贝叶斯网络BN

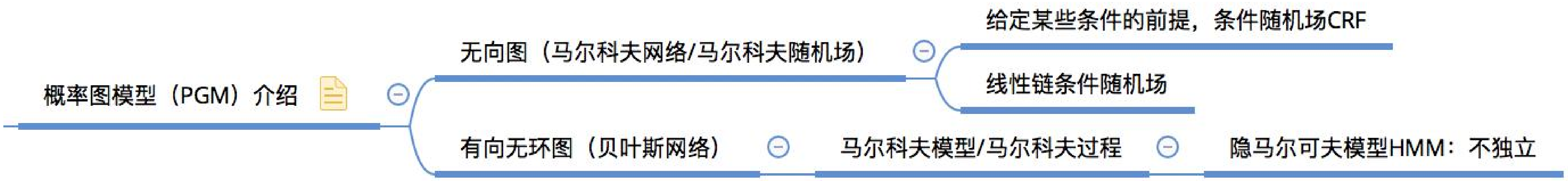
邱小丽 **2020-04-30**

贝叶斯网络

1. 了解概率图模型
2. 了解贝叶斯网络
    1. 三种结构形式
    2. 因子图
    3. **summ-product**算法
    4. 马尔科夫模型
3. 马尔科夫模型**demo**

概率图模型

- 概率图模型是用图来表示变量概率依赖关系的理论，结合概率论与图论的知识，利用图来表示与模型有关的变量的联合概率分布。由图灵奖获得者**Pearl**开发出来。
- 对于一个实际问题，我们希望能够挖掘隐含在数据中的知识。概率图模型构建了这样一幅图，用观测结点表示观测到的数据，用隐含结点表示潜在的知识，用边来描述知识与数据的相互关系，最后基于这样的关系图获得一个概率分布，非常“优雅”地解决了问题。
- 从概率论的角度，节点对应于随机变量，边对应于随机变量的依赖或相关关系，其中有向边表示单向的依赖，无向边表示相互依赖关系。

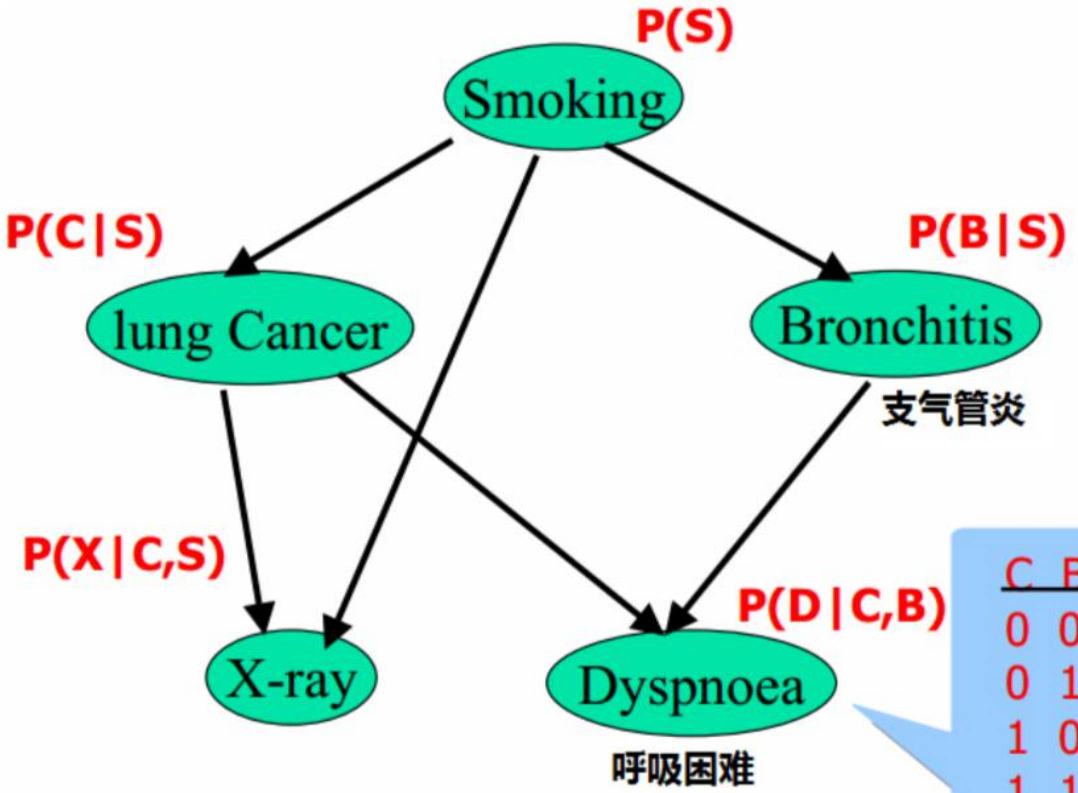


LDA主题模型

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**),又称为信念网络(**belief network**)或者有向无环图模型(**directed acyclic graphical modal, 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_3 . . . X_n}及其n组条件概率分布(**Conditional Probability Distributions,CPD**)的性质。
- 一般而言，贝叶斯网络的有向无环图中的结点表示随机变量，可以是可观察的变量、隐变量、未知参数等。箭头代表此两个随机变量非条件独立，两结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。

贝叶斯网络举例



BN(G, Θ)

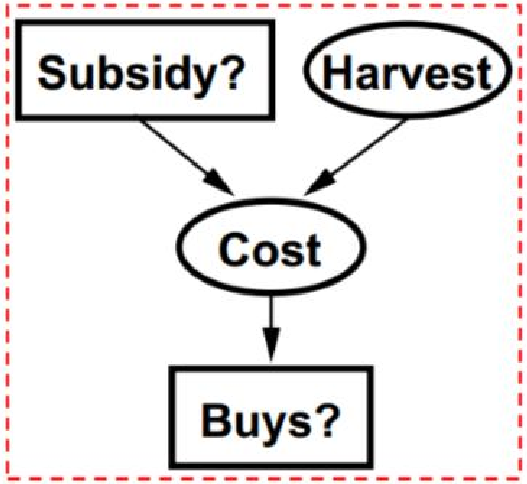
- G:有向无环图
- G的结点：随机变量
- G的边：结点间的**有向依赖**
- Θ：所有条件概率分布的参数集合
- 结点X的条件概率：P(X|parent(X))

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

CPD:

C	B	D=0	D=1
0	0	0.9	0.1
0	1	0.3	0.7
1	0	0.2	0.8
1	1	0.1	0.9

混合(离散+连续)网络

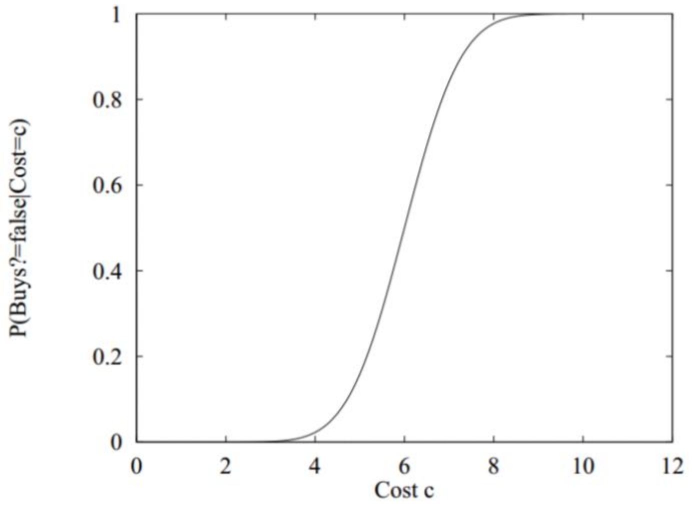


1.孩子结点是连续的

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

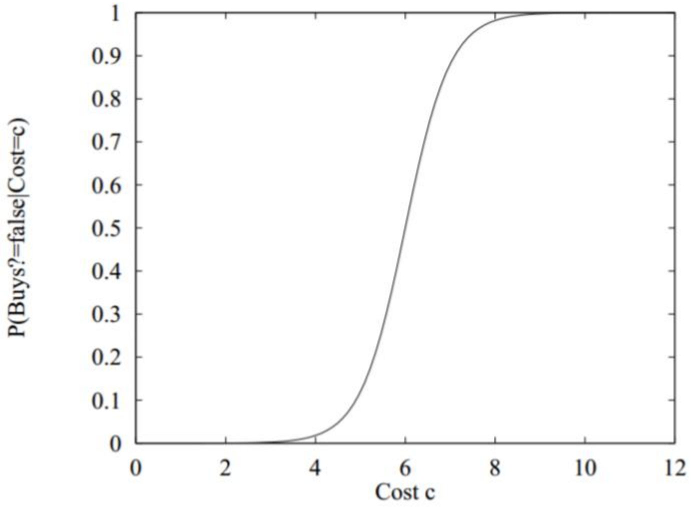
2.孩子结点是离散的，父结点是连续的

$$\Phi(x) = \int_{-\infty}^{x} N(0,1)(x)dx$$
$$P(Buys?=true \mid Cost=c) = \Phi((-c+\mu)/\sigma)$$

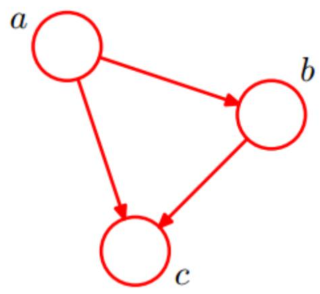


3.孩子结点是离散的，父结点是连续的

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

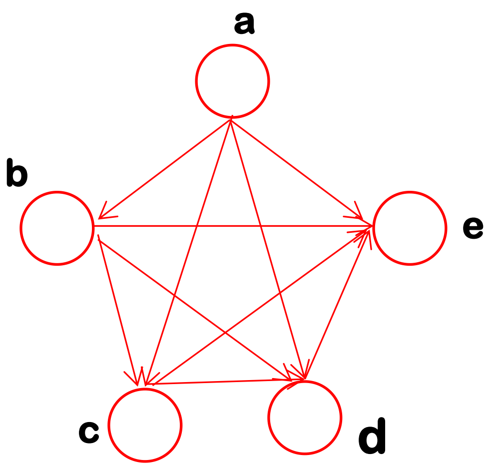


一个简单的贝叶斯网络



P(a,b,c) = p(c|ab)p(b|a)p(a)

全连接贝叶斯网络

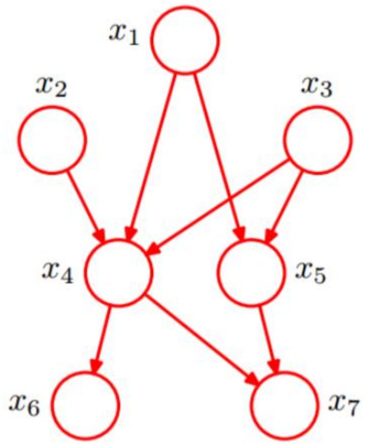


每一对结点之间都有边连接

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

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

一个“正常”的贝叶斯网络

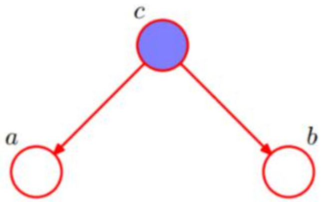


- 有些边缺失
- 直观上：
  - **x1**和**x2**独立
  - **x6** 和**x7**在**x4**给定的条件下独立
- **x1,x2,…x7**的联合分布：

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

叶斯网络中3种结构形式： 通过贝叶斯网络判定条件独立

tail-to-tail



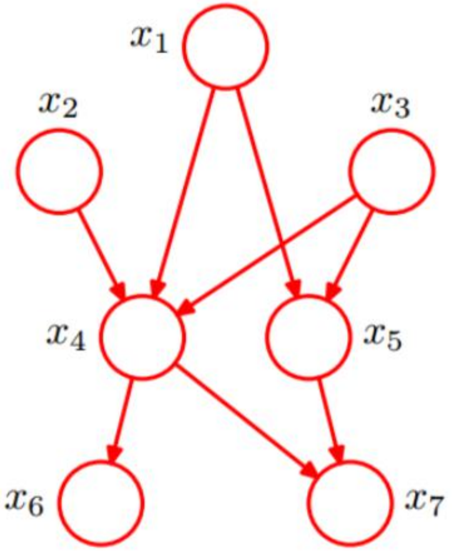
$P(a,b,c)=P(c)*P(a|c)*P(b|c)$

两边都除以**P(c)**

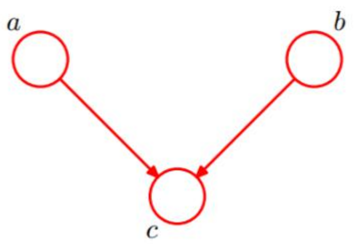
$P(a,b|c)=P(a|c)*P(b|c)$

c未知，**a,b**不独立

c已知，**C**是可观测的结点时，是确定一个条件时，**a,b**是阻断的。**a**和**b**是条件独立的；



head-to-head



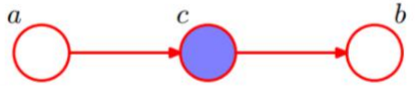
$$P(a,b,c)=P(a)*P(b)*P(c|a,b)$$

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

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

C未知，**a**、**b**被阻断**(blocked),a**和**b**是条件独立的；

head-to-tail



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

c未知，**a,b**不独立

C是可观测的结点时，是确定一个条件时，**a**和**b**是条件独立的；

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

D-separation: 有向分离

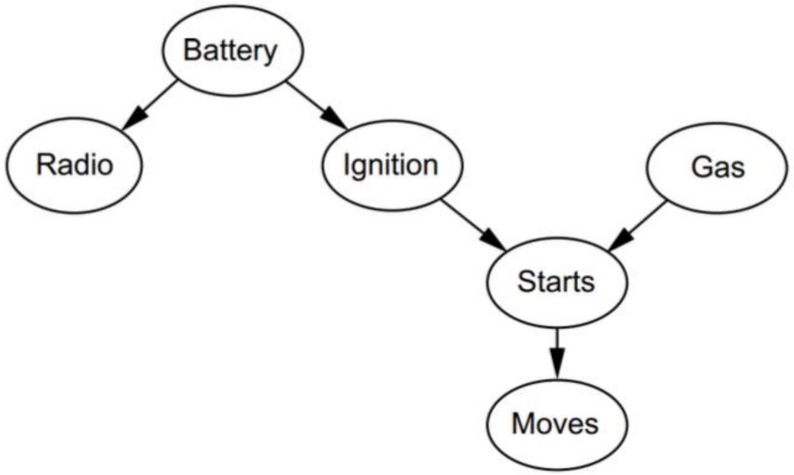
将上述结论推广到结点集

D-separation：有向分离

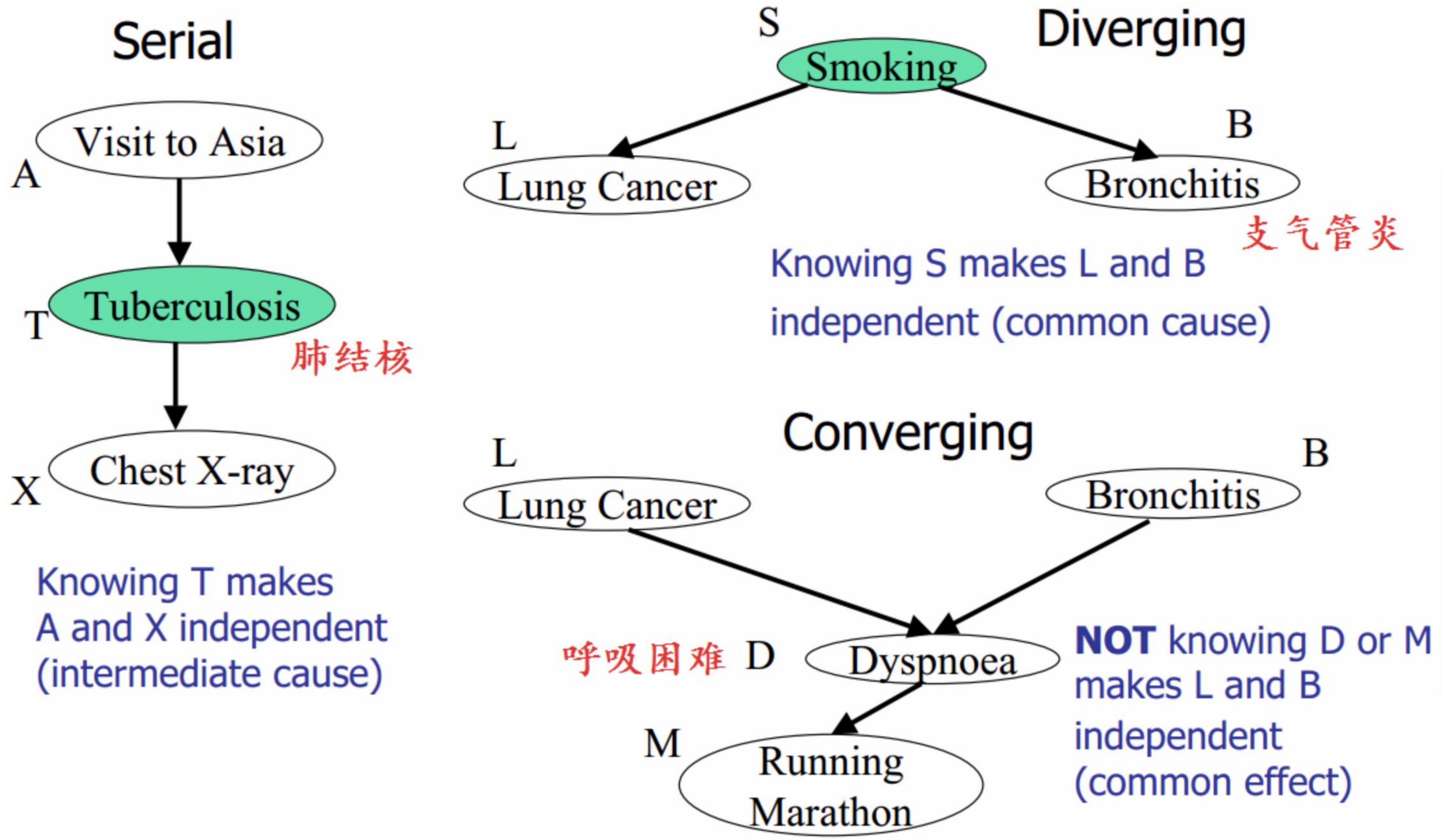
对于任意的结点集A，B，C，考察所有通过A中任意结点到B中任意结点的路径，若要求A，B条件独立，则需要所有的路径都被阻断(blocked)，即满足下列两个前提之一：

- 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；

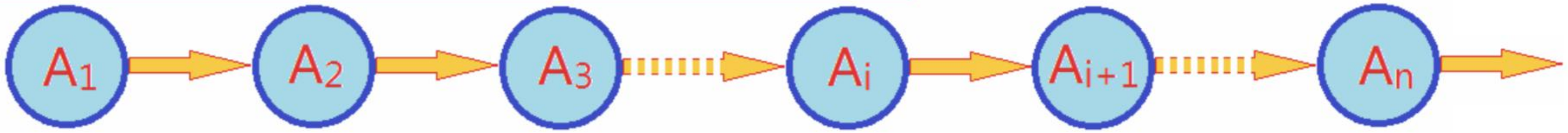
如果A,B不满足D-separation，A,B有时被称为D-connected.



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


Serial
Visit to Asia
A
Tuberculosis
T
肺结核
Chest X-ray
X
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)

再来分析链式网络

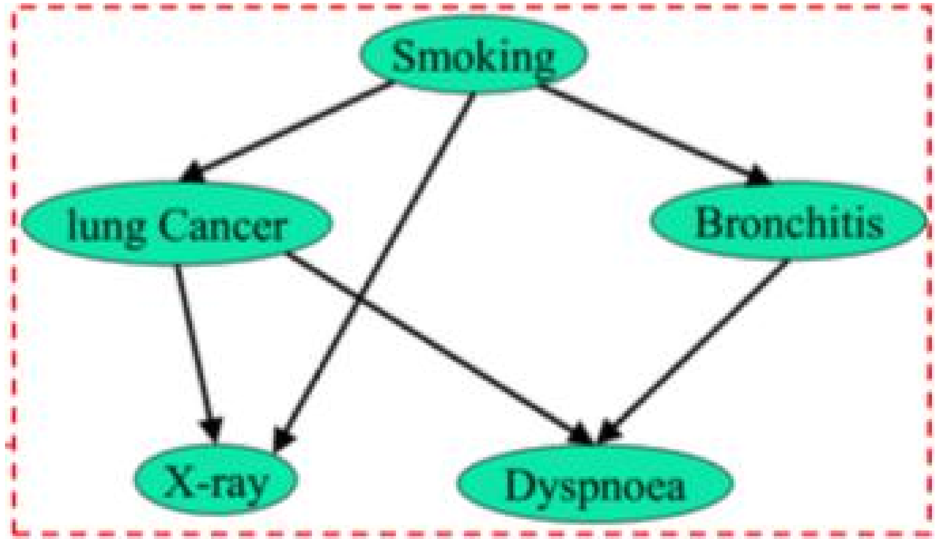


- 结点形成一条链式网络，称作**马尔科夫模型**
  - A_{i+1}只与A_i有关，与$A_1,\ldots,A_{i-1}$无关

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**。

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

变量消除



P (smoking| dyspnoea=yes) = ?

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

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

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

Variable Elimination

在概率图中，求某个变量的边缘分布是常见的问题

- 方法之一就是可以把贝叶斯网络或马尔科夫随机场转换成因子图，然后用消息传递算法**sum-product**算法求解

推导贝叶斯推断的通用公式

- 由贝叶斯网络得到因子图(Factor Graph)
- 通过在因子图中消息传递的思想，计算概率

因子图

- **wikipedia**上是这样定义因子图的：将一个具有多变量的全局函数因子分解，得到几个局部函数的乘积，以此为基础得到的一个双向图叫做因子图**(Factor Graph)**.在概率论及其应用中,因子图是一个在贝叶斯推理中得到广泛应用的模型。

定义

编辑

因子图使用一种二模图用来表示函数因式分解后的结果。设有函数 $g(X_1, X_2, \ldots, X_n)$,

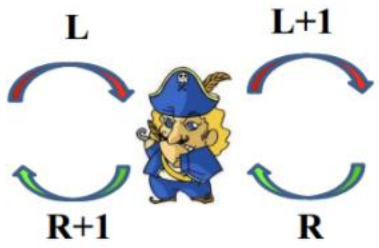
$$g(X_1, X_2, \ldots, X_n) = \prod_{j=1}^{m} f_j(S_j),$$

其中其对应的因子图 $G = (X, F, E)$ 包括变量节点 $X = \{X_1, X_2, \ldots, X_n\}$,因子节点 $F = \{f_1, f_2, \ldots, f_m\}$,和边 E.边通过下列因式分解结果得到：在因子节点 f_j 和变量节点 X_k 之间存在边的充要条件是 $X_k \in S_j$ 存在。[1]

利用因子图的**sum-product**算法求边缘概率分布

理解消息传递

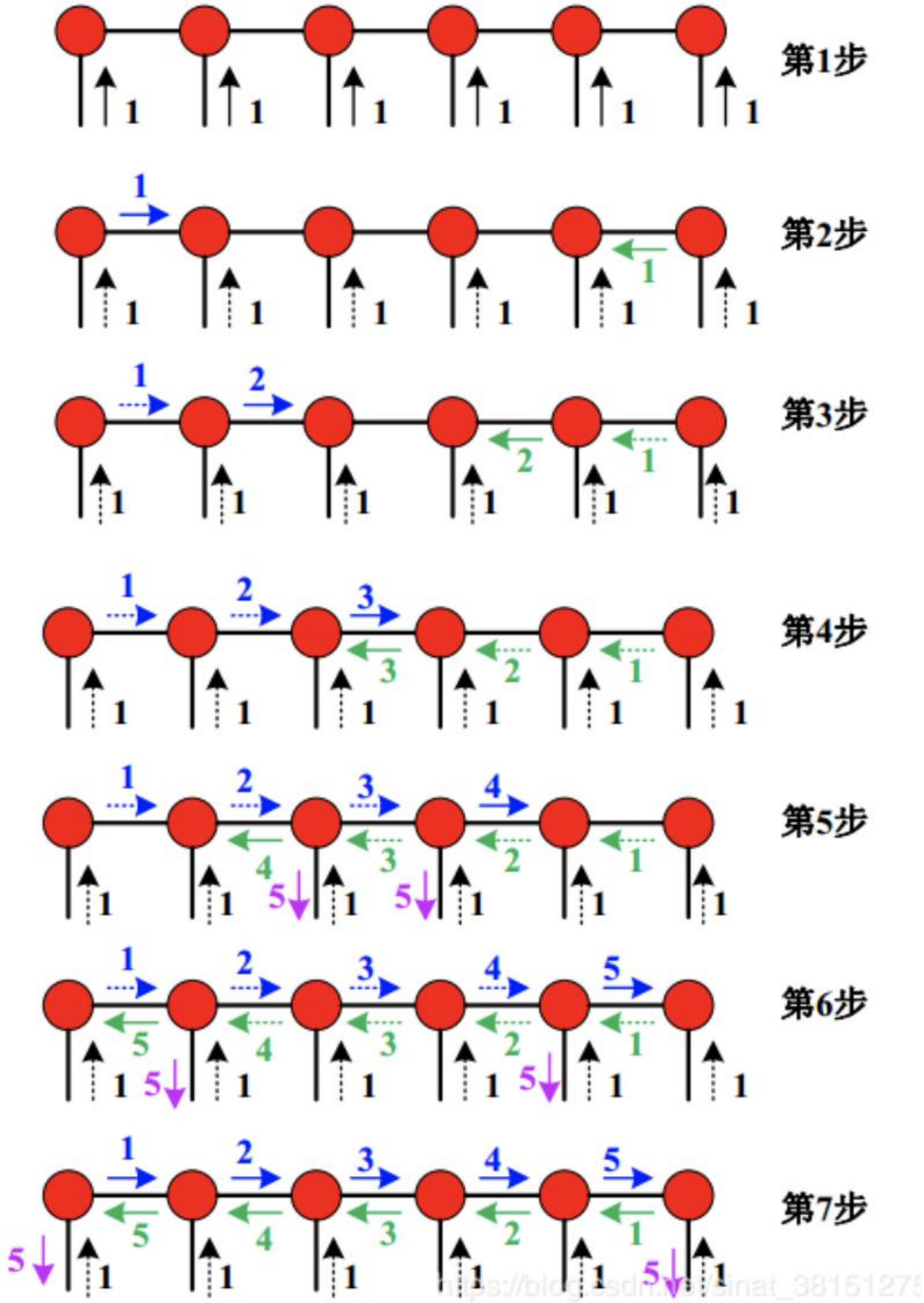
例如， 若干士兵排成一队列， 每个士兵只能与他相邻的

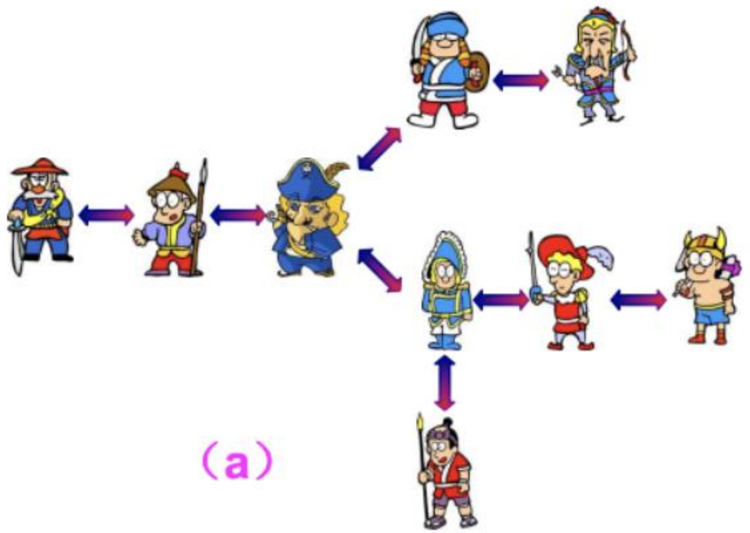


总人数：L+R+1

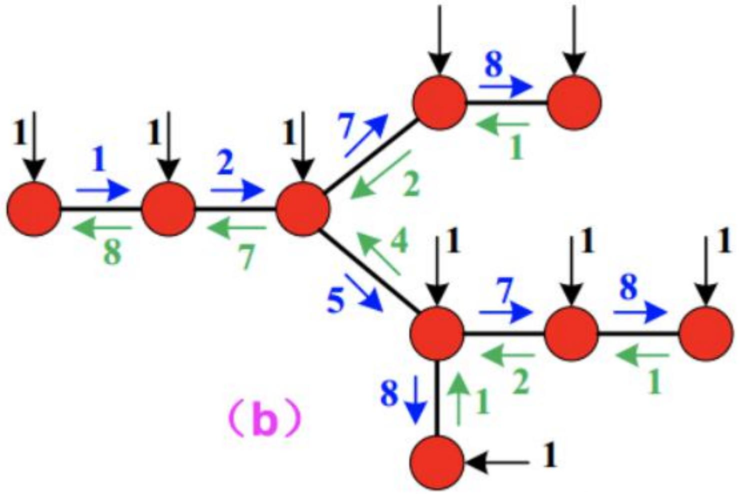
先验信息**P**、 外信息**E**、后验信息**A(P+E)**

两个方向同时计算的前向/后向消息传递过程





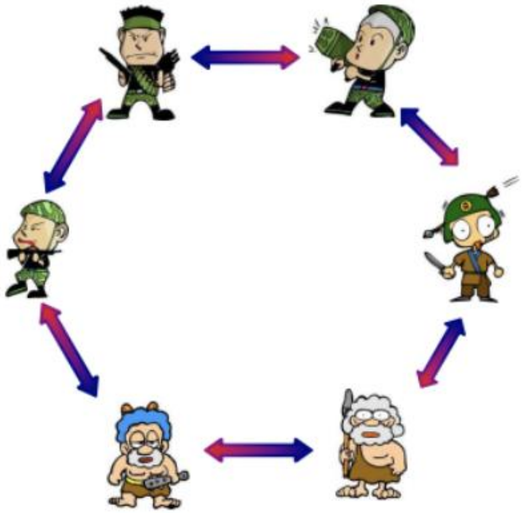
（a）

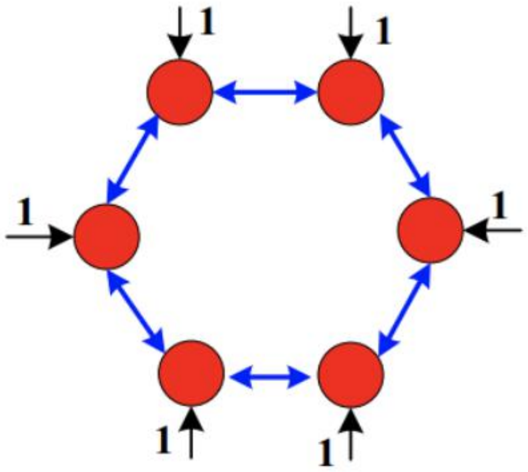


（b）

每个士兵同样可以获得其相邻士兵给他的外信息， 同时加上自身的信息然后传递给相邻的士兵

每个士兵节点的信息只需在所有其相邻节点上进行一次前向和后向的计算， 则每个士兵就可知道总人数。 这样的图有一个共同特点：所有节点构成一棵树， 而树结构中是没有环路的。





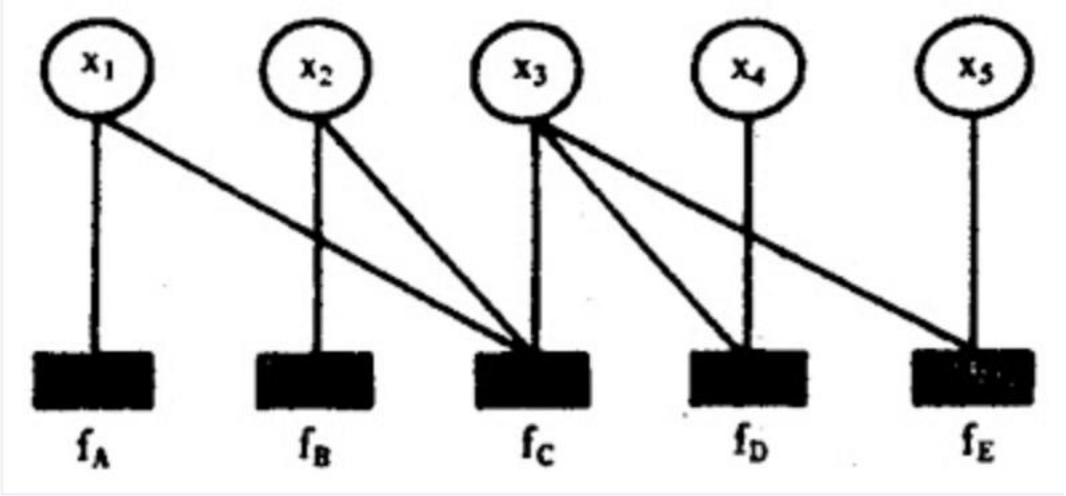
如果有环路， 会导致什么结果？

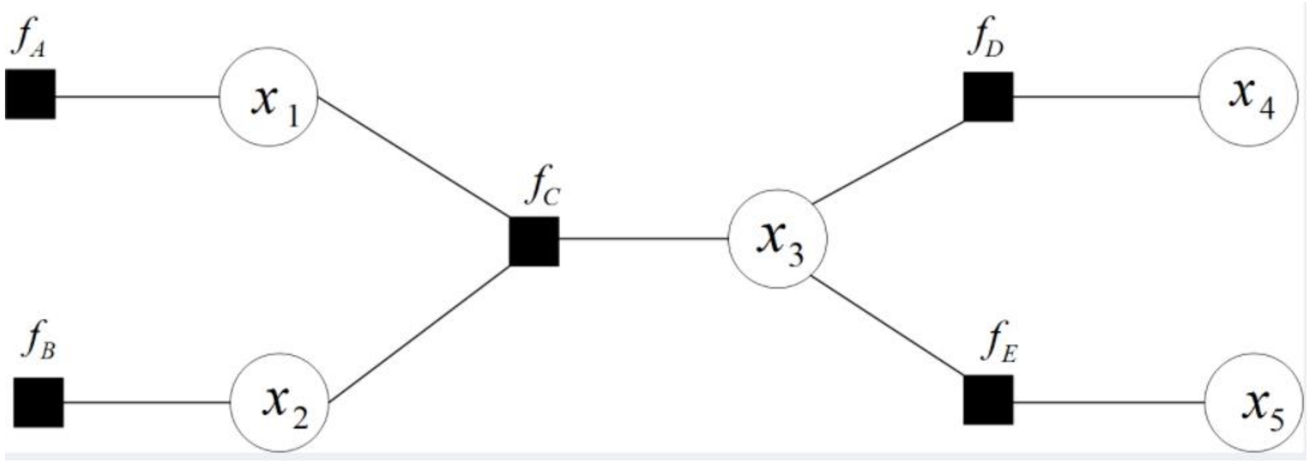
将前述消息传递算法中的节点构成的图更一般化就得到了因子图。 因子图是一种用于描述多变量函数的“二部图” **(Bipartite Graph)**。
在因子图中存在两类节点： 变量节点和对应的函数节点； 变量节点所代表的变量是函数节点的自变量。
边线表示它们之间的函数关系， 同类节点之间没有边直接相连。

举个例子，现在有一个全局函数，其因式分解方程为：

$$g(x_1,x_2,x_3,x_4,x_5) = f_A(x_1)f_B(x_2)f_C(x_1,x_2,x_3)f_D(x_3,x_4)f_E(x_3,x_5)$$

其中**fA,fB,fC,fD,fE**为各函数表示变量之间的关系，可以是条件概率也可以是其他关系**(**如马尔可夫随机场**Markov Random Fields**中的势函数**)**。



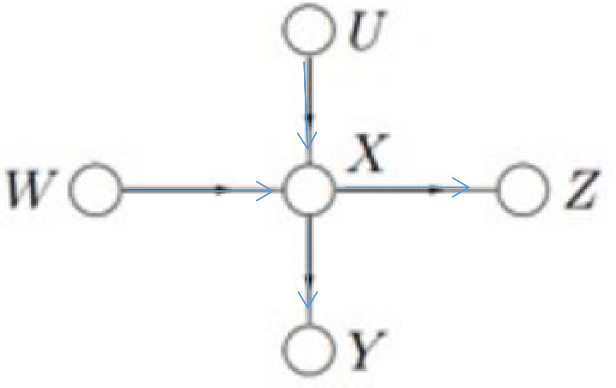


在因子图中，所有顶点不是变量节点就是函数节点，

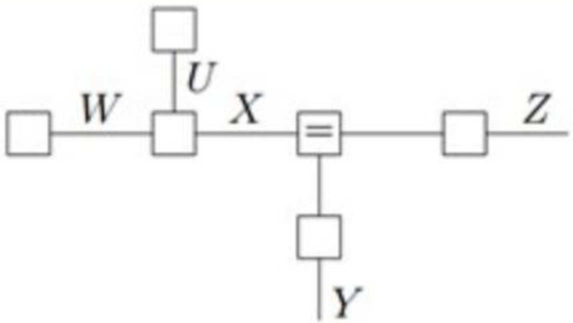
因子图是干嘛的？为什么要引入？用途？意义？

事实上，因子图跟贝叶斯网络和马尔可夫随机场（**Markov Random Fields**）一样，也是是概率图的一种。

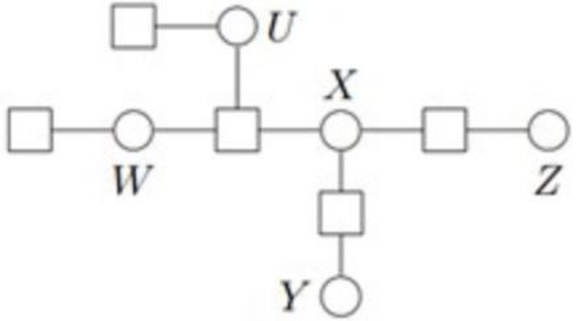
贝叶斯网络(马尔科夫随机场)-->因子图



Bayesian network.

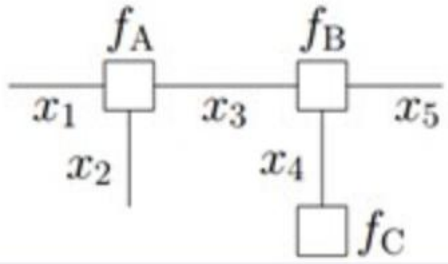


Forney-style factor graph.



Original factor graph [FKLW 1997].

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



由上述例子总结出由贝叶斯网络构造因子图的方法:

贝叶斯网络中的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

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

Summary-product算法;belief progagation;和积算法

问题：如何由联合概率分布求边缘概率分布

联合概率：表示两个事件共同发生的概率。**A**与**B**的联合概率表示为**P(A,B)**。
边缘概率**:**（又称先验概率）是某个事件发生的概率。

边缘概率是这样得到的：在联合概率中，把最终结果中不需要的那些事件合并成其事件的全概率而消失（对离散随机变量用求和得全概率，对连续随机变量用积分得全概率**,**这称为边缘化**(marginalization)**
A的边缘概率表示为**P(A)**，**B**的边缘概率表示为**P(B)**。

事实上，某个随机变量**fk**的边缘概率可由**x1,x2,x3, ..., xn**的联合概率求到，具体公式为：

$$\bar{f}_k(x_k) \triangleq \sum_{\substack{x_1,\ldots,x_n \\ \text{except } x_k}} f(x_1,\ldots,x_n)$$

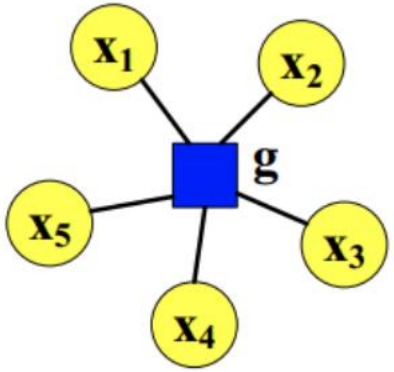
换言之，如果有

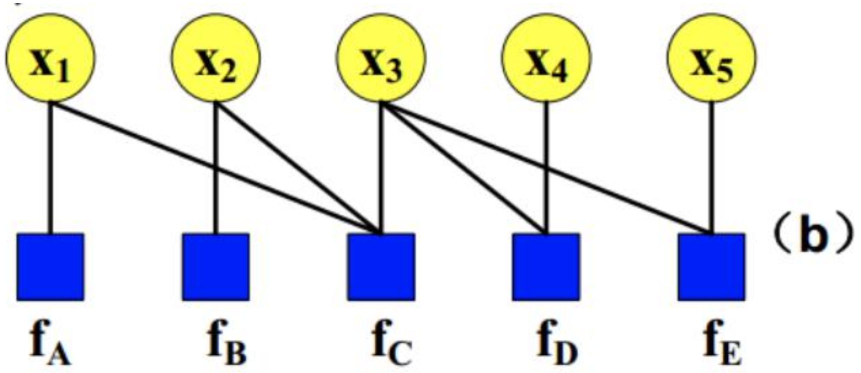
$$f(x_1,\ldots,f_n) = f_1(x_1)f_2(x_2)\cdots f_n(x_n)$$

$$\bar{f}_k(x_k) = \left(\sum_{x_1} f_1(x_1)\right)\cdots\left(\sum_{x_{k-1}} f_{k-1}(x_{k-1})\right)f_k(x_k)\cdots\left(\sum_{x_n} f_n(x_n)\right)$$

左图表示有**5**个自变量的函数**g**的因子图，函数**g**可以分解成几个“局部函数” 之积的形式，即

$$g(x_1,x_2,x_3,x_4,x_5)=f_A(x_1)f_B(x_2)f_C(x_1,x_2,x_3)f_D(x_3,x_4)f_E(x_3,x_5)$$





$$\tilde{g}(x_1)=\sum_{x_2,x_3,x_4,x_5} g(x_1,x_2,x_3,x_4,x_5)$$

$$\tilde{g}(x_1)=\sum_{x_2}\sum_{x_3}\sum_{x_4}\sum_{x_5} f_A(x_1)f_B(x_2)f_C(x_1,x_2,x_3)f_D(x_3,x_4)f_E(x_3,x_5)$$

$$=f_A(x_1)\sum_{x_2} f_B(x_2)\sum_{x_3} f_C(x_1,x_2,x_3)\sum_{x_4} f_D(x_3,x_4)\sum_{x_5} f_E(x_3,x_5)$$

也可写为

$$=f_A(x_1)\sum_{\sim\{x_1\}}\left\{f_B(x_2)f_C(x_1,x_2,x_3)\cdot\left(\sum_{\sim\{x_3\}} f_D(x_3,x_4)\right)\cdot\left(\sum_{\sim\{x_3\}} f_E(x_3,x_5)\right)\right\}$$

$$\sum_{\sim\{x_2\}} h(x_1,x_2,x_3) \triangleq \sum_{x_1\in A_1}\sum_{x_3\in A_3} h(x_1,x_2,x_3)$$

计算复杂，如何简化呢？

$$\tilde{g}(x_1) = \sum_{x_2}\sum_{x_3}\sum_{x_4}\sum_{x_5} f_A(x_1) f_B(x_2) f_C(x_1,x_2,x_3) f_D(x_3,x_4) f_E(x_3,x_5)$$
$$= f_A(x_1)\sum_{x_2} f_B(x_2)\sum_{x_3} f_C(x_1,x_2,x_3)\sum_{x_4} f_D(x_3,x_4)\sum_{x_5} f_E(x_3,x_5)$$

首先进行一些数学定义：

$$f_I(x_3) = \sum_{x_5} f_E(x_3,x_5)$$

$$f_{II}(x_3) = \sum_{x_4} f_D(x_3,x_4)$$

$$f_{III}(x_3) = f_I(x_3)\cdot f_{II}(x_3)$$

可得到：

$$f_{IV}(x_1,x_2) = \sum_{x_3} f_C(x_1,x_2,x_3) f_{III}(x_3)$$

$$f_V(x_1) = \sum_{x_2} f_B(x_2) f_{IV}(x_1,x_2)$$

最终：

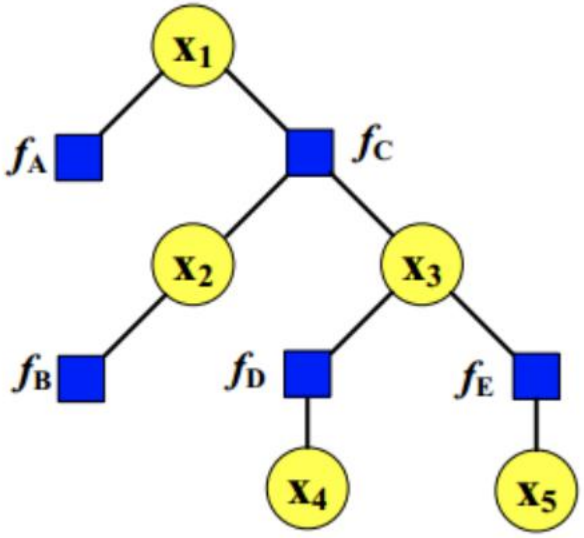
$$\boxed{\tilde{g}(x_1) = f_A(x_1)\cdot f_V(x_1)}$$

简化的思想是基于将一个复杂任务分解成多个小任务，每个小任务对应到因子图上就是一个函数节点。这使得其计算时不需要来自因子图其他部分的信息，且传送其计算结果仅由这些局部函数的自变量来承担，从而简化了计算。

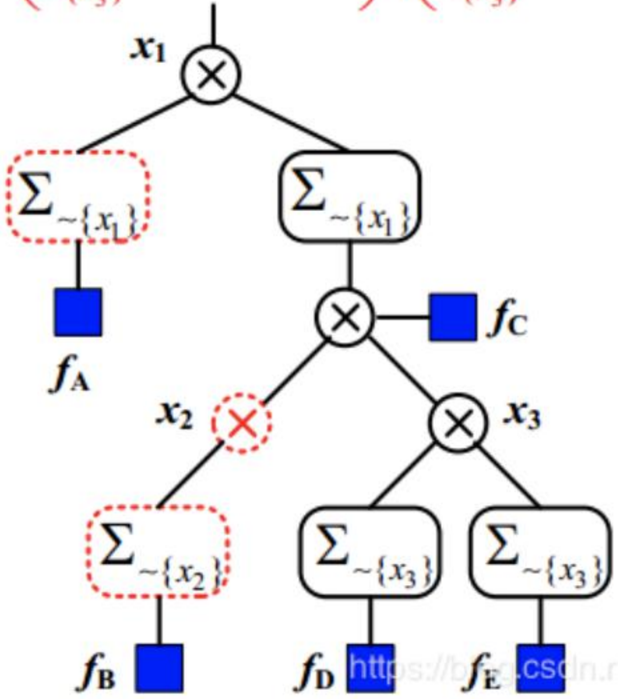
- 和积算法从本质上就是一种消息传递算法，它可从全局函数计算出各个不同的边缘函数。我们在前面求出了边缘函数 $\tilde{g}(x_1)$ ，其因子图表示如（a）所示。因子图只包含了变量与函数的对应关系，而运算关系并没有得到明确、具体的可视化，因此人们又提出了表达式树图（expression trees）的概念， $\tilde{g}(x_1)$ 的表达式树图如（b）所示。

$$g(x_1,x_2,x_3,x_4,x_5)=f_A(x_1)f_B(x_2)f_C(x_1,x_2,x_3)f_D(x_3,x_4)f_E(x_3,x_5)$$

$$\tilde{g}(x_1)=f_A(x_1)\sum_{\sim\{x_1\}}\left\{f_B(x_2)f_C(x_1,x_2,x_3)\cdot\left(\sum_{\sim\{x_3\}}f_D(x_3,x_4)\right)\cdot\left(\sum_{\sim\{x_3\}}f_E(x_3,x_5)\right)\right\}$$



（a）

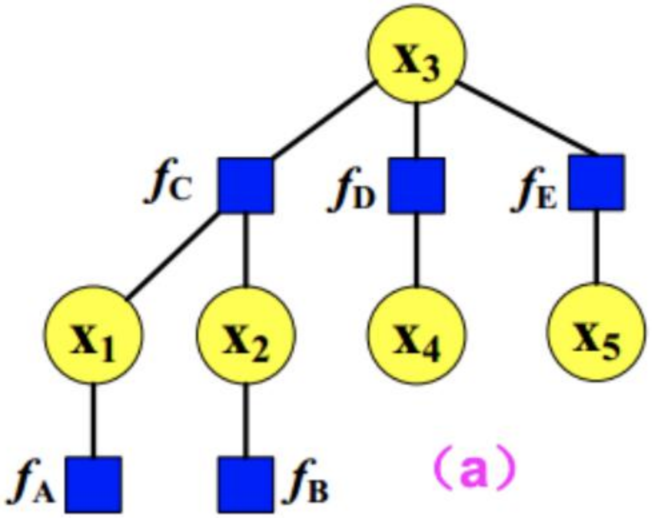


（b）

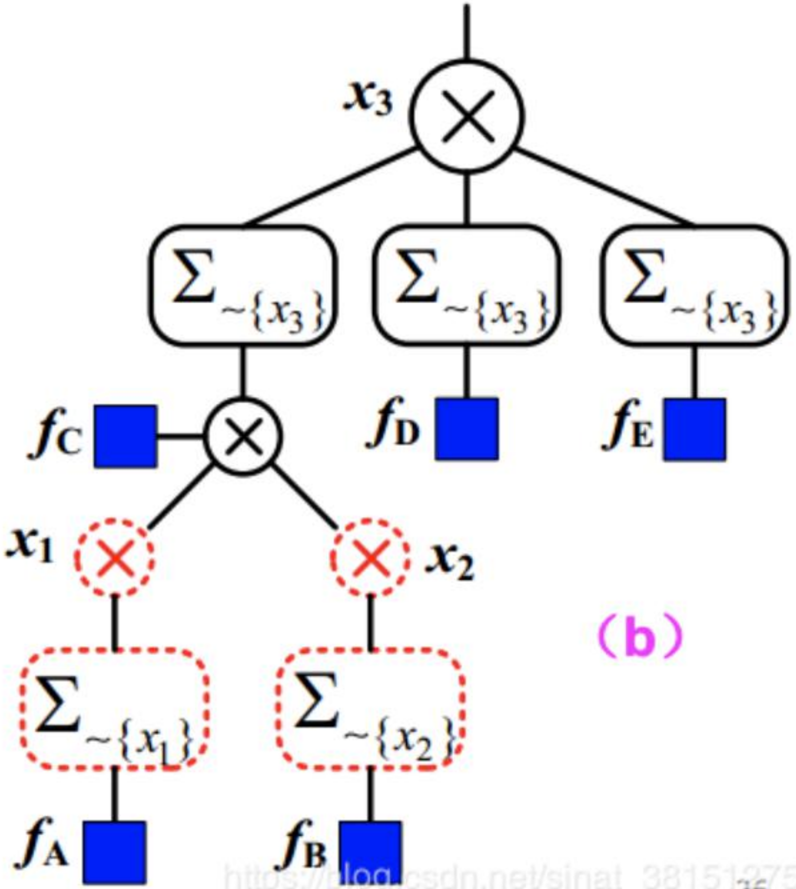
➢ 再例如，我们希望得到边缘函数 $\tilde{g}(x_3)$ 。根据下式，我们可以画出其因子图，如图（a）所示，其表达式树图如（b）所示。

$$g(x_1,x_2,x_3,x_4,x_5)=f_A(x_1)f_B(x_2)f_C(x_1,x_2,x_3)f_D(x_3,x_4)f_E(x_3,x_5)$$

$$\tilde{g}(x_3)=\left(\sum_{\sim\{x_3\}} f_A(x_1)f_B(x_2)f_C(x_1,x_2,x_3)\right)\cdot\left(\sum_{\sim\{x_3\}} f_D(x_3,x_4)\right)\cdot\left(\sum_{\sim\{x_3\}} f_E(x_3,x_5)\right)$$



（a）



（b）

因子图和表达式树图的对应关系？

- 从表示全局函数的因子图到表示各边缘函数 $\tilde{g}(x_i)$ 的表达式树图的转换，应注意以下几点：

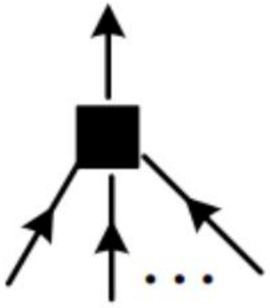
① 因子图中的每个变量用乘积运算符“$\otimes$”代替；

② 因子图中的函数节点用“$\otimes$—■ f”运算符代替（“form product and multiply f”）；

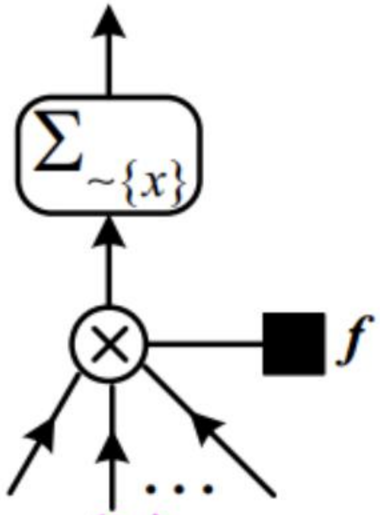
③ 在函数节点 f 和其parent x之间插入“$\sum_{\sim\{x\}}$”和式运算符；

- 这些局部转换如下图所示。图（a）对应于变量节点，图（b）对应于有parent x的函数节点。

没有操作对象的乘积运算符“$\otimes$”可当作乘以1对待，或当它处在表达式树图的末端时，可以忽略； 如果和式运算符“$\sum_{\sim\{x\}}$”运用到只有一个自变量的函数中时，可以省略；

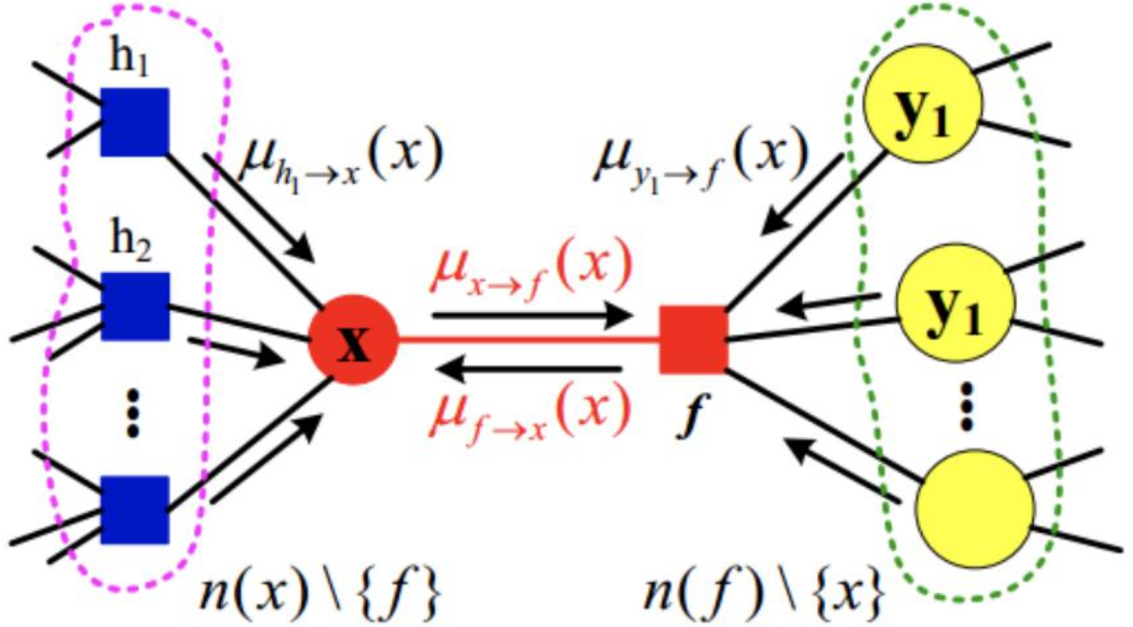


（a）



（b）

➢ 在消息传递的过程中，需要计算不同的和与积，因此又称为和积算法（Sum-Product Algorithm）。和积算法的更新如下图所示



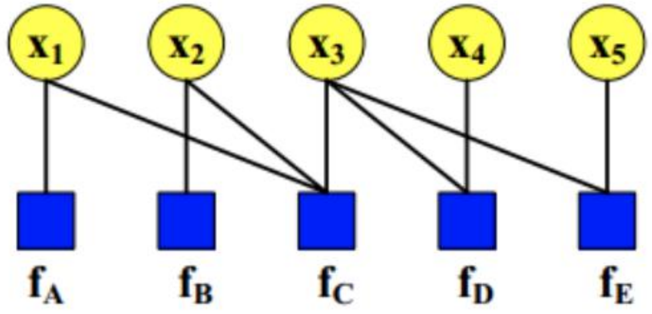
设$u_{x\to f}(x)$表示消息是从节点x到节点f，$u_{f\to x}(x)$表示消息是从节点f到节点x，n(v)表示在因子图中给定节点v的邻居集合。

变量到局部函数：
$$\mu_{x\to f}(x) = \prod_{h\in n(x)\setminus\{f\}} \mu_{h\to x}(x)$$

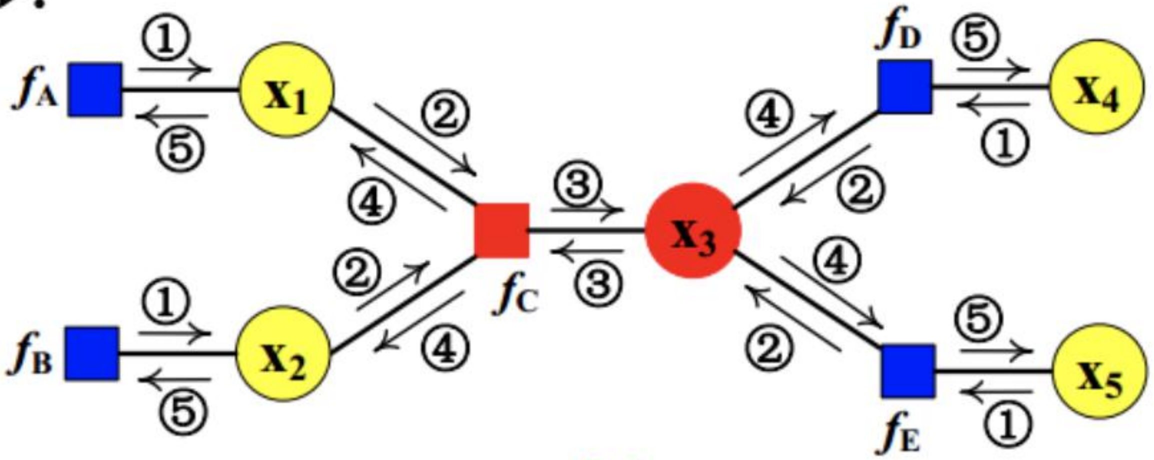
局部函数到变量：
$$\mu_{f\to x}(x) = \sum_{\sim\{x\}} \left(f(X) \cdot \prod_{y\in n(f)\setminus\{x\}} \mu_{y\to f}(y) \right)$$

其中X=n(f)是函数f的自变量集合

➢ 例：前面的因子图如图（a）所示，和积算法产生的消息流如图（b）所示，这些消息的产生有五步：



（a）



（b）

步骤1：

$$\mu_{f_A\to x_1}(x_1)=\sum_{\sim\{x_1\}} f_A(x_1)=f_A(x_1) \qquad \mu_{x_4\to f_D}(x_4)=1$$

$$\mu_{f_B\to x_2}(x_2)=\sum_{\sim\{x_2\}} f_B(x_2)=f_B(x_2) \qquad \mu_{x_5\to f_E}(x_5)=1$$

步骤2：

$$\mu_{x_1\to f_C}(x_1)=\mu_{f_A\to x_1}(x_1) \qquad \mu_{f_D\to x_3}(x_3)=\sum_{\sim\{x_3\}}\left(f_D(x_3,x_4)\cdot\mu_{x_4\to f_D}(x_4)\right)$$

$$\mu_{x_2\to f_C}(x_2)=\mu_{f_B\to x_2}(x_2) \qquad \mu_{f_E\to x_3}(x_3)=\sum_{\sim\{x_3\}}\left(f_E(x_3,x_5)\cdot\mu_{x_5\to f_E}(x_5)\right)$$

步骤3：

$$\mu_{f_C \to x_3}(x_3) = \sum_{\sim\{x_3\}} \left(f_C(x_1, x_2, x_3) \cdot \mu_{x_1 \to f_C}(x_1) \cdot \mu_{x_2 \to f_C}(x_2) \right)$$

$$\mu_{x_3 \to f_C}(x_3) = \mu_{f_D \to x_3}(x_3) \cdot \mu_{f_E \to x_3}(x_3)$$

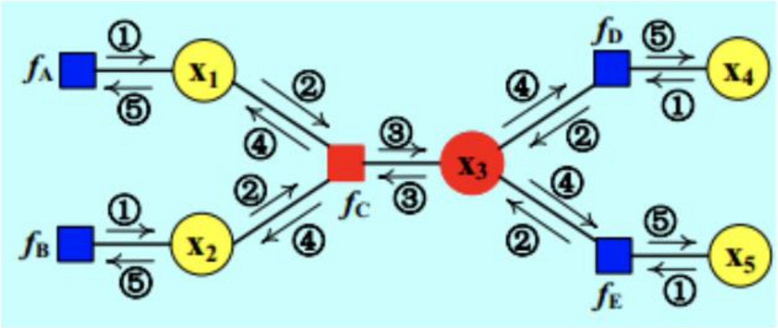
步骤4：

$$\mu_{f_C \to x_1}(x_1) = \sum_{\sim\{x_1\}} \left(f_C(x_1, x_2, x_3) \cdot \mu_{x_3 \to f_C}(x_3) \cdot \mu_{x_2 \to f_C}(x_2) \right)$$

$$\mu_{f_C \to x_2}(x_2) = \sum_{\sim\{x_2\}} \left(f_C(x_1, x_2, x_3) \cdot \mu_{x_3 \to f_C}(x_3) \cdot \mu_{x_1 \to f_C}(x_1) \right)$$

$$\mu_{x_3 \to f_D}(x_3) = \mu_{f_C \to x_3}(x_3) \cdot \mu_{f_E \to x_3}(x_3)$$

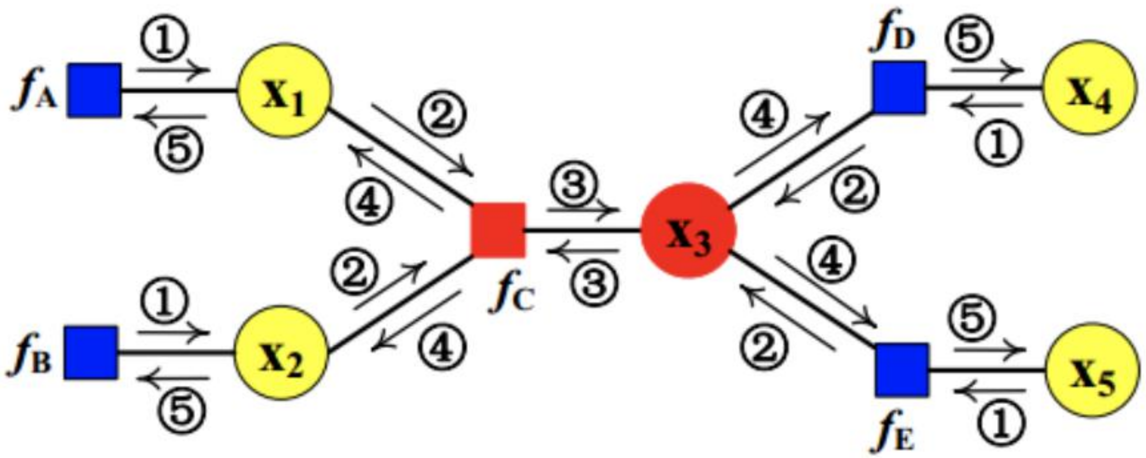
$$\mu_{x_3 \to f_E}(x_3) = \mu_{f_C \to x_3}(x_3) \cdot \mu_{f_D \to x_3}(x_3)$$



步骤5：

$$\mu_{x_1 \to f_A}(x_1) = \mu_{f_C \to x_1}(x_1) \qquad \mu_{f_D \to x_4}(x_4) = \sum_{\sim\{x_4\}} \left(f_D(x_3, x_4) \cdot \mu_{x_3 \to f_D}(x_3) \right)$$

$$\mu_{x_2 \to f_B}(x_2) = \mu_{f_C \to x_2}(x_2) \qquad \mu_{f_E \to x_5}(x_5) = \sum_{\sim\{x_5\}} \left(f_E(x_3, x_5) \cdot \mu_{x_3 \to f_E}(x_3) \right)$$



➢ 我们计算某个边缘函数 $\tilde{g}(x_i)$ ，就等效为送给 x_i 所有消息的乘积，因此就可得到每个边缘函数。

$$\tilde{g}(x_1)=\mu_{f_A\to x_1}(x_1)\cdot\mu_{f_C\to x_1}(x_1)$$

$$\tilde{g}(x_2)=\mu_{f_B\to x_2}(x_2)\cdot\mu_{f_C\to x_2}(x_2)$$

$$\tilde{g}(x_3)=\mu_{f_C\to x_3}(x_3)\cdot\mu_{f_D\to x_3}(x_3)\cdot\mu_{f_E\to x_3}(x_3)$$

$$\tilde{g}(x_4)=\mu_{f_D\to x_4}(x_4)$$

$$\tilde{g}(x_5)=\mu_{f_E\to x_5}(x_5)$$

值得一提的是：如果因子图是无环的，则一定可以准确的求出任意一个变量的边缘分布，如果是有环的，则无法用**sum-product**算法准确求出来边缘分布。

Sum-Product算法

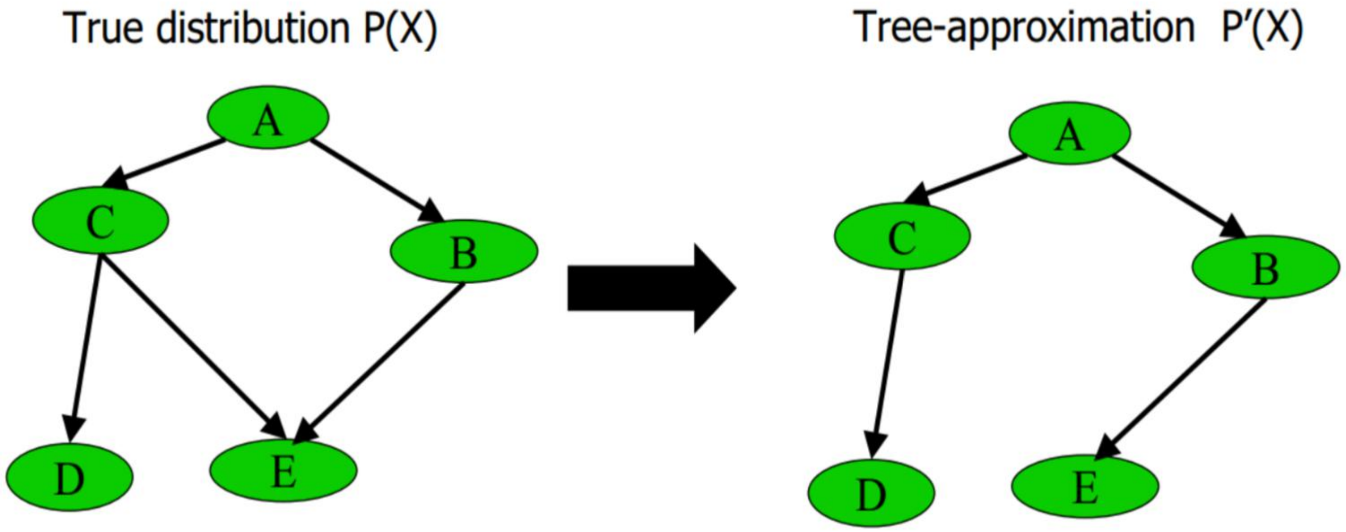
- 从计算来看，**Sum-Product**算法是将计算需要的中间过程进行了保存。如果计算多个概率分布，往往更有效。
- **Sum-Product**算法有点类似动态规划的思想：将一个概率分布写成两个因子的乘积，而这两个因子可以继续分解或者通过已知得到。

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
  - 选择**loopy belief propagation**算法

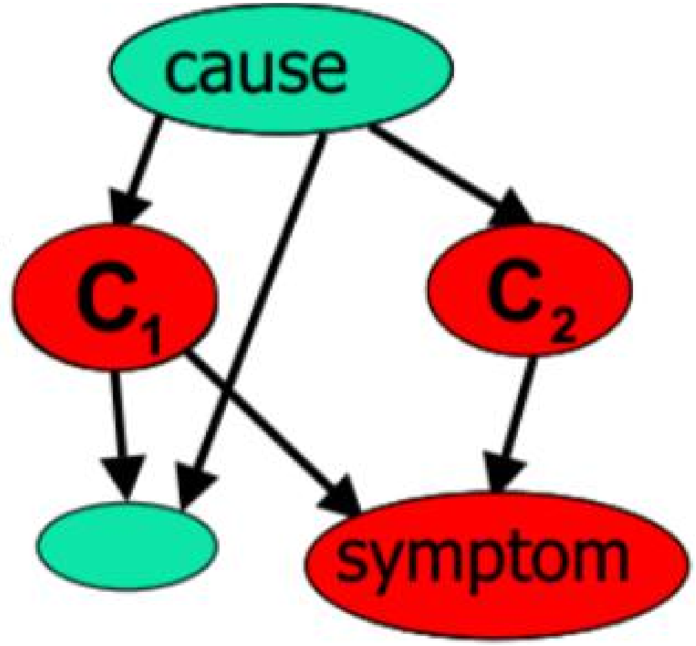
❖ **sum-product**
❖ **max-product**

原贝叶斯网络的近似树结构



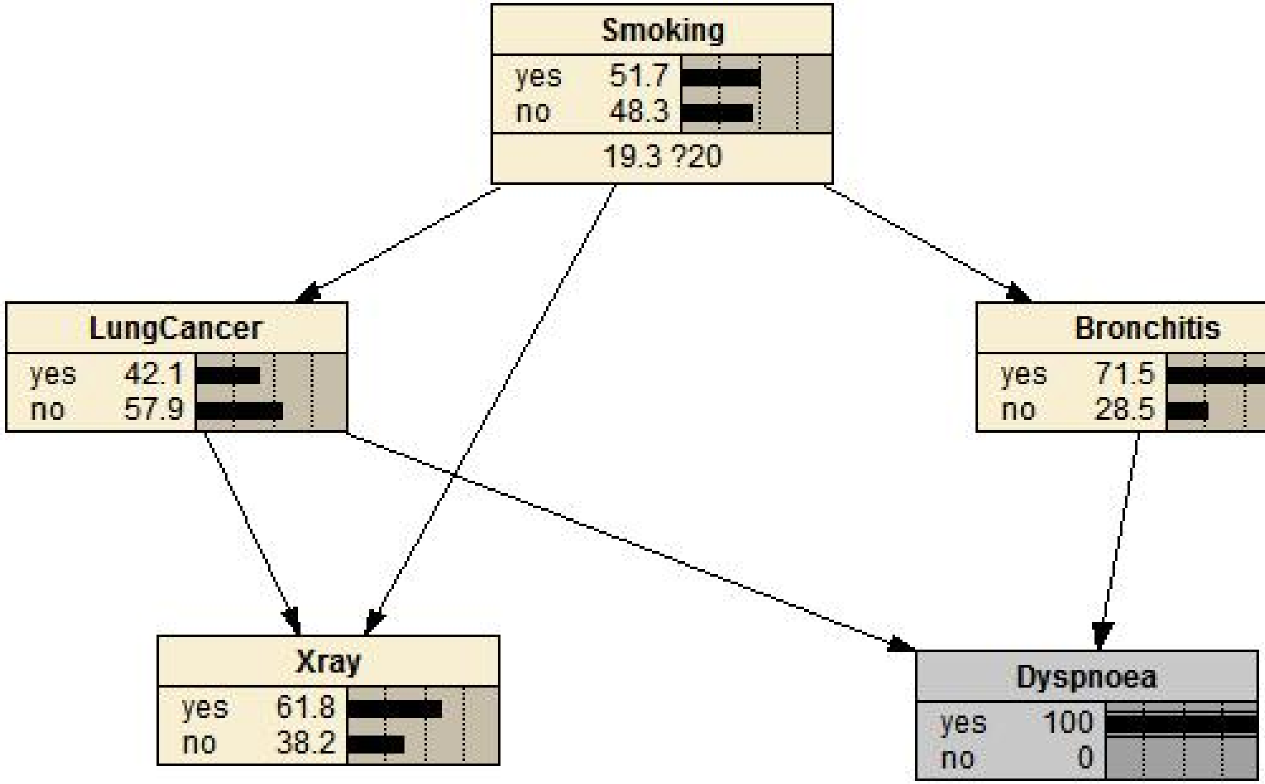
贝叶斯网络用途

- 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。

Netica


Smoking
yes 51.7
no 48.3
19.3 ?20
LungCancer
yes 42.1
no 57.9
Bronchitis
yes 71.5
no 28.5
Xray
yes 61.8
no 38.2
Dyspnoea
yes 100
no 0

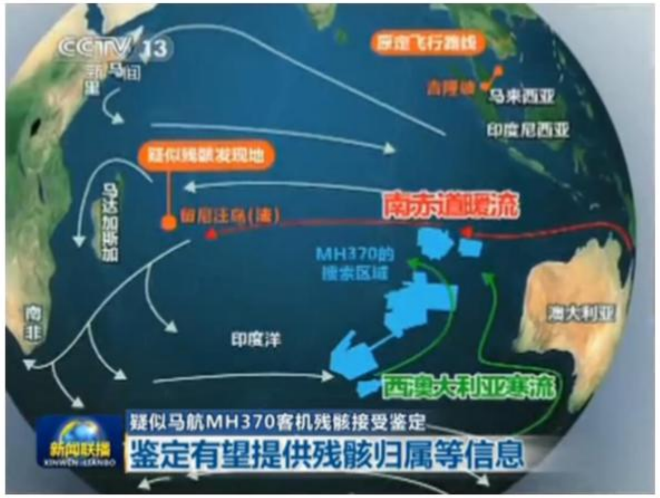
马尔科夫模型模拟实验:

寻找马航MH370

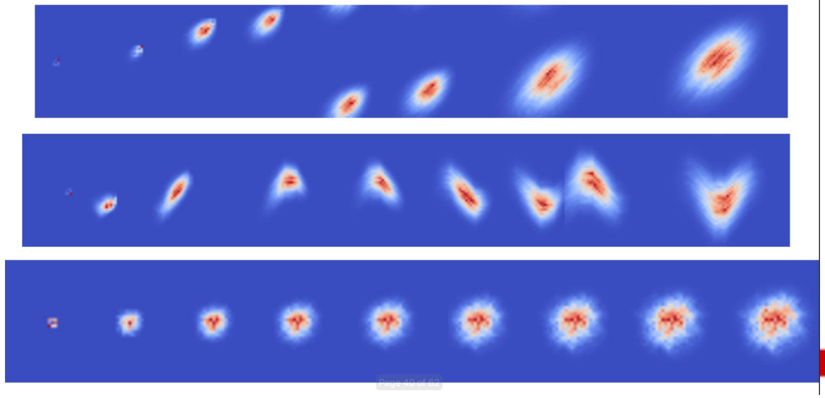
- 2014年3月8日，马来西亚航空公司MH370航班(波音777-200ER)客机凌晨0:41分从吉隆坡飞往北京；凌晨1:19分，马航MH370与空管失去联系。凌晨2:14分飞机最后一次出现在军事雷达上之后人间消失。
- 2015年7月29日在法属留尼汪岛(l'île de la Reunion)发现襟副翼残骸；2015年8月6日，马来西亚宣布，该残骸确属马航MH370。随后法国谨慎宣布，“有很强的理由推测认为，…残骸属于马航MH370航班的波音777客机…但最终的比对结果还需要进一步的技术验证加以确认。”

MH370最后消失区域

- 可否根据雷达最后消失区域和洋流、大气等因素：
- 判断留尼汪岛是否位于可能区域？
- 残骸漂流到该岛屿的概率有多大？



- 概率优势方向：Direct/Sin/Random



- 在每个时刻，物体的当前可能区域是上一时刻所有可能区域和相应转移概率的乘积和，这恰好是矩阵乘法(矩阵和向量乘法)的定义。
- 当前可能区域只和上一个时刻的区域有关，而与更上一个时刻无关，因此，是马尔科夫模型。
- 思考：可以使用“漂流位置”建立马尔科夫模型，该可能位置是不可观察的，而将“转移位置”认为是“漂流位置”的转换结果，“转移位置”是残骸的最终真实位置，使用增强的隐马尔科夫模型。
  - 不要过得累加模型的复杂度，适时使用奥卡姆剃刀(Occam‘s Razor)。
  - 该模型仅个人观点。

Reference

- 因子图与和积算法 **https://blog.csdn.net/sinat_38151275/article/details/83621805**
- 北京**10**月机器学习班第**9**次课，邹博讲贝叶斯网络的**PPT**：**http://pan.baidu.com/s/1o69Lp1K**；
- 从贝叶斯方法谈到贝叶斯网络**: https://blog.csdn.net/zdy0_2004/article/details/41096141**

Thanks & Questions