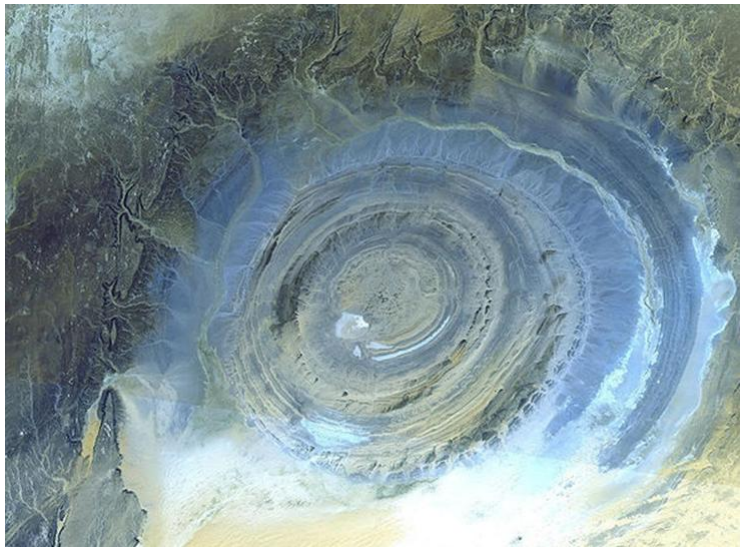


补充：条件独立性和贝叶斯网络

主讲：王　星　2021年5月7日
办公电话:86-10-82500167
电子邮箱:wangxingwisdom@126.com

贝叶斯分类

- 贝叶斯Thomas Bayes(约1701-1761)
  - 英国数学家。约1701年出生于伦敦，做过神甫。1742年成为英国皇家学会会员。1761年4月7日逝世。贝叶斯在数学方面主要研究概率论。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。他死后，理查德·普莱斯(Richard Price)于1763年将他的著作《机会问题的解法》(An essay towards solving a problem in the doctrine of chances)寄给了英国皇家学会，对于现代概率论和数理统计产生了重要的影响

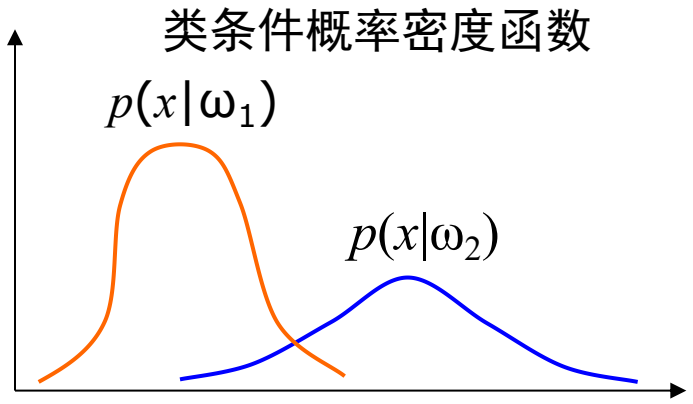
后验　似然　先验

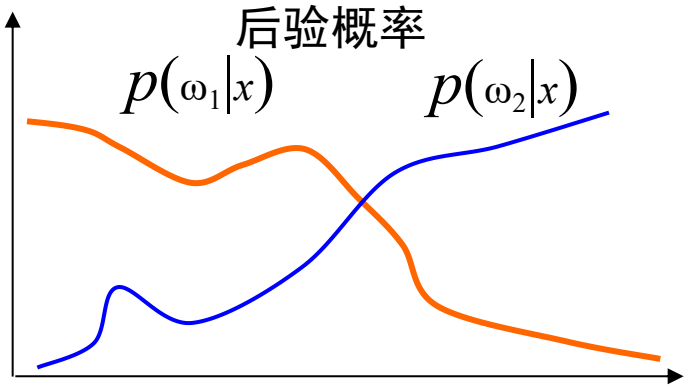
$$P(\omega_j \mid x) = P(x \mid \omega_j) \cdot P(\omega_j) / P(x)$$

证据

贝叶斯的推断

- 贝叶斯决策就是在信息不完整的情况下，对部分未知的状态用主观概率（先验概率）估计，然后用贝叶斯公式对发生概率进行修正，最后再利用期望值和修正概率做出最优决策
- 贝叶斯决策理论方法是统计模型决策中的一个基本方法，其基本思想是：
  - 已知类条件概率密度参数表达式和先验概率
  - 利用贝叶斯公式转换成后验概率
  - 根据后验概率大小进行决策分类





朴素Bayes分类模型

- 输入空间：$\mathcal{X} \subseteq \mathrm{R}^n$，输出空间为类标记集合：$\mathcal{Y} = \{c_1, c_1, \ldots, c_k\}$
- 学习联合概率分布$P(X, Y)$
  - 属于生成模型
- 训练数据集 $T = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$，由联合概率分布独立同分布产生
- 朴素贝叶斯通过训练数据集学习联合概率分布P(X,Y)
  - 先验概率分布：$P(y = c_k), k = 1,2, \ldots, K$
  - 条件概率分布：$P(X = x|Y = c_k) = P\left(X^{(1)} = x^{(1)}, \ldots, X^{(n)} = x^{(n)} \middle| Y = c_k\right), k = 1,2, \ldots, K$
    - 类条件概率，参数个数多，估算计算量大（连乘）
- 朴素贝叶斯分类器(将后验概率最大的类作为x的类输出)
  - $y = f(x) = \arg\max_{c_k} P(Y = c_k|X = x)$

直接估计的难题

联合概率：$P(X_1, X_2, \ldots, X_N)$

二值，则有2^N可能的值，其中2^{N-1}个独立。不是二值的话，NP问题？

如果相互独立：

$$P(X_1, X_2, \ldots, X_N) = P(X_1)\, P(X_2) \ldots P(X_N)$$

条件概率：

$$P(X_1, X_2, \ldots, X_N) = P(X_1 | X_2, \ldots, X_N)\, P(X_2, \ldots, X_N)$$

迭代表示：

$$P(X_1, X_2, \ldots, X_N) = P(X_1)\, P(X_2 | X_1)\, P(X_3 | X_2 X_1) \ldots P(X_N | X_{N-1}, \ldots, X_1)$$

$$= P(X_N)\, P(X_{N-1} | X_N)\, P(X_{N-2} | X_{N-1} X_N) \ldots P(X_1 | X_2, \ldots, X_N)$$

.

- 独立：如果X与Y相互独立，则

$$P(X,Y) = P(X)P(Y)$$

$$P(X|Y) = P(X)$$

- 条件独立：如果在给定Z的条件下，X与Y相互独立，则

$$P(X|Y, Z) = P(X|Z)$$

实际中，条件独立比完全独立更普遍

Conditional Independence

Definition: X is conditionally independent of Y given Z, if the probability distribution governing X is independent of the value of Y, given the value of Z

$$(\forall i, j, k) P(X = x_i | Y = y_j, Z = z_k) = P(X = x_i | Z = z_k)$$

Which we often write

$$P(X|Y, Z) = P(X|Z)$$

E.g.,

$$P(Thunder|Rain, Lightning) = P(Thunder|Lightning)$$

Naïve Bayes uses assumption that the X_i are conditionally independent, given Y

then:

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

$$P(X_1 \ldots X_n|Y) = \prod_i P(X_i|Y)$$

How many parameters needed now for P(X|Y)? P(Y)?

$$\theta_{ij} \equiv P(X = x_i|Y = y_j) \qquad \pi_j \equiv P(Y = y_j)$$

Naïve Bayes assumes

X=$\langle X_1, \ldots, X_n \rangle$, Y discrete-valued

$$P(X_1 \ldots X_n|Y) = \prod_i P(X_i|Y)$$

i.e., that X_i and X_j are conditionally independent given Y, for all $i \neq j$

朴素bayes方法

- “朴素”贝叶斯：条件独立性假设（特征的各个分量独立）
  - $P(X=x|Y=c_k)=P\left(X^{(1)}=x^{(1)},\dots,X^{(n)}=x^{(n)}\middle|Y=c_k\right)=$
    $\prod_{j=1}^{n}P(X^{(j)}=x^{(j)}|Y=c_k)$

- 对给定的输入X，通过学习到的模型计算后验概率分布$P(X=x|Y=c_k)$
- 将后验概率最大的类作为x的类输出

$$P(Y=c_k|X=x)=\frac{P(X=x,Y=c_k)}{P(X=x)}=\frac{P(X=x|Y=c_k)P(Y=c_k)}{\sum_k P(X=x|Y=c_k)P(Y=c_k)}=\frac{P(Y=c_k)\prod_{j=1}^{n}P(X^{(j)}=x^{(j)}|Y=c_k)}{\sum_k P(Y=c_k)\prod_{j=1}^{n}P(X^{(j)}=x^{(j)}|Y=c_k)}$$

  - 后验概率转先验概率的计算

$$y=f(x)=\arg\max_{c_k}P(Y=c_k|X=x)$$

$$y=f(x)=\arg\max_{c_k}P(Y=c_k|X=x)=\arg\max_{c_k}\frac{P(Y=c_k)\prod_{j=1}^{n}P(X^{(j)}=x^{(j)}|Y=c_k)}{\sum_k P(Y=c_k)\prod_{j=1}^{n}P(X^{(j)}=x^{(j)}|Y=c_k)}$$

后验概率最大化的

朴素贝叶斯法将实例分到后验概率最大的类中，等价于期望风险最小化（结论）

假设选择0-1损失函数：$L(Y,f(X))=\begin{cases}1, Y\neq f(X)\\ 0, Y=f(X)\end{cases}$，$f(X)$为决策函数

期望风险函数：

$$R_{exp}(f)=E[L(Y,f(X))]=E_X\left(\sum_Y L()\,P(Y|X)\right)=E_X\sum_{k=1}^{K}[L(c_k,f(X))P(c_k|X)]$$

$R_{exp}(f)$最小化：对$E_X()$中的$\sum_{k=1}^{K}[L(c_k,f(X))P(c_k|X)]$的每一个$X=x$都最小化

$$f(x)=\arg\min_{y\in\mathcal{Y}}\sum_{k=1}^{K}[L(c_k,y)P(c_k|X=x)]=\arg\min_{y\in\mathcal{Y}}\sum_{k=1}^{K}P(y\neq c_k|X=x)$$

$$=\arg\min_{y\in\mathcal{Y}}(1-P(y=c_k|X=x))=\arg\max_{y\in\mathcal{Y}}P(y=c_k|X=x)$$

即最大化后验概率

Maximum likelihood estimates:

$$\hat{\pi}_k = \hat{P}(Y = y_k) = \frac{\#D\{Y = y_k\}}{|D|}$$

$$\hat{\theta}_{ijk} = \hat{P}(X_i = x_{ij}|Y = y_k) = \frac{\#D\{X_i = x_{ij} \wedge Y = y_k\}}{\#D\{Y = y_k\}}$$

MAP estimates (uniform Dirichlet priors):

$$\hat{\pi}_k = \hat{P}(Y = y_k) = \frac{\#D\{Y = y_k\} + l}{|D| + lR}$$

$$\hat{\theta}_{ijk} = \hat{P}(X_i = x_{ij}|Y = y_k) = \frac{\#D\{X_i = x_{ij} \wedge Y = y_k\} + l}{\#D\{Y = y_k\} + lM}$$

- 平滑参数l，R表示Y的类别数，M表示的X变量的取值数To account for estimation from small samples, probability estimates are adjusted or *smoothed*.
- Laplace smoothing using an *m*-estimate assumes that each feature is given a prior probability, *p*, that is assumed to have been previously observed in a “virtual” sample of size *m*.

在 Laplace 估计中，先验概率 P(Y)被定义如下：

$$p(Y = y_j) = \frac{n_c + k}{N + n*k}$$

其中，n_c 是满足 $Y = \{y_j\}$ 的实例个数，N 是训练集个数，n 是类的个数，并且 k=1。

0-1竞争模型例子：

- $X_1 \ldots X_n$服从两点分布，概率θ，
- 则 $\sum x_i = r$ 服从二项分布

- 求θ的估计 $p(r \mid \theta) = \binom{n}{r} \theta^r (1-\theta)^{n-r}$
- 设θ先验分布是beta(a,b)
- 求得θ后验分布:beta(a+r,b+n-r)
- 求得E(θ|r)=(a+r)/(a+b+n)

Dirichlet priors

7. Dirichlet 分布

Dirichlet 分布在贝叶斯分析中可以作为多项分布参数的共轭先验分布。

定义 5.2.7 若随机变量 T 有密度函数

$$f(t_1,\cdots,t_k|\boldsymbol{\alpha})=\frac{1}{\beta(\boldsymbol{\alpha})}\prod_{i=1}^{k}t_i^{\alpha_i-1},\quad t_i>0,\quad \sum_{i=1}^{k}t_i=1,$$

其中 k 维参数 $\boldsymbol{\alpha}>0$，而

$$\beta(\boldsymbol{\alpha})=\frac{\prod_{i=1}^{k}\Gamma(\alpha_i)}{\Gamma\left(\sum_{i=1}^{k}\alpha_i\right)},$$

则我们称 T 具有参数为 $\boldsymbol{\alpha}$ 的 Dirichlet 分布，记作 $T\sim$ Dirichlet($\boldsymbol{\alpha}$)。$k=2$ 时的 Dirichlet 分布称作 Beta 分布。

Dirichlet 分布的密度函数表示在已知 k 个竞争事件已经出现了 α_i-1 次的条件下，它们出现的概率为 t_i 的信念。

Dirichlet 分布有如下数字特征：

$$ET=\frac{\alpha_i}{\alpha_0},\quad VarT=\frac{\alpha_i(\alpha_0-\alpha_i)}{\alpha_0^2(\alpha_0+1)},\quad Cov(T_i,T_j)=-\frac{\alpha_0\alpha_i}{\alpha_0^2(\alpha_0+1)},$$

其中

$$\alpha_0=\sum_{i=1}^{k}\alpha_i\ 。$$

- Learning (generative) classifiers based on Bayes rule
- Conditional independence
  - What it is
  - Why it's important
- Naïve Bayes assumption and its consequences
- Naïve Bayes with discrete inputs, continuous inputs

Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

P(p) = 9/14
P(n) = 5/14

outlook	
P(sunny|p) = 2/9	**P(sunny|n) = 3/5**
P(overcast|p) = 4/9	**P(overcast|n) = 0**
P(rain|p) = 3/9	**P(rain|n) = 2/5**
temperature	
P(hot|p) = 2/9	**P(hot|n) = 2/5**
P(mild|p) = 4/9	**P(mild|n) = 2/5**
P(cool|p) = 3/9	**P(cool|n) = 1/5**
humidity	
P(high|p) = 3/9	**P(high|n) = 4/5**
P(normal|p) = 6/9	**P(normal|n) = 1/5**
windy	
P(true|p) = 3/9	**P(true|n) = 3/5**
P(false|p) = 6/9	**P(false|n) = 2/5**

文本中Naïve-Bayes的用法

Text	Category
A great game（一个伟大的比赛）	Sports（体育运动）
The election was over（选举结束）	Not sports（不是体育运动）
Very clean match（没内幕的比赛）	Sports（体育运动）
A clean but forgettable game （一场难以忘记的比赛）	Sports（体育运动）
It was a close election （这是一场势均力敌的选举）	Not sports（不是体育运动）

我们遇到了一个问题：“close”不会出现在任何sports样本中！那就是说P（close | Sports）= 0。这是相当不方便的，因为我们将把它与其他概率相乘，所以我们最终会得到P(a|Sports)×P(very|Sports)×0×P(game|Sports)等于0。这样做的事情根本不会给我们任何信息，所以我们必须找到一个办法。

我们该怎么做呢？通过使用一种被称为拉普拉斯平滑的方法：我们为每个计数添加1，所以它不会为零。为了平衡这一点，我们将可能的词数加到除数，因此这部分将永远不会大于1。在我们的案例中，可能的话是 [“a” ， “great” ， “very” ， “over” ， 'it' ， 'but' ， 'game' ， 'election' ， 'close' ， 'clean' ， 'the' ， 'was' ， 'forgettable' ， 'match'] 。
由于可能的单词数是14，应用拉普拉斯平滑我们得到了。全部结果如下：

$$P(game|sports) = \frac{2+1}{11+14}。$$

Word	P(word | Sports)	P(word | Not Sports)
a	$\frac{2+1}{11+14}$	$\frac{1+1}{9+14}$
very	$\frac{1+1}{11+14}$	$\frac{0+1}{9+14}$
close	$\frac{0+1}{11+14}$	$\frac{1+1}{9+14}$
game	$\frac{2+1}{11+14}$	$\frac{0+1}{9+14}$

朴素贝叶斯分类的优点和缺点

- 优点
  - 计算速度比较快
  - 规则清楚易懂
  - 独立事件假设，大多数问题不至于发生太大偏差
- 不足：
  - 只能用于类别变量
  - 只能用于分类
  - 假设变量之间独立互不影响，使用时需要谨慎分析变量之间的相关性。

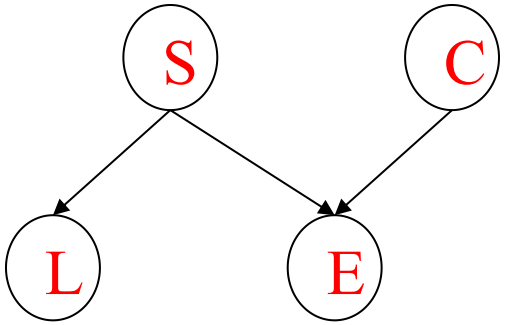
第三节 Bayes 网络

- 1988年由Pearl提出, 贝叶斯网络（Bayes Network）成功地应用于知识发现领域，成为表示不确定性知识和推理的一种流行的方法。基于贝叶斯方法的贝叶斯网络是一种适应性很广的手段和工具，具有坚实的数学理论基础。在综合先验信息（领域知识）和数据样本信息的前提下，还可避免只使用先验信息可能带来的主观偏见。虽然很多贝叶斯网络涉及的学习问题是NP难解的。但是，由于已经有了一些成熟的近似解法，加上一些限制后计算可大为简化，很多问题可以利用近似解法求解。
- 研究变量和变量之间关系的重要方法。
- 是图论与概率论的结合。
- 贝叶斯网络又称信度网络，是Bayes方法的扩展，不确定知识表达和推理领域最有效的理论模型之一。

贝叶斯网络（因果关系网络）

假设：

- 命题S(smoker)：该患者是一个吸烟者
- 命题C(coal Miner)：该患者是一个煤矿矿井工人
- 命题L(lung Cancer)：他患了肺癌
- 命题E(emphysema)：他患了肺气肿
- 由专家给定的假设可知：
  - 命题S对命题L和命题E有因果影响，
  - 而C对E也有因果影响。
  - 命题之间的关系可以描绘成因果关系网。每一个节点代表一个证据，每一条弧代表一条规则（假设），连接结点的弧表达了有规则给出的节点间的直接因果关系。其中，节点S，C是节点L和E的父节点或称双亲节点，同时，L，E也称为是S和C的子节点或称后代节点。



贝叶斯网络的由来

- 全联合概率计算复杂性十分巨大
- 朴素贝叶斯太过简单
- 现实需要一种自然、有效的方式来捕捉和推理——不确定性知识
- 变量之间的独立性和条件独立性可大大减少为了定义全联合概率分布所需的概率数目

贝叶斯网络的定义

一个 Bayes 网络是一个二元组 $B = \langle G, \Theta \rangle$，其中

1) $G = \{\Pi_1, \Pi_2, \cdots, \Pi_N\}$ 是一个有向无环图(DAG)，其结点为

$$U = \{X_1, X_2, \cdots, X_N\}, N \geqslant 1,$$

$\Pi_i \subset 2^U, i = 1, \cdots, N$ 是结点 X_i 的父结点集合；

2) $\Theta = \{P(X_i \mid \Pi_i) \mid X_i \in U, i = 1, \cdots, N\}$ 是一组条件概率的集合，称为网络参数.

一个 Bayes 网络 $B = \langle G, \Theta \rangle$ 确定了一个概率空间 $\langle \Omega, F, P \rangle$，其中

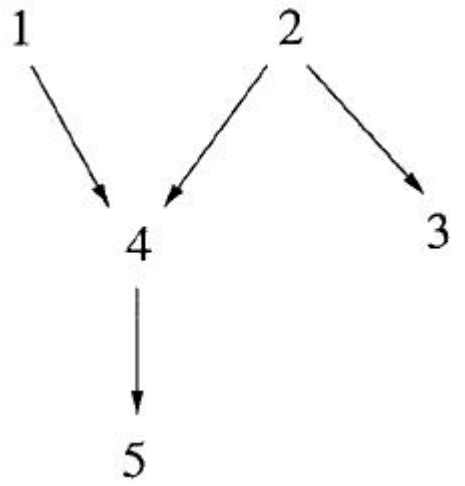
$$P(U) = \prod_{i=1}^{N} P(X_i \mid \Pi_i). \qquad (1)$$

根据 Bayes 网络的定义，从数据样本中学习 Bayes 网络，就是要学习网络模型的结构 G 和参数

- 有向无环图(**Directed Acyclic Graph, DAG**)
- 随机变量集组成网络节点，变量可离散或连续
- 一个连接节点对的有向边或箭头集合

■条件独立：贝叶斯网络中的一个结点，如果父母结点已知，则它条件独立于它的所有非后代结点

■每个节点*Xi*都有一个条件概率分布表：*P(Xi|Parents(Xi))*，每个节点受其他节点的影响都可以转化为只受其父节点对该节点的影响



$$p(\boldsymbol{X}) = p(X_5 \mid X_4)\, p(X_4 \mid X_1, X_2)\, p(X_3 \mid X_2)\, p(X_2)\, p(X_1)$$

给出一个图 G，观测D的似然函数可以表示为：

$$P(D \mid G) = \prod_{i=1}^{n} p\left(X_i \mid parents\ (X_i)\right)$$

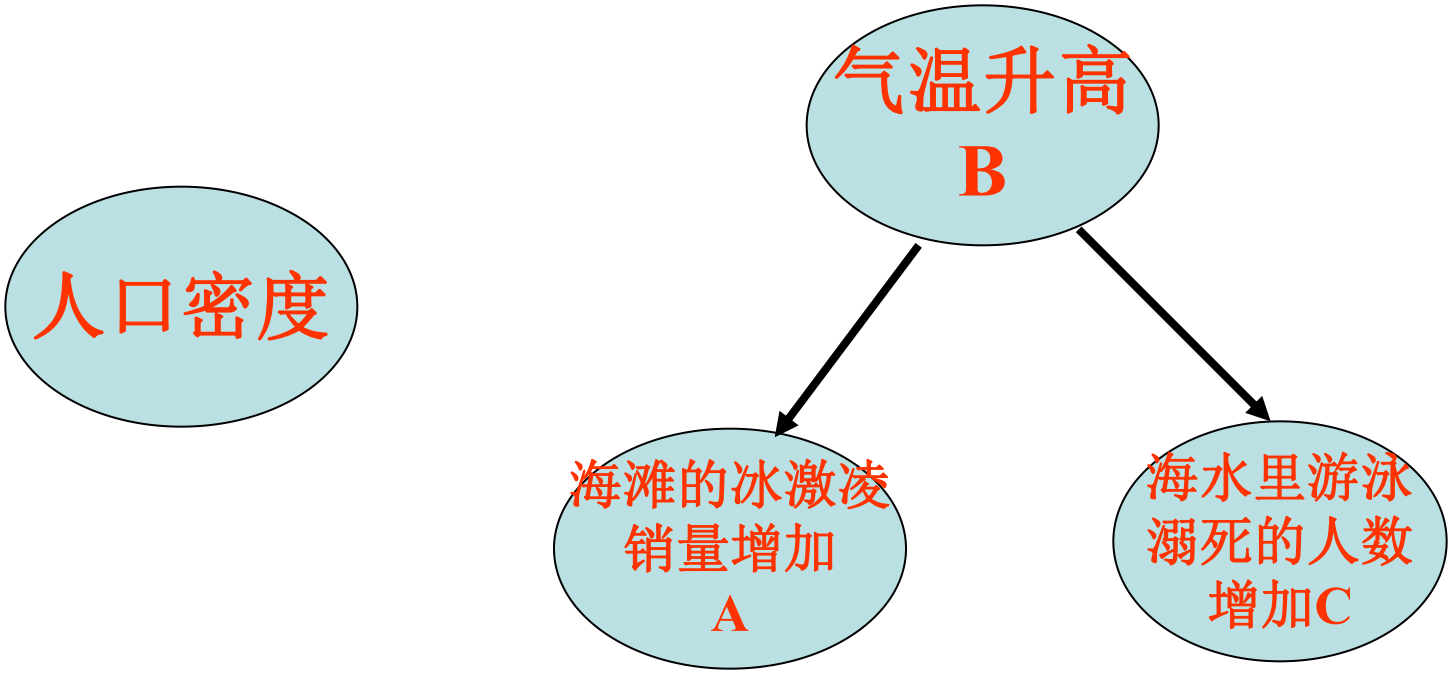
贝叶斯网络的其他名称

- 信念网(Belief Network)
- 概率网络(Probability Network)
- 因果网络(Causal Network)
- 知识图(Knowledge Map)
- 图模型(Graphical Model)或概率图模型(PGM)
- 决策网络(Decision Network)
- 影响图(Influence Diagram)

贝叶斯网络的应用

- 医疗诊断
- 故障诊断
- 信息检索
- 规划学习
- 序列分类

独立和条件独立性



- 没有边的两个节点之间互相独立，例如： Population和其它3个变量相互独立
- 如果B隔离了A和C时，那么可以认为A与C是关于B条件独立的
给定气温升高后，冰激凌销量和海水里溺死的人数条件独立

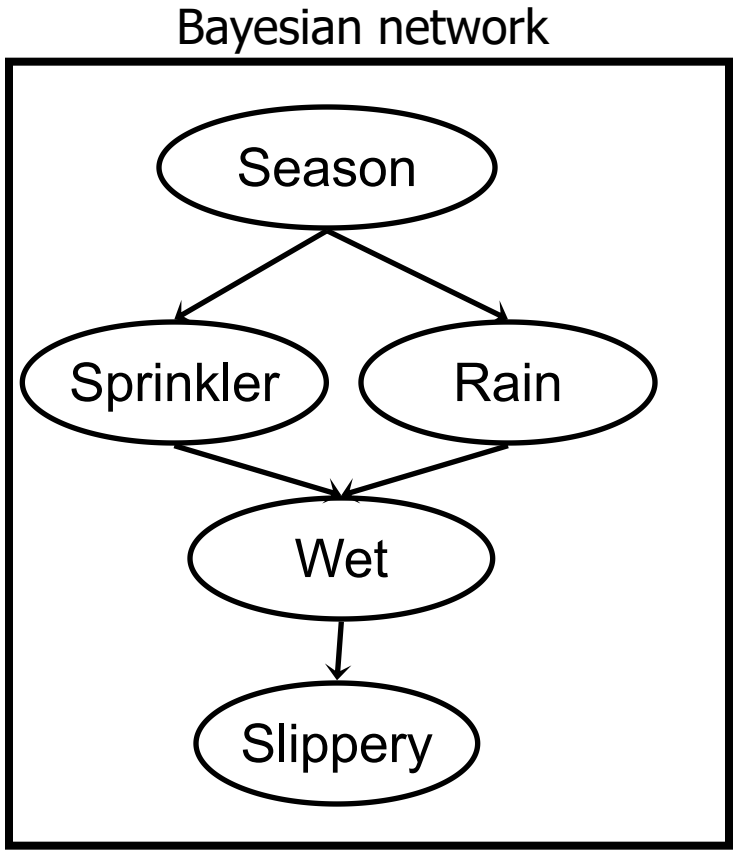
Bayesian Network的应用

We want to describe the causal relationship between the following events:

1) The season
2) Whether it is raining outside
3) The sprinkler is on
4) The sidewalk is wet
5) The sidewalk is slippery

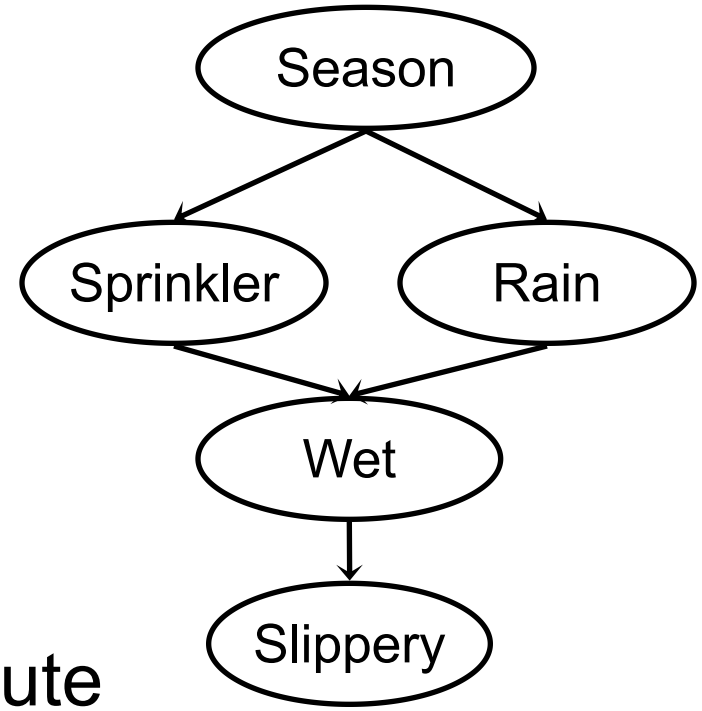
假设:

- “Sprinkler on” and “Rain” are determined by “Season”
- “Sidewalk wet” is determined by “Sprinkler on” and “Rain”
- “Sidewalk slippery” is determined by “Sidewalk wet”



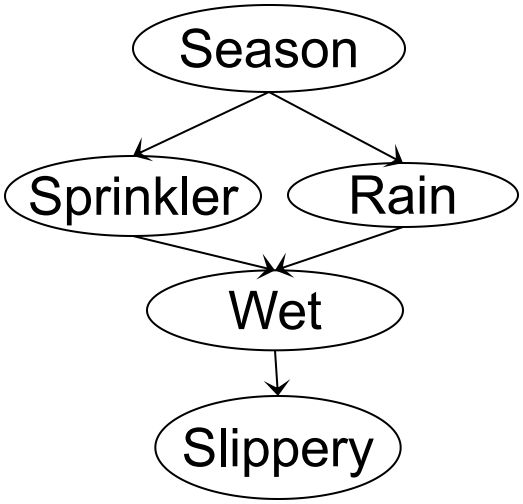
- 每个节点代表一个随机变量，这里这个随机变量是一个二值变量，所以代表一个随机事件是否发生。

Properties of Bayesian Networks



- Links are *not* absolute
  - If the sprinkler is on, this does not always mean that the sidewalk is wet
  - For example, the sprinkler may be aimed away from the sidewalk

Properties of Bayesian Networks



- Given that the sidewalk is wet, we can calculate the probability that the sprinkler is on:

  P(sprinkler on | sidewalk wet)

- Bayesian networks allow us to calculate such values from a small set of probabilities in a process called reasoning or Bayesian Inference

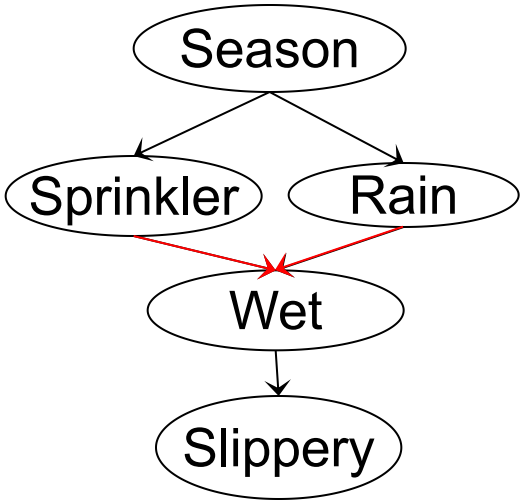
Bayesian Networks
三种常用的推理方式

- Bayesian networks通过父子传递可以在任何一个方向上推理

1) If the sprinkler is on, the sidewalk is probably wet (prediction预测)

2) If the sidewalk is wet, it is more likely that the sprinkler is on or it is raining (diagnosis诊断)

3) If the sidewalk is wet and the sprinkler is on, the likelihood that it is raining is reduced (explaining away解释远离，消解影响)



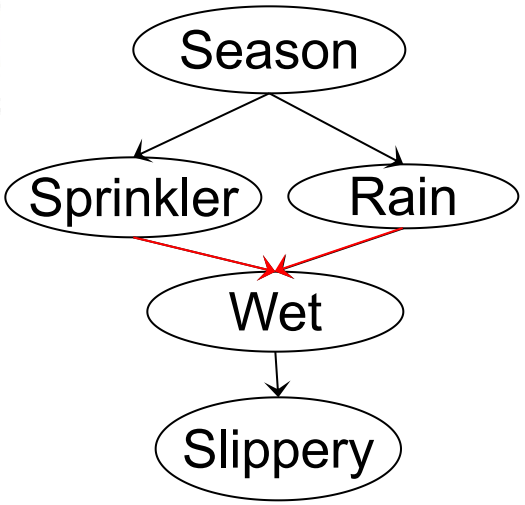
- Explaining away is a special type of reasoning that is especially difficult to model in other network models

解释远离

$$P(r|w,s) = \frac{P(w|r,s)P(r|s)}{P(w|s)}$$

如果 $P(w|s) = P(w|r,s)$

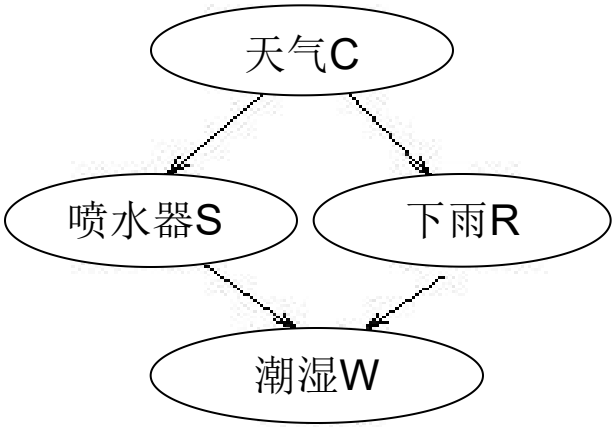
$$P(r|w,s) = P(r|s) = P(r)$$



这说明*S*决定了*W*的发生，*R*则发生了对*W*的解释远离

Bayesian Network的分布表

P(C=F)	P(C=T)
0.5	0.5

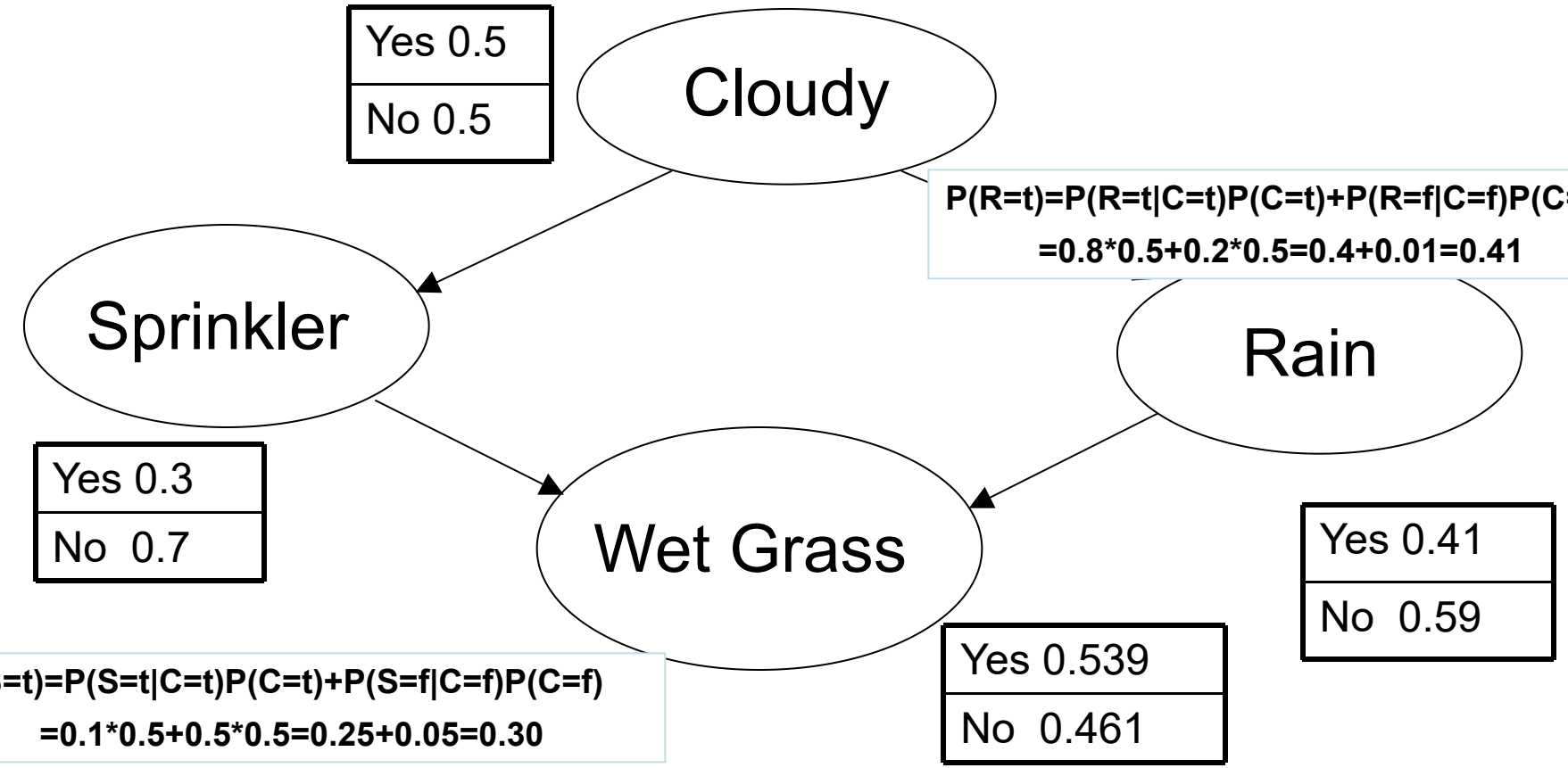


C	P(S=F)	P(S=T)
F	0.5	0.5
T	0.9	0.1

C	P(R=F)	P(R=T)
F	0.8	0.2
T	0.2	0.8

S	R	P(W=F)	P(W=T)
F	F	1.0	0.0
T	F	0.1	0.9
F	T	0.1	0.9
T	T	0.01	0.99

Bayesian Network

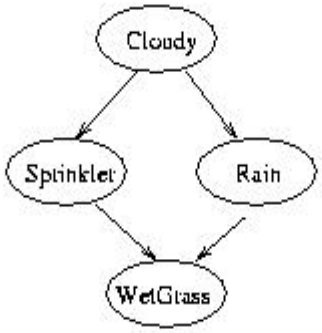


P(W=t)=P(W=t|S=t,R=t)P(S=t)P(R=t)+P(W=t|S=t,R=f)P(S=t)P(R=f)
+P(W=t|S=f,R=t)P(S=f)P(R=t)+P(W=t|S=f,R=f)P(S=f)P(R=f)
=0.99*0.41*0.30+0.9*(1-0.41)*0.30+0.9*0.41*(1-0.30)+0.0*(1-0.30)*(1-0.41)
= 0.53937

Diagnosis

We observe the grass is wet- 2 causes sprinkler or rain .. Which is more likely ???

P(C=F)	P(C=T)
0.5	0.5



C	P(S=F)	P(S=T)
F	0.5	0.5
T	0.9	0.1

C	P(R=F)	P(R=T)
F	0.8	0.2
T	0.2	0.8

S	R	P(W=F)	P(W=T)
F	F	1.0	0.0
T	F	0.1	0.9
F	T	0.1	0.9
T	T	0.01	0.99

$\Pr(S=1|W=1) = \Sigma \Pr(S=1, W=1) / \Pr(W=1) = 0.2781/0.539$

$\Pr(R=1|W=1) = \Sigma \Pr(R=1, W=1) / \Pr(W=1) = 0.4581/0.539$

Normalizing $\Pr(W=1) = 0.539$

贝叶斯网络的语义

- 贝叶斯网络的两种含义
  - ■对联合概率分布的表示 — 构造网络
  - ■对条件依赖性语句集合的编码 — 设计推理过程

- 贝叶斯网络的语义

$$P(x_1,\ldots, x_n) = P(x_1|parent(x_1)) \ldots P(x_n|parent(x_n))$$

贝叶斯网络建模的方法

贝叶斯网络建模一般有三种方法：

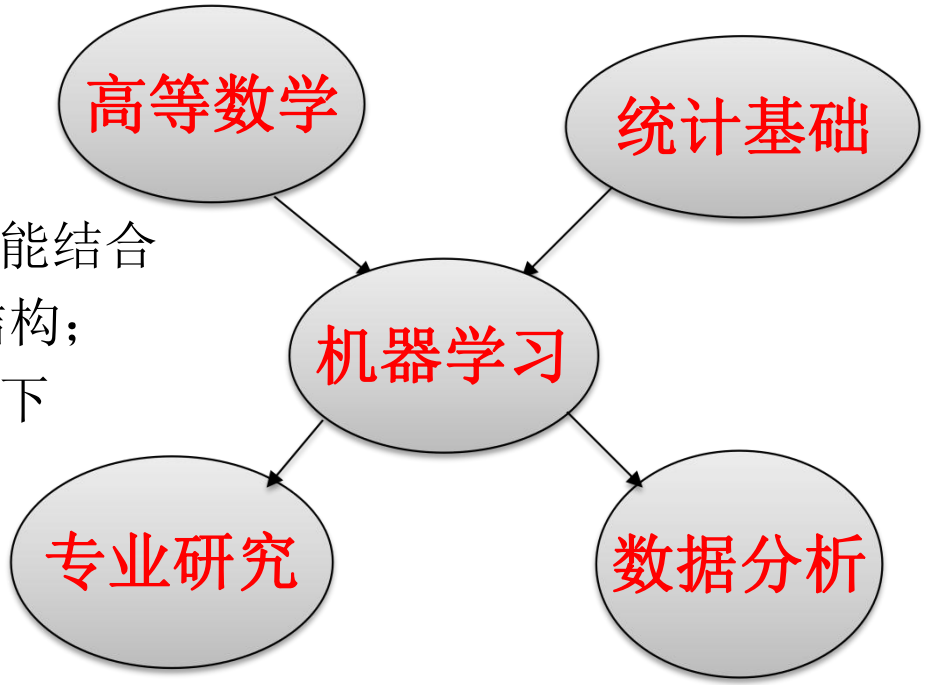
1）依靠专家和背景建模；

2）从数据中学习；

3）从知识库中创建。

贝叶斯网络学习主要研究**结构学习**与**参数学习**两个方面。

1. 结构学习是指利用训练样本集，尽可能结合先验知识，确定合适的贝叶斯网络拓扑结构；
2. 参数学习是指在网络结构确定的情况下从数据中学习每一个节点的条件概率表

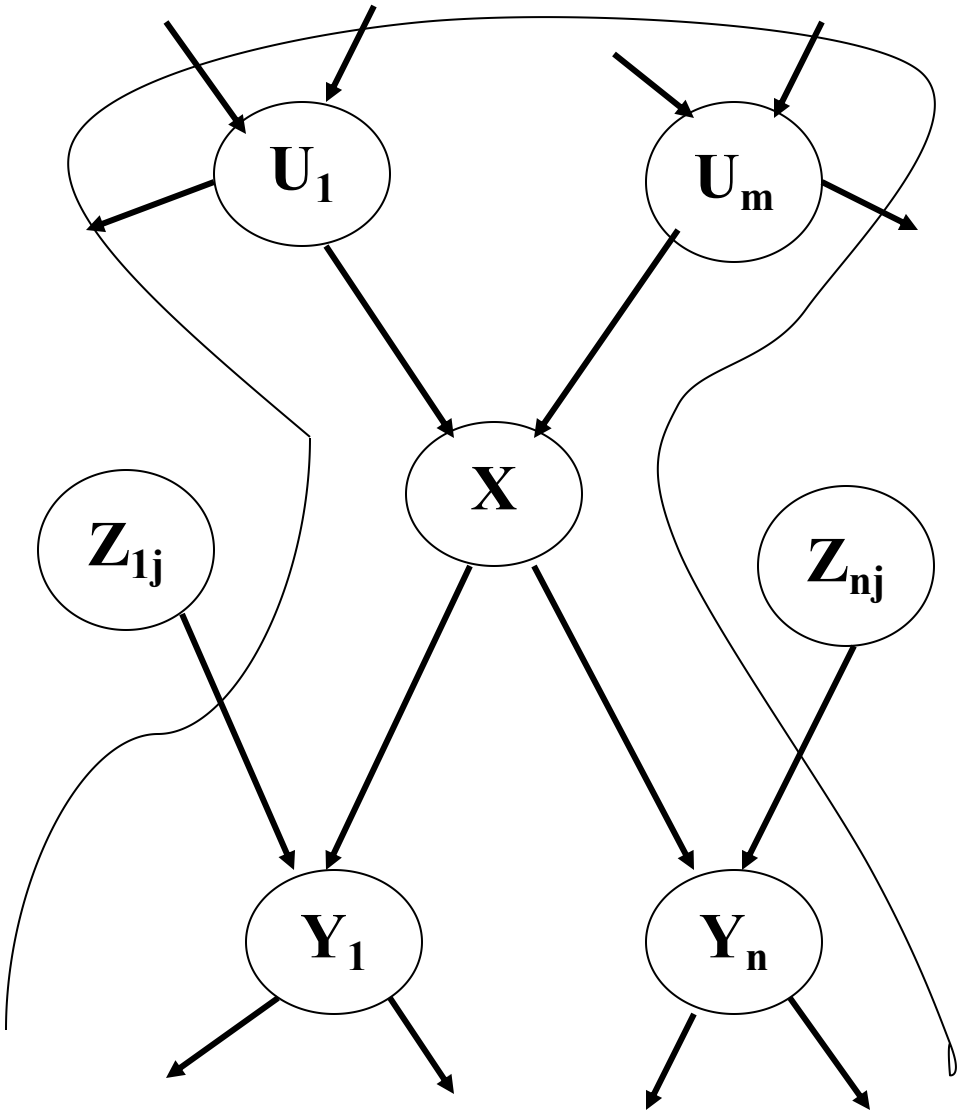


贝叶斯网络的特性：

- 作为对域的一种完备而无冗余的表示，贝叶斯网络比全联合概率分布紧凑得多
- BN的紧凑性是**局部结构化**(Locally structured, 也称**稀疏**, Sparse)系统一个非常普遍特性的实例
- BN中每个节点只与数量有限的其它节点发生**直接的**相互作用
- 假设节点数n=30, 每节点有5个父节点，则BN需$30x2^5$=960个数据，而全联合概率分布需要2^{30}= 10亿个！

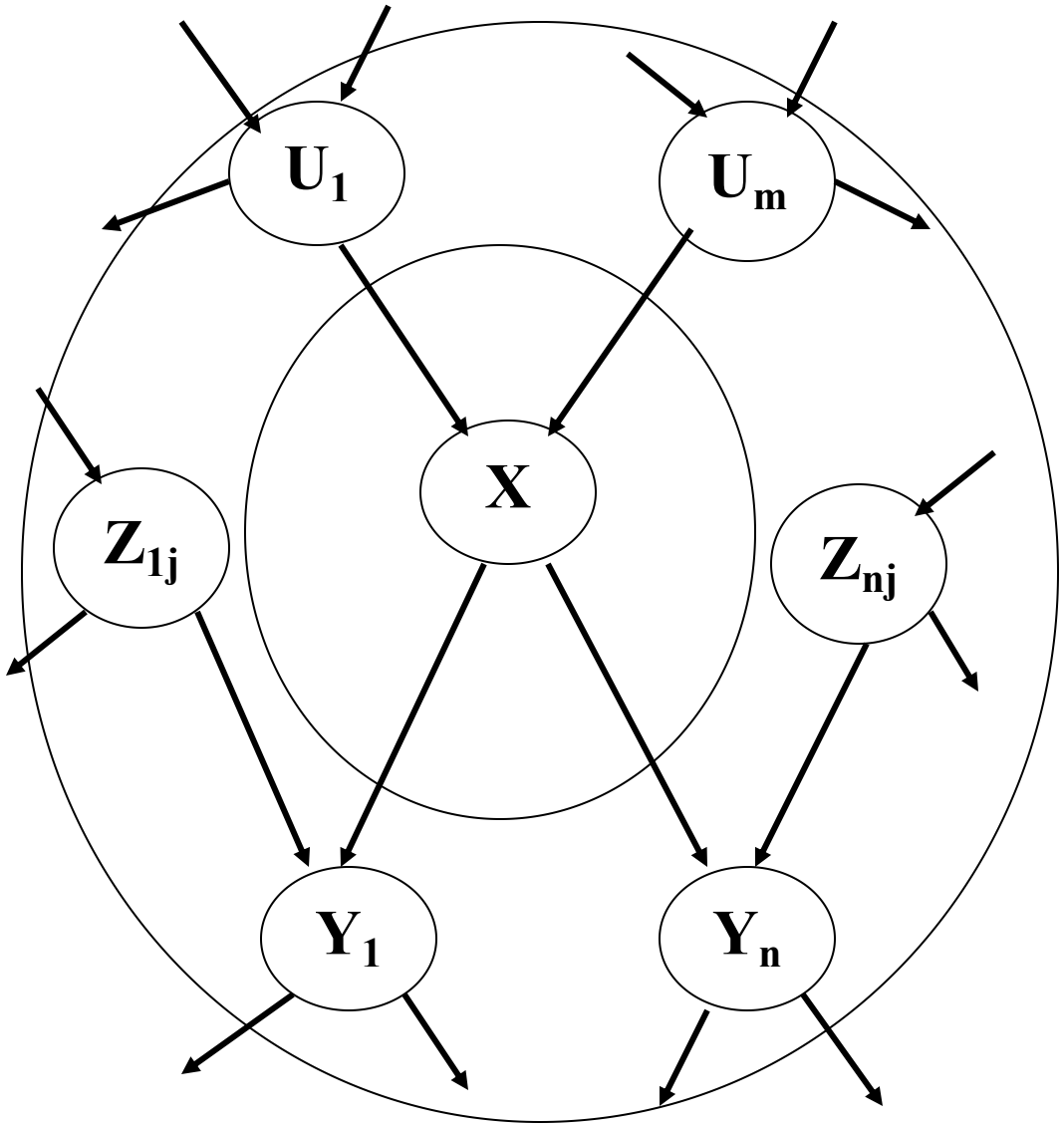
贝叶斯网络中的条件独立关系：

- 给定父节点，一个节点与它的非后代节点是条件独立的
- 给定一个节点的父节点、子节点以及子节点的父节点——马尔可夫覆盖(Markov blanket)，这个节点和网络中的所有其它节点是条件独立的



【说明】：

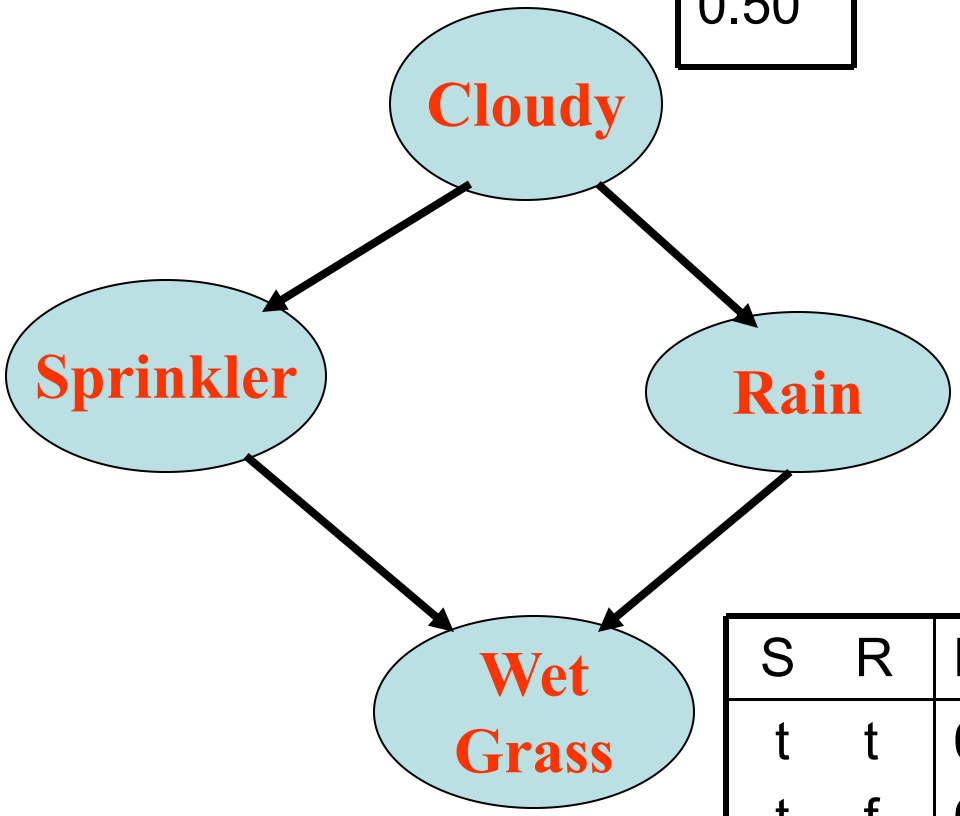
给定节点X的父节点U_1... U_m，节点X与它的非后代节点（即Z_{ij}）是条件独立的。



【说明】：

给定马尔可夫覆盖（两圆圈之间的区域），节点X和网络中所有其它节点都是条件独立的。

多连通网络及其CPT（Conditional Probability Table）：



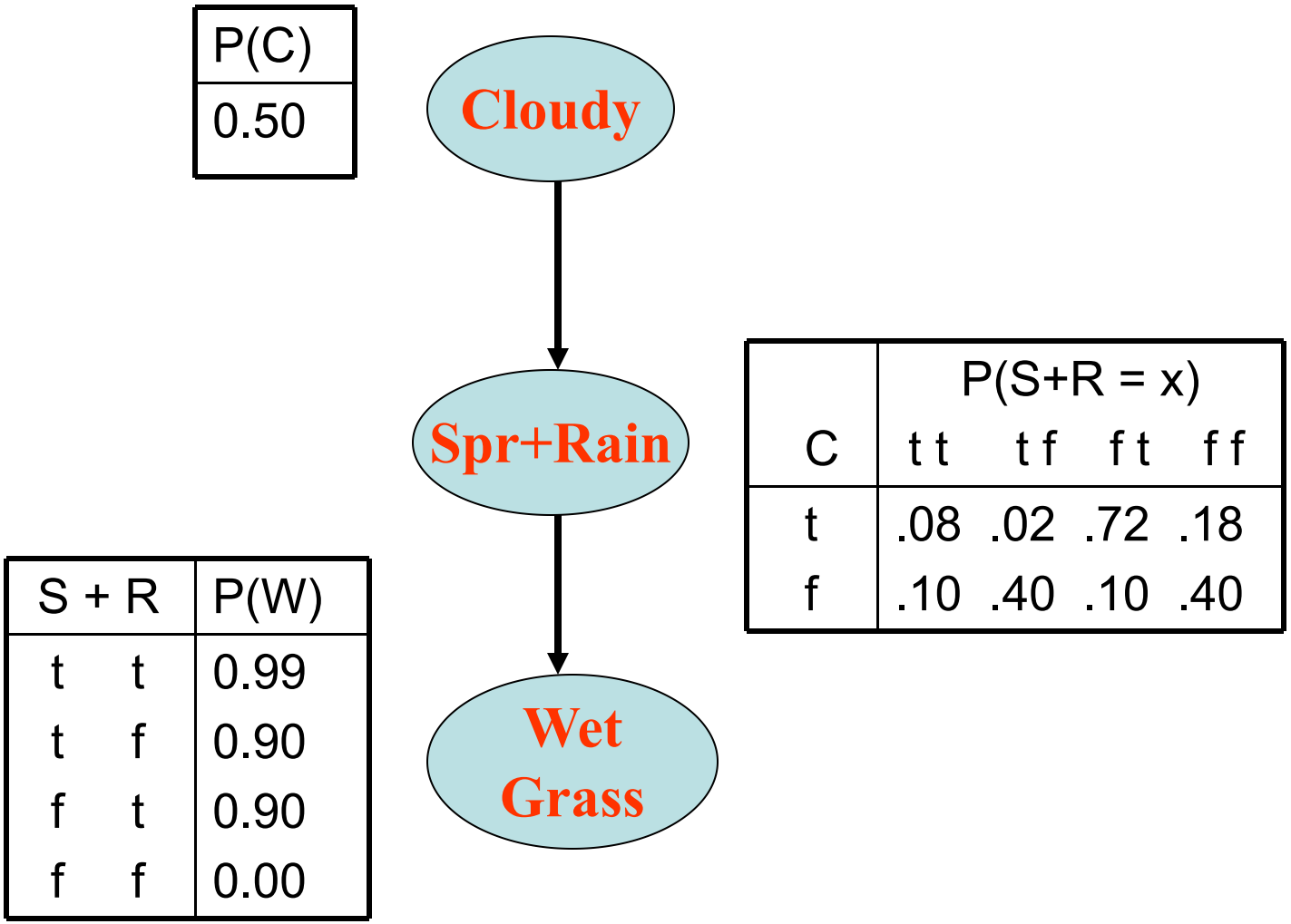
P(C)
0.50

C	P(S)
t	0.10
f	0.90

C	P(R)
t	0.80
f	0.20

S	R	P(W)
t	t	0.99
t	f	0.90
f	t	0.90
f	f	0.00

等价的联合树及其CPT：



网络模型结构G和参数θ

小规模贝叶斯网络构造原则：

- 首先，添加“根本原因”节点
- 然后，加入受它们直接影响的变量
- 依次类推，直到叶节点，即对其它变量没有直接因果影响的节点
- 两节点间的有向边的取舍原则：更高精度概率的重要性与指定额外信息的代价的折衷
- “因果模型”比“诊断模型”需要更少的数据，且这些数据也更容易得到

贝叶斯网络的结构学习的要点

贝叶斯网络的结构学习经常被视为是一种最优化问题，计算的任务就是找到最佳的结构使得统计学意义上的得分最高。在构建贝叶斯网络的过程中，并不能对所有的结构分别进行计算评估，只能采用启发式搜索算法，在有限的搜索空间中寻优。

贝叶斯网络算法的两种类型

- **变量相关性分析：**

  最大加权生成树算法(maximum weighted spanning tree，MWST）

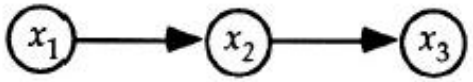
  - 三阶段相关性分析算法（three phase dependency analysis，TPDA）等；
- **基于搜索评分的学习：比如网络得分**，如B算法、K2算法、基于蚁群优化的B算法（ant colony optimization—B algorithm，ACOB；
- 例：K2算法,1992年,目标是估计后验概率
  - 该算法要求先确定网络中节点变量的次序，对先验知识的依赖性很大。
  - 在该算法中提出了节点模块化思路，即各节点的父节点集相互独立

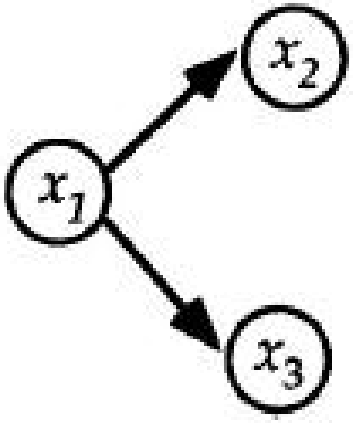
$$\max_{Bs}[P(G,D)] = C\prod_{i=1}^{n}\max_{\Pi_i}[\prod_{j=1}^{q_i}\frac{(r_i-1)!}{(N_{ij}+r_i-1)!}\prod_{k=1}^{r_i}N_{ijk}!]$$

  - 困难：不了解相关的领域知识或没有专家指导的情况下，确定变量的次序就变得相当困难。

Table 1. A database example. The term *case* in the first column denotes a single training instance (record) in the database—as for example, a patient case. For brevity, in the text we sometimes use 0 to denote *absent* and 1 to denote *present*.

	Variable values for each case		
Case	x_1	x_2	x_3
1	*present*	*absent*	*absent*
2	*present*	*present*	*present*
3	*absent*	*absent*	*present*
4	*present*	*present*	*present*
5	*absent*	*absent*	*absent*
6	*absent*	*present*	*present*
7	*present*	*present*	*present*
8	*absent*	*absent*	*absent*
9	*present*	*present*	*present*
10	*absent*	*absent*	*absent*





$$P(B_S, D) = \int_{B_P} P(D \mid B_S, B_P) f(B_P \mid B_S) P(B_S) dB_P,$$

- B_P是一个向量，其值表示在bayes网络结构B_S下的条件概率分布。

$$P(B_S, D) = \int_{B_P} \left[\prod_{h=1}^{m} P(C_h \mid B_S, B_P) \right] f(B_P \mid B_S) P(B_S) dB_P,$$

where m is the number of cases in D and C_h is the hth case in D.

Theorem 1. Let Z be a set of n discrete variables, where a variable x_i in Z has r_i possible value assignments: $(v_{i1}, \ldots, v_{ir_i})$. Let D be a database of m cases, where each case contains a value assignment for each variable in Z. Let B_S denote a belief-network structure containing just the variables in Z. Each variable x_i in B_S has a set of parents, which we represent with a list of variables π_i. Let w_{ij} denote the jth unique instantiation of π_i relative to D. Suppose there are q_i such unique instantiations of π_i. Define N_{ijk} to be the number of cases in D in which variable x_i has the value v_{ik} and π_i is instantiated as w_{ij}. Let

$$N_{ij} = \sum_{k=1}^{r_i} N_{ijk}.$$

1. The variables in Z are discrete
2. Cases occur independently, given a belief-network model
3. There are no cases that have variables with missing values
4. Before observing D, we are indifferent regarding which numerical probabilities to assign to the belief network with structure B_S.

$$P(B_S, D) = P(B_S) \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{(r_i - 1)!}{(N_{ij} + r_i - 1)!} \prod_{k=1}^{r_i} N_{ijk}!$$

大规模贝叶斯网络的近似推理

- **大规模多连通BN的精确推理是不可操作的，只能通过近似推理来解决。**
- **后验概率计算的主要采样方法**

  直接采样方法,直接采样算法,拒绝采样(Rejection sampling)算法,似然加权(Likelihood weighting)算法

  - **马尔可夫链蒙特卡罗（MCMC）方法**
  - 变分法(Variational method)
  - 环传播(Loopy propagation)方法

马尔可夫链蒙特卡罗结构学习算法（Metropolis Hasting）

算法的主要思想是：由于网络结构的后验分布 $P(G \mid D)$是无法直接计算得到的，可以首先构造一个 markov 链，使其极限分布收敛于网络结构的后验分布 $P(G \mid D)$；然后使用 Monte Carlo 方法对此 markov 链进行抽样，得到网络结构的样本序列，即 $G^0 G^1,\ldots G^i,\ldots)$；最后从此序列中挑出具有最大后验概率的网络结构，来近似网络的最优结构。算法中从第 i 个网络结构 G 转移到新网络结构 G' 的接受概率为：

$$\alpha(G,G') = \min\{1, R_\alpha\} \tag{2}$$

$$R_\alpha = \frac{\#(nbd(G))P(G' \mid D)}{\#(nbd(G'))P(G \mid D)} \tag{3}$$

式中，*# nbd G表示由G 和那些对G 实行一次边的简单操*作（删除边、增加边、改变边方向）得到的图构成的集合，称为*G 的邻近域。#(( )) nbd G 为 G 的邻近域中元素的个数。*