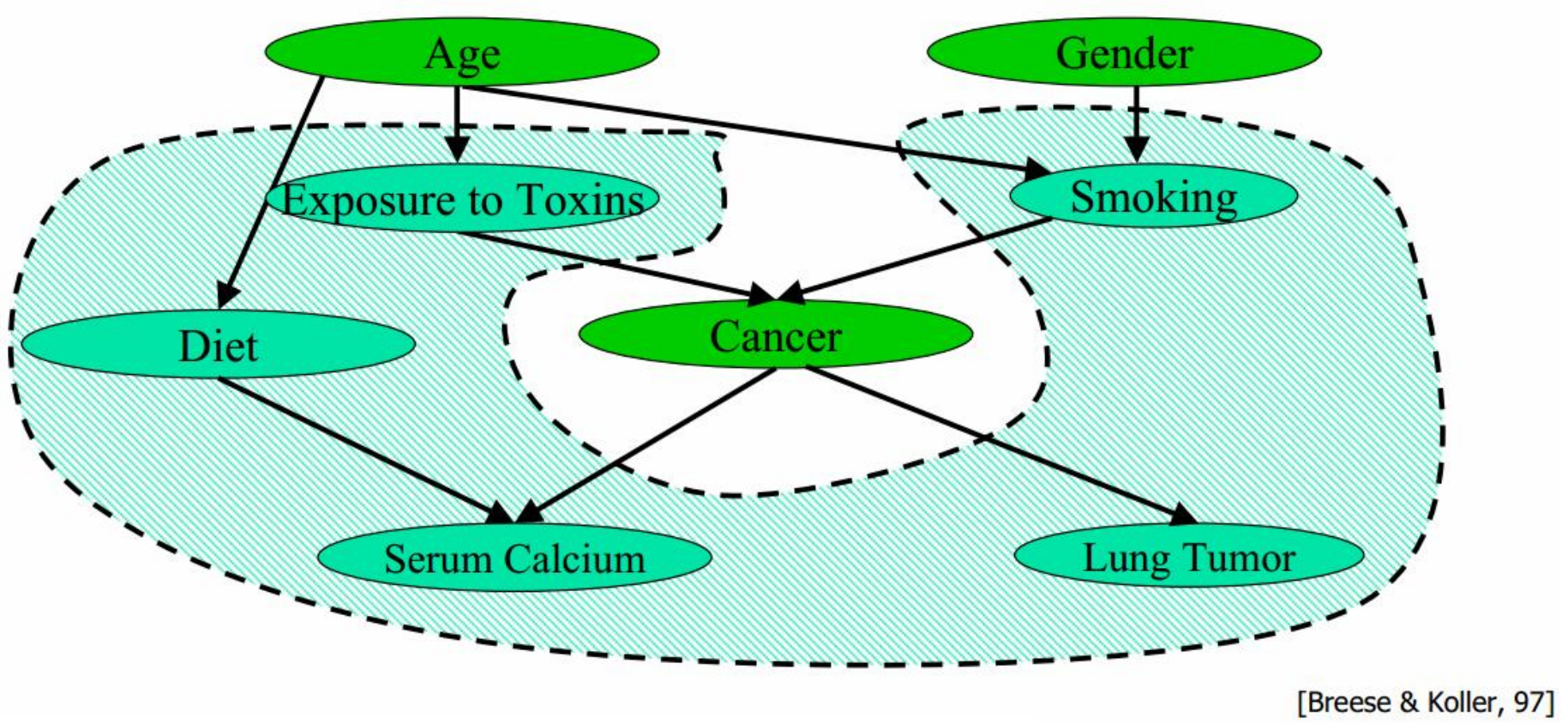
- 有D-separation可知，在xi给定的条件下，xi+1的分布和x1,x2...xi-1条件独立。即：xi+1的分布状态只和xi有关，和其他变量条件独立，这种顺次演变的随机过程模型，叫做马尔科夫模型。

$$P(X_{n+1} = x|X_0, X_1, X_2, \ldots, X_n) = P(X_{n+1} = x|X_n)$$

Markov Blanket

- 一个结点的Markov Blanket是一个集合，在这个集合中的结点都给定的条件下，该结点条件独立于其他所有结点。
- 即：一个结点的Markov Blanket是它的parents,children以及spouses(孩子的其他parent)

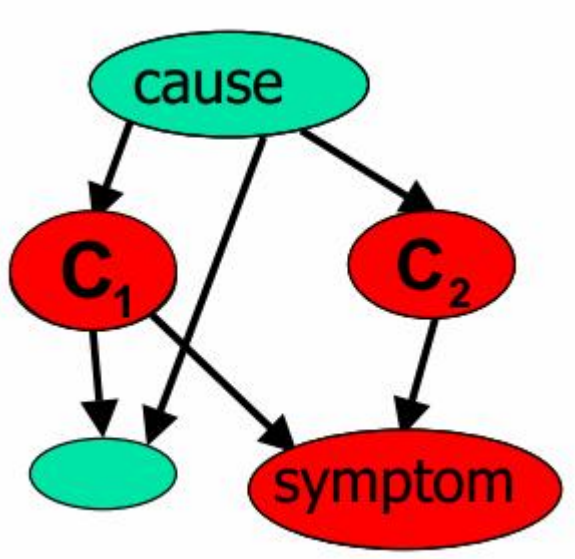
Markov Blanket



[Breese & Koller, 97]

补充知识：Serum Calcium(血清钙浓度)高于2.75mmol/L即为高钙血症。许多恶性肿瘤可并发高钙血症。以乳腺癌、骨肿瘤、肺癌、胃癌、卵巢癌、多发性骨髓瘤、急性淋巴细胞白血病等较为多见，其中乳腺癌约1/3 可发生高钙血症。

贝叶斯网络的用途



- 诊断：P(病因|症状)
- 预测：P(症状|病因)
- 分类：$\max_{class} P(类别|数据)$

- 通过给定的样本数据，建立贝叶斯网络的拓扑结构和结点的条件概率分布参数。这往往需要借助先验知识和极大似然估计来完成。
- 在贝叶斯网络确定的结点拓扑结构和条件概率分布的前提下，可以使用该网络，对未知数据计算条件概率或后验概率，从而达到诊断、预测或者分类的目的。

应用实例

APRI system developed at AT&T Bell Labs

learns & uses Bayesian networks from data to identify customers liable to default on bill payments

NASA Vista system

predict failures in propulsion systems

considers time criticality & suggests highest utility action

dynamically decide what information to show

- 由AT&T贝尔实验室开发的APRI系统
  - 从数据中学习和使用贝叶斯网络，用来识别那些有赖账倾向的客户
- NASA vista系统
  - 预测推进系统的失败率
  - 分析更精确的时间窗口，提供高可靠度的行动
  - 动态决定显示哪些信息

贝叶斯网络的构建

- 依次计算每个变量的D-separation的局部测试结果，综合每个结点得到贝叶斯网络。
- 算法过程：
  - 选择变量的一个合理顺序：$X_1,X_2,...X_n$
  - 对于i=1到n
    - 在网络中添加X_i结点
    - 在$X_1,X_2,...X_{i-1}$中选择X_i的父母，使得：

$$P(X_i|Parent(X_i))=P(X_i|X_1,X_2\cdots X_{i-1})$$

- 这种构造方法，显然保证了全局的语义要求：

$$P(X_1,X_2\cdots X_n)=\prod_{i=1}^{n}P(X_i|X_1,X_2\cdots X_{i-1})=\prod_{i=1}^{n}P(X_i|Parent(X_i))$$

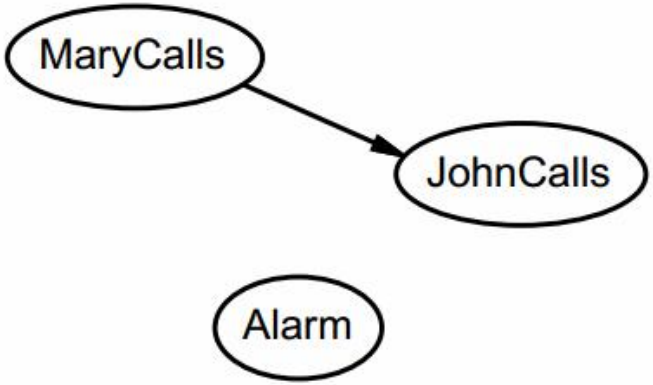
贝叶斯网络的构建举例

Suppose we choose the ordering M, J, A, B, E

$P(J|M) = P(J)$?

构建举例

Suppose we choose the ordering M, J, A, B, E

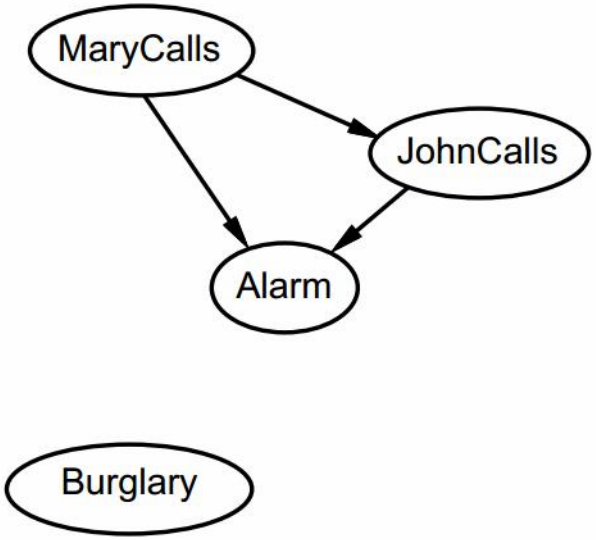


$P(J|M) = P(J)$? No

$P(A|J, M) = P(A|J)$? $P(A|J, M) = P(A)$?

构建举例

Suppose we choose the ordering M, J, A, B, E



$P(J|M) = P(J)$? No

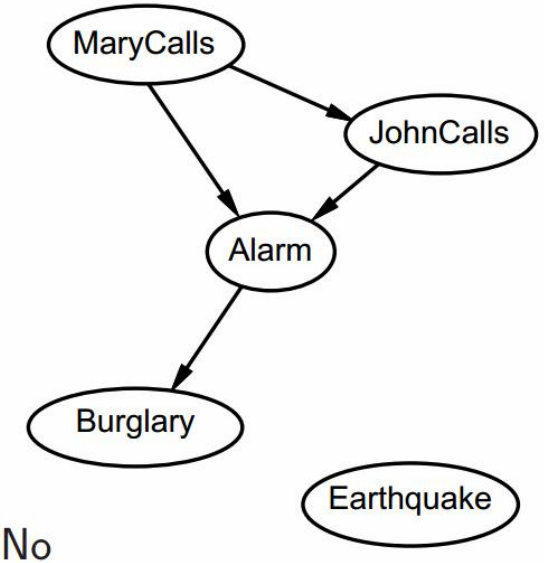
$P(A|J, M) = P(A|J)$? $P(A|J, M) = P(A)$? No

$P(B|A, J, M) = P(B|A)$?

$P(B|A, J, M) = P(B)$?

构建举例

Suppose we choose the ordering M, J, A, B, E



$P(J|M) = P(J)$? No

$P(A|J, M) = P(A|J)$? $P(A|J, M) = P(A)$? No

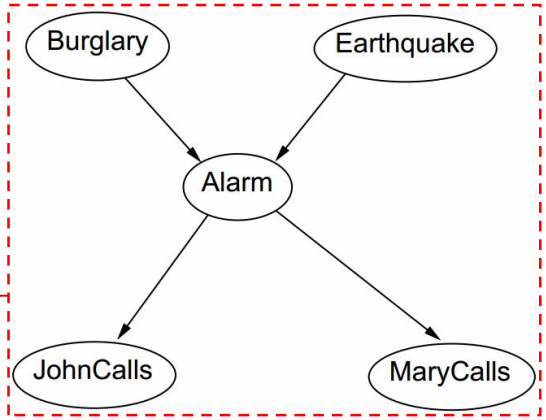
$P(B|A, J, M) = P(B|A)$? Yes

$P(B|A, J, M) = P(B)$? No

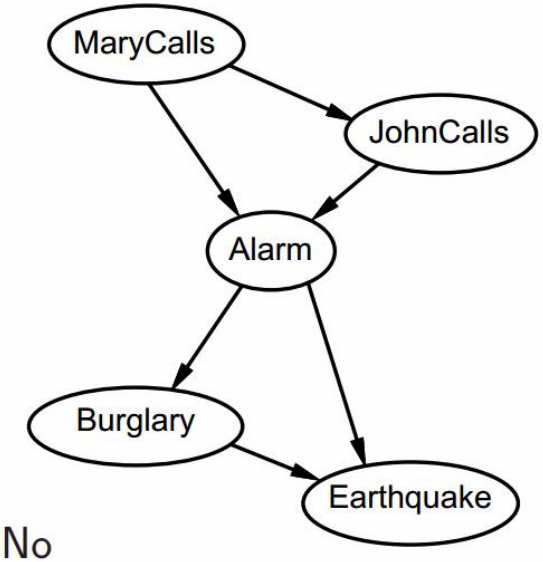
$P(E|B, A, J, M) = P(E|A)$?

$P(E|B, A, J, M) = P(E|A, B)$?

构建举例



Suppose we choose the ordering M, J, A, B, E



$P(J|M) = P(J)$? No

$P(A|J, M) = P(A|J)$? $P(A|J, M) = P(A)$? No

$P(B|A, J, M) = P(B|A)$? Yes

$P(B|A, J, M) = P(B)$? No

$P(E|B, A, J, M) = P(E|A)$? No

$P(E|B, A, J, M) = P(E|A, B)$? Yes

1+2+4+2+4=13

压缩条件分布参数数目

- Noisy-OR分布模型
- 节点$U_1,U_2,...U_k$是X的所有父节点；
- 有如下等式：

$$P\left(X\middle|U_1,\cdots U_j,\neg U_{j+1},\cdots\neg U_k\right)=1-\prod_{i=1}^{j}q_i$$

- 该模型的参数是关于父节点个数线性的。

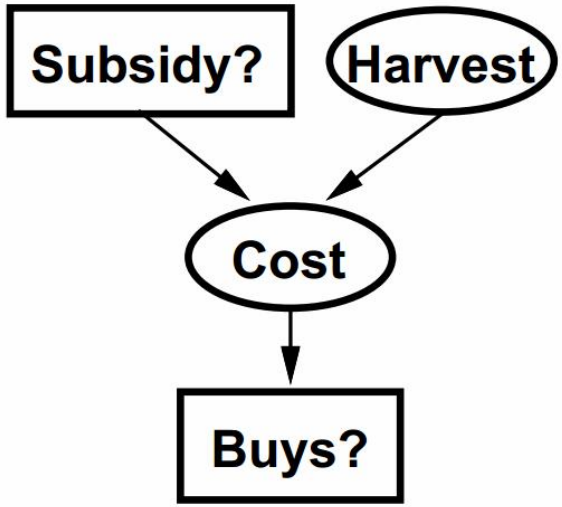
NoisyOR分布模型举例

$Cold$	Flu	$Malaria$	$P(Fever)$	$P(\neg Fever)$
F	F	F	**0.0**	1.0
F	F	T	0.9	**0.1**
F	T	F	0.8	**0.2**
F	T	T	0.98	$0.02 = 0.2 \times 0.1$
T	F	F	0.4	**0.6**
T	F	T	0.94	$0.06 = 0.6 \times 0.1$
T	T	F	0.88	$0.12 = 0.6 \times 0.2$
T	T	T	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

$$P\left(X \middle| U_1, \cdots U_j, \neg U_{j+1}, \cdots \neg U_k\right) = 1 - \prod_{i=1}^{j} q_i$$

混合(离散+连续)网络

Discrete (*Subsidy?* and *Buys?*); continuous (*Harvest* and *Cost*)



Option 1: discretization—possibly large errors, large CPTs
Option 2: finitely parameterized canonical families

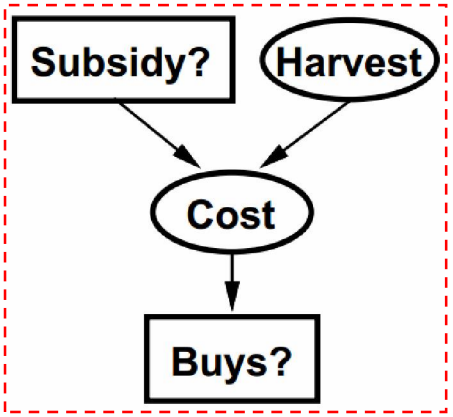
1) Continuous variable, discrete+continuous parents (e.g., *Cost*)
2) Discrete variable, continuous parents (e.g., *Buys?*)

孩子结点是连续的

Need one conditional density function for child variable given continuous parents, for each possible assignment to discrete parents

Most common is the linear Gaussian model, e.g.,:

$$
\begin{aligned}
&P(Cost = c | Harvest = h, Subsidy? = true) \\
&= N(a_t h + b_t, \sigma_t)(c) \\
&= \frac{1}{\sigma_t \sqrt{2\pi}} exp\left(-\frac{1}{2}\left(\frac{c - (a_t h + b_t)}{\sigma_t}\right)^2\right)
\end{aligned}
$$

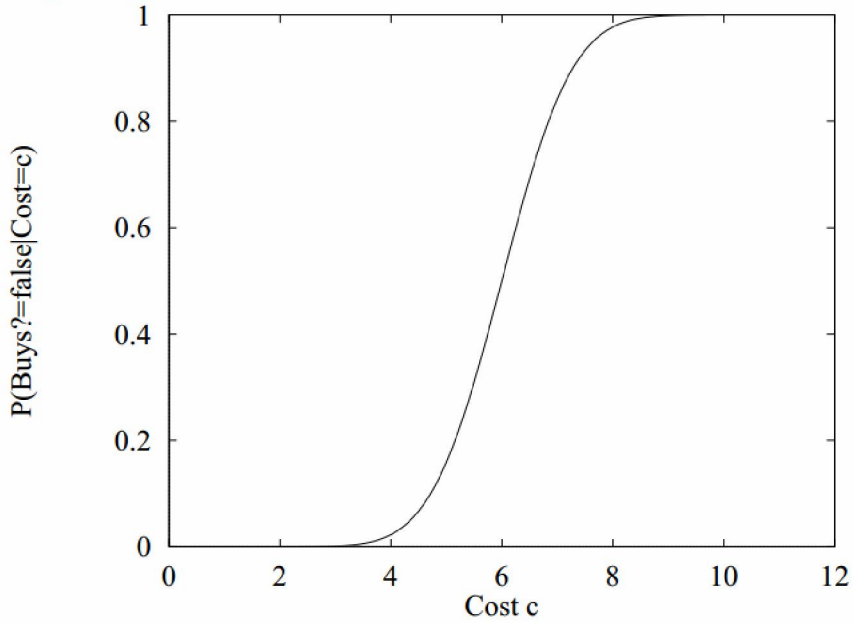


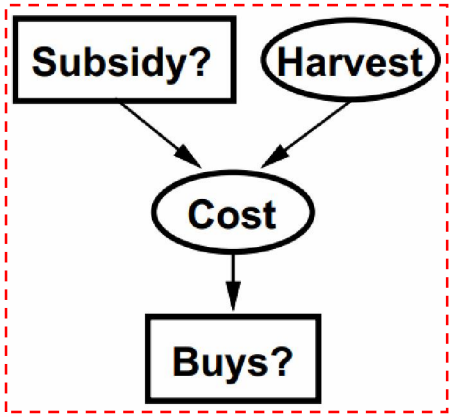
Mean $Cost$ varies linearly with $Harvest$, variance is fixed

Linear variation is unreasonable over the full range
but works OK if the **likely** range of $Harvest$ is narrow

孩子结点是离散的，父节点是连续的

Probability of $Buys?$ given $Cost$ should be a "soft" threshold:





Probit distribution uses integral of Gaussian:

$\Phi(x) = \int_{-\infty}^{x} N(0,1)(x)dx$

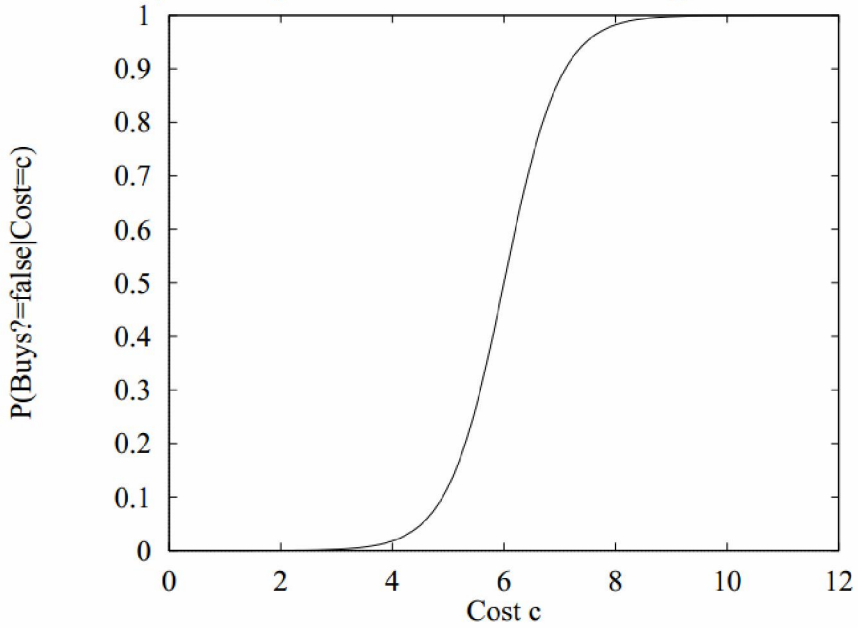
$P(Buys? = true \mid Cost = c) = \Phi((-c + \mu)/\sigma)$

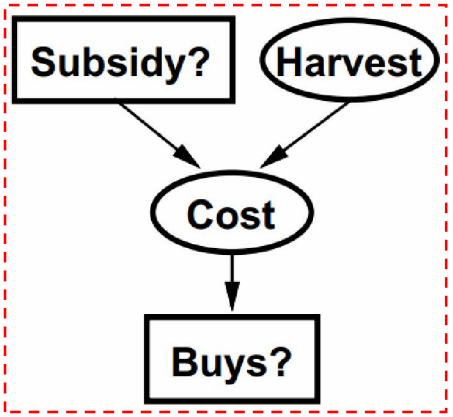
孩子结点是离散的，父节点是连续的

Sigmoid (or logit) distribution also used in neural networks:

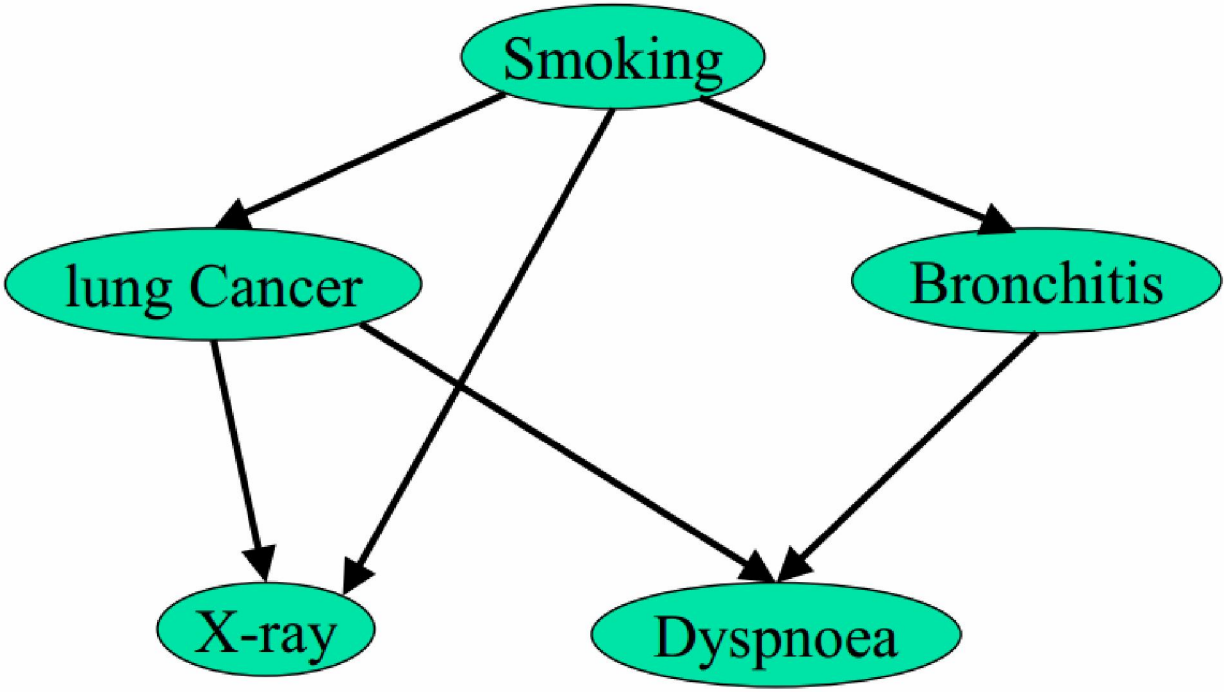
$$P(Buys? = true \mid Cost = c) = \frac{1}{1 + exp(-2\frac{-c+\mu}{\sigma})}$$

Sigmoid has similar shape to probit but much longer tails:



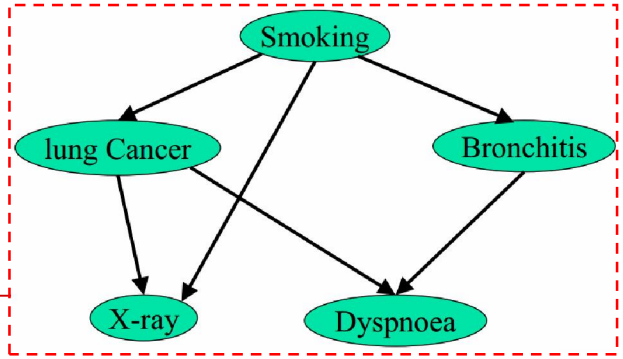


贝叶斯网络的推导



P (smoking| dyspnoea=yes) = ?

贝叶斯网络的推导



$$P(s|d=1) = \frac{P(s,d=1)}{P(d=1)} \propto P(s,d=1) =$$

$$\sum_{d=1,b,x,c} P(s)P(c|s)P(b|s)P(x|c,s)P(d|c,b) =$$

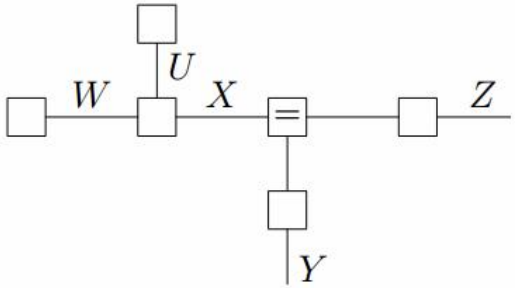
$$P(s)\sum_{d=1}\sum_{b}P(b|s)\sum_{x}\sum_{c}P(c|s)P(x|c,s)P(d|c,b)$$

Variable Elimination

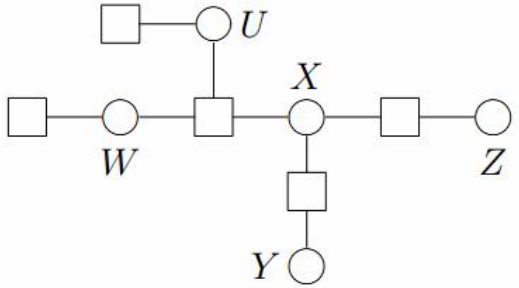
$$f(s,d,b,x)$$

因子图

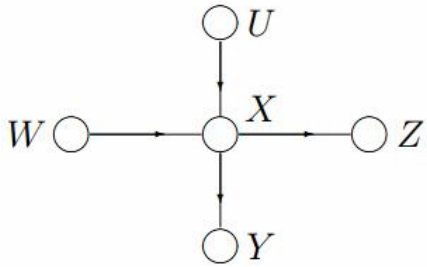
$$p(u, w, x, y, z) = p(u)p(w)p(x|u, w)p(y|x)p(z|x)$$



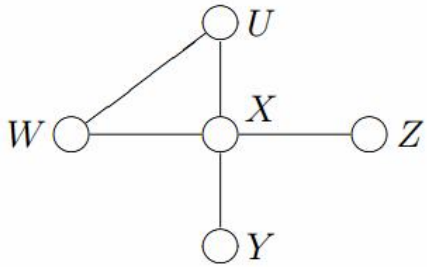
Forney-style factor graph.



Original factor graph [FKLW 1997].



Bayesian network.

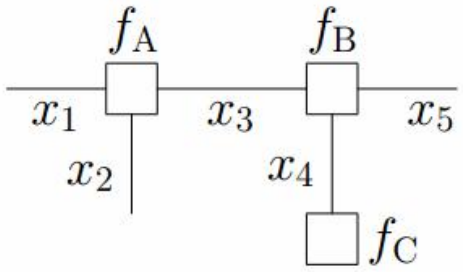


Markov random field.

因子图的构造

- 由贝叶斯网络构造因子图的方法：
  - 一个因子对应因子图中的一个结点
  - 贝叶斯网络中的每一个变量在因子图上对应边或者半边
  - 结点g和边x相连当且仅当变量x出现在因子g中

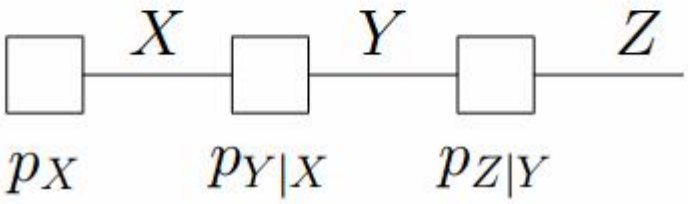
$$f(x_1, x_2, x_3, x_4, x_5) = f_A(x_1, x_2, x_3) \cdot f_B(x_3, x_4, x_5) \cdot f_C(x_4)$$



因子图举例

- 马尔科夫模型

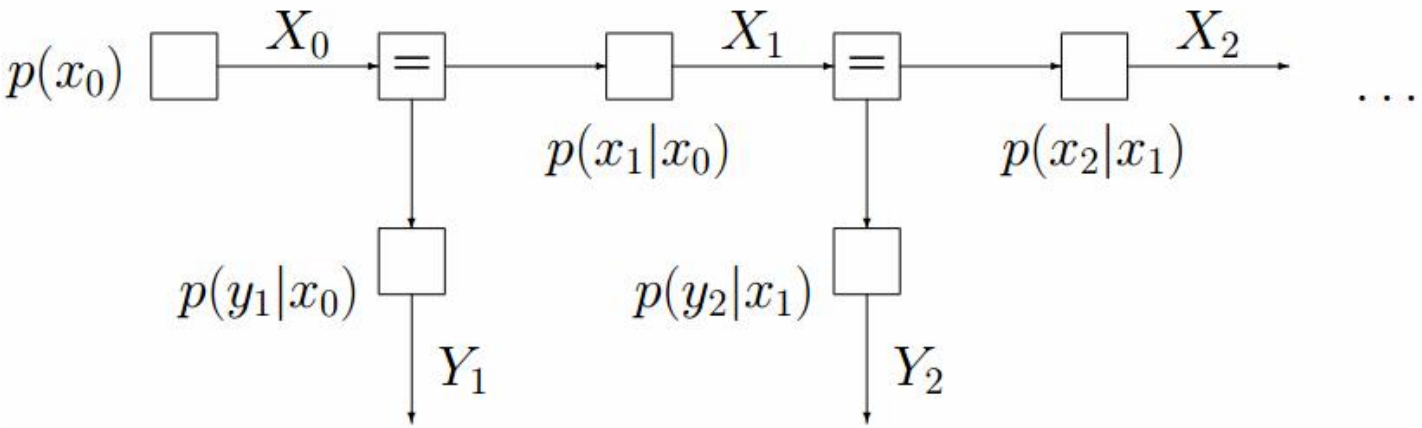
$$p_{XYZ}(x,y,z) = p_X(x)\, p_{Y|X}(y|x)\, p_{Z|Y}(z|y)$$



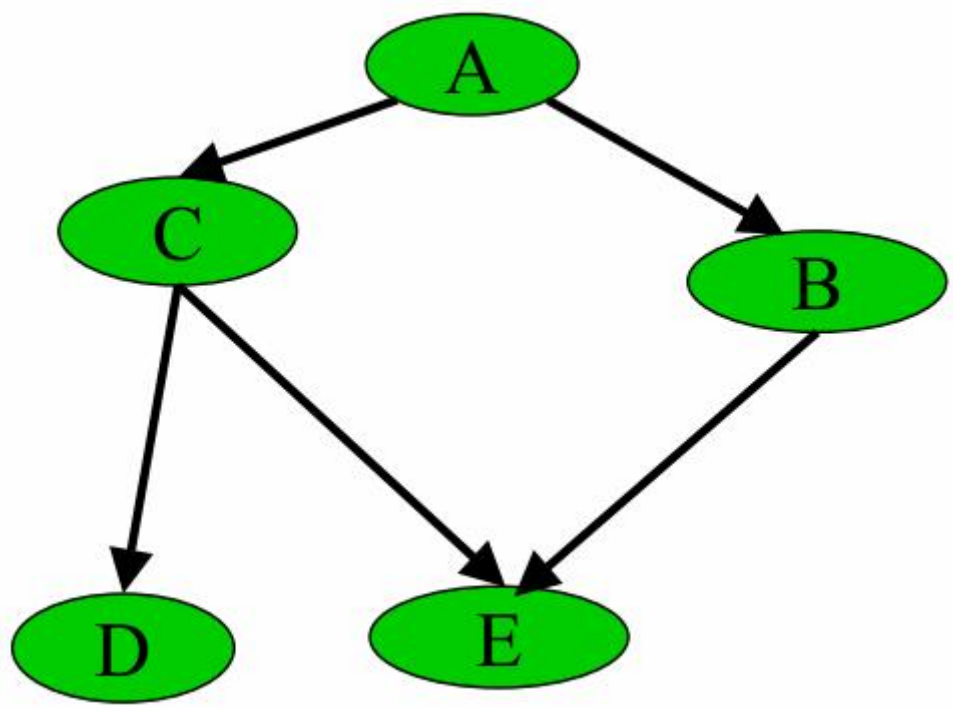
因子图举例

- 隐马尔科夫模型

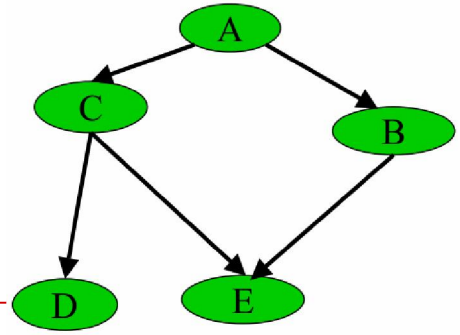
$$p(x_0, x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n) = p(x_0) \prod_{k=1}^{n} p(x_k|x_{k-1}) p(y_k|x_{k-1})$$



试给出该贝叶斯网络的因子图

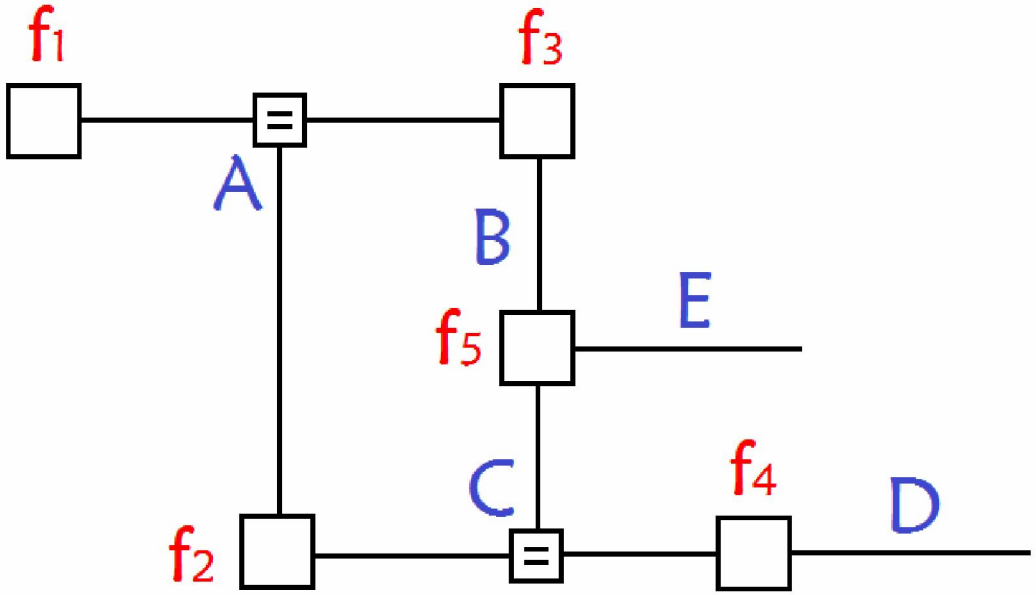


上述贝叶斯网络的因子图



$$P(A,B,C,D,E)=P(A)*P(C|A)*P(B|A)*P(D|C)*P(E|B,C)$$

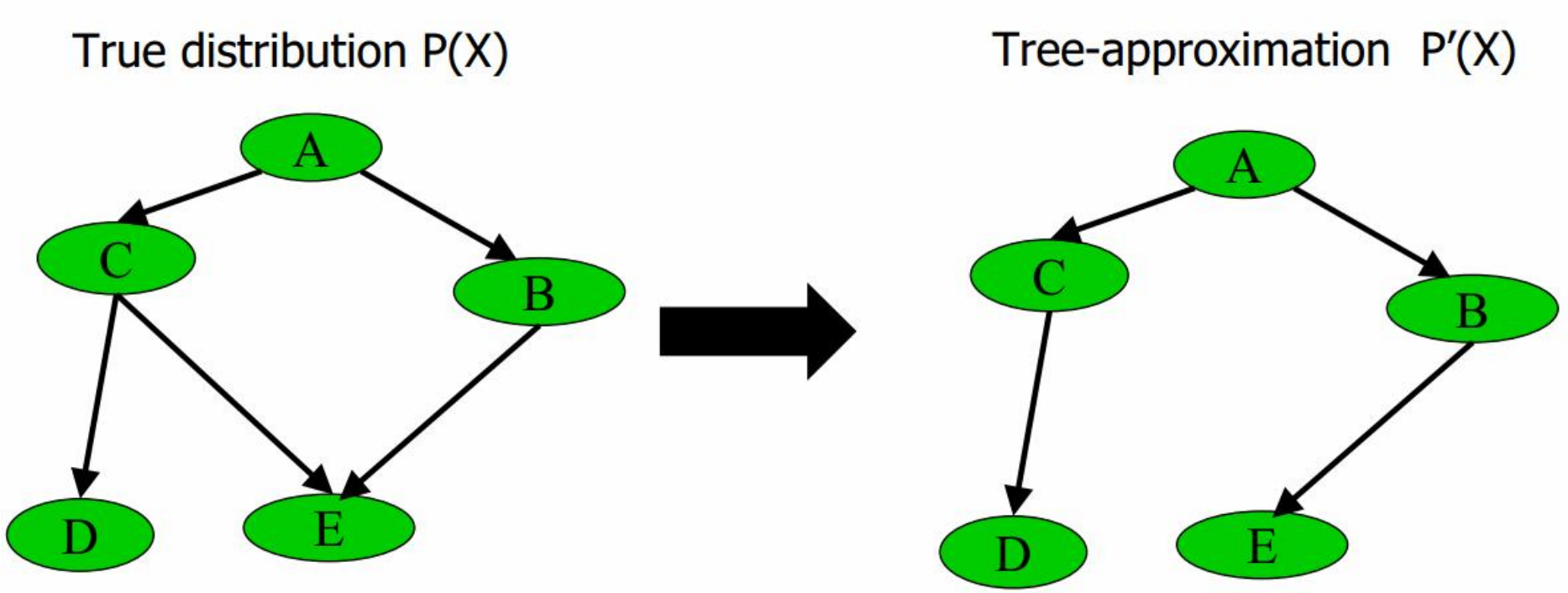
$$\overset{def}{=} f1(A)*f2(A,C)*f3(A,B)*f4(C,D)*f5(B,C,E)$$



无向环

- 可以发现，若贝叶斯网络中存在“环”(无向)，则因此构造的因子图会得到环。而使用消息传递的思想，这个消息将无限传输下去，不利于概率计算。
- 解决方法：
  - 删除贝叶斯网络中的若干条边，使得它不含有无向环
  - 重新构造没有环的贝叶斯网络

原贝叶斯网络的近似树结构



$$D(p \| q) = \sum_x p(x) \log \frac{p(x)}{q(x)} = E_{p(x)} \log \frac{p(x)}{q(x)}$$

将两图的相对熵转换成变量的互信息

Theorem [Chow and Liu, 1968]

Given a joint PDF P(x), the KL-divergence D(P,P') is minimized by projecting P(x) on a *maximum-weight spanning tree (MSWT)* over nodes in X, where the weight on the edge (X_i, X_j) is defined by the mutual information measure

$$I(X_i; X_j) = \sum_{x_i, x_j} P(x_i, x_j) \log \frac{P(x_i, x_j)}{P(x_i) P(x_j)}$$

最大权生成树MSWT的建立过程

- 1. 对于给定的分布P(x)，对于所有的i≠j，计算联合分布P(xi|xj)；
- 2.使用第1步得到的概率分布，计算任意两个结点的互信息I(Xi,Yj)，并把I(Xi,Yj)作为这两个结点连接边的权值；
- 3.计算最大权生成树(Maximum-weight spanning tree)
  - a. 初始状态：n个变量(结点)，0条边
  - b. 插入最大权重的边
  - c. 找到下一个最大的边，并且加入到树中；要求加入后，没有环生成。否则，查找次大的边；
  - d. 重复上述过程c过程直到插入了n-1条边(树建立完成)
- 4. 选择任意结点作为根，从根到叶子标识边的方向；
- 5. 可以保证，这课树的近似联合概率P'(x)和原贝叶斯网络的联合概率P(x)的相对熵最小。

附：Chow-Liu算法

1. From the given distribution P(x) (or from data generated by P(x)), compute the joint distribution $P(x_i \mid x_j)$ for all $i \neq j$
2. Using the pairwise distributions from step 1, compute the mutual information $I(X_i; X_j)$ for each pair of nodes and assign it as the weight to the corresponding edge (X_i, X_j) .
3. Compute the maximum-weight spanning tree (MSWT):
   a. Start from the empty tree over n variables
   b. Insert the two largest-weight edges
   c. Find the next largest-weight edge and add it to the tree if no cycle is formed; otherwise, discard the edge and repeat this step.
   d. Repeat step (c) until n-1 edges have been selected (a tree is constructed).
4. Select an arbitrary root node, and direct the edges outwards from the root.
5. Tree approximation P'(x) can be computed as a projection of P(x) on the resulting directed tree (using the product-form of P'(x)).

参考文献

- Pattern Recognition and Machine Learning Chapter 8, M. Jordan, J. Kleinberg, ect, 2006
- An Introduction to Factor Graphs,Hans-Andrea Loeliger,MLSB 2008
- Factor graph and sum-product algorithm, Frank R. Kschischang, Brendan J.Frey, ect, 1998
- A Tutorial on Inference and Learning in Bayesian Networks, Irina Rish
- A Tutorial on Learning With Bayesian Networks, David Heckerman, 1996
- http://en.wikipedia.org/wiki/Factor_graph(factor graph)
- http://www.eng.yale.edu/pjk/eesrproj_02/luckenbill_html/node4.html(sum-product)

我们在这里

- 更多算法面试题在7|七月算法官网
  - http://www.julyedu.com/
    - 免费视频
    - 直播课程
    - 问答社区

- contact us：微博
  - @研究者July
  - @七月问答
  - @邹博_机器学习

感谢大家！

恳请大家批评指正！