第6节　贝叶斯网络理论及方法

—— 推理问题及学习实践

6.1 贝叶斯网推理问题

- 推理是通过计算回答查询的过程。
- 三大类推理问题：
  - ✓ 后验概率问题
  - ✓ 最大后验假设问题
  - ✓ 最大解释可能问题

6.1.1 后验概率问题

- 后验概率问题指的是已知贝叶斯网中某些变量的采样值，计算另外一些变量的后验概率分布的问题。
- 概率推理的四种类型：
  - ✓ 诊断推理
  - ✓ 预测推理
  - ✓ 原因关联推理
  - ✓ 混合推理

6.1.2 最大后验假设问题

- 最大后验假设(MAP)：我们在所有可能的假设中，找出后验概率最大的 h^* 那个假设

即：

$$\boldsymbol{\theta}^* = \boldsymbol{arg}\max_{\theta_i} \boldsymbol{P}(\boldsymbol{\theta}|\boldsymbol{D})$$

- **例6.1** 如下图所示，假设病人的 X 光胸透结果有阴影，但却没有呼吸困难的症状，即证据为 $\{ X = y, D = n \}$，问病人患哪个或哪些疾病的可能性最大?

由图中关系可知：$\{ T, L, B \}$ 的组合有八种。我们需要的是后验概率即 $P(T, L, B \mid X = y, D = n)$最大的那个组合。

6.1.3. 最大可能解释问题

- 在贝叶斯网中，证据 D 的一个解释指的是网络中全部变量的一个状态组合。而后验概率最大的那个解释即为最大可能解释(MPE)。

在例6.1中，对证据 $\{X = y, D = n\}$ 的一个解释为：

$$\{A = y, S = y, T = y, L = y, B = y, X = y, D = n\}$$

共有 $2^5 = 32$种解释，MPE问题即是要找出概率最大的那个解释。

6.2 推理中的变量消元算法

- 概率分布的分解与推理复杂度
- 消元运算
- 算法描述

6.2.1 概率分布的分解与推理复杂度

1、以下图为例：

$$A \rightarrow B \rightarrow C \rightarrow D$$

由图可知：C与A相互独立，D与A、B相互独立，则

$$P(D)=\sum_{A,B,C}P(A,B,C,D)=\sum_{A,B,C}P(A)P(B|A)P(C|B)P(D|C) \tag{1}$$

表达式的复杂度为：28次乘法和14次加法。

- 我们注意到：与变量 A 相关的表达式为 $P(A)$ 和 $P(B|A)$，与变量 B 相关的表达式为 $P(B|A)$ 和 $P(C|B)$，与变量 C 相关的表达式为 $P(C|B)$ 和 $P(D|C)$。则式(1)可改写为如下形式：

$$P(D)=\sum_{C}P(D|C)\sum_{B}P(C|B)\sum_{A}P(A)P(B|A) \tag{2}$$

复杂度为：12次乘法和6次加法。

联合分布分解的流程图：

$$F = \{ P(A), P(B|A), P(C|B), P(D|C)$$

$$\psi_1(B) = \sum_A P(A)P(B|A)$$

$$F = \{ \psi_1(B), P(C|B), P(D|C)\}$$

$$\psi_2(C) = \sum_B P(C|B)\,\psi_1(B)$$

$$F = \{ \psi_2(B), \mathrm{P}(D|C)\}$$

$$P(D) = \psi_3(D) = \sum_C P(D|C)\psi_2(C)$$

6.2.2 消元运算

$F=\{f_1, f_2, \cdots, f_m\}$

- 设 $F(X_1, X_2, \ldots, X_n)$ 是变量 $\{X_1, X_2, \ldots, X_n\}$的一个函数，从$F(X_1, X_2, \ldots, X_n)$ 出发，可通过如下方式获得变量 $\{X_2, \ldots, X_n\}$ 的一个函数：

$$G(X_2, \ldots, X_n) = \sum_{X_1} F(X_1, X_2, \ldots, X_n)$$

这个过程称为消元，即从函数 $F(X_1, X_2, \ldots, X_n)$ 中消去 X_1，得到函数 $G(X_2, \ldots, X_n)$。

6.2.3 算法描述

- 定义函数上的运算：设 $f(X, Y)$ 是两个变量 X 和 Y 的函数，其中 $\mathrm{X} \cap \mathrm{Y} = \varnothing$，并设 x 为 X 的一取值。在设 $f(X, Y)$ 中，将 X 设置为 x 得到一个关于 Y 的函数 $f_{X=x}(Y)$：

$$f_{X=x}(Y=y) = f(X=x, Y=y)$$

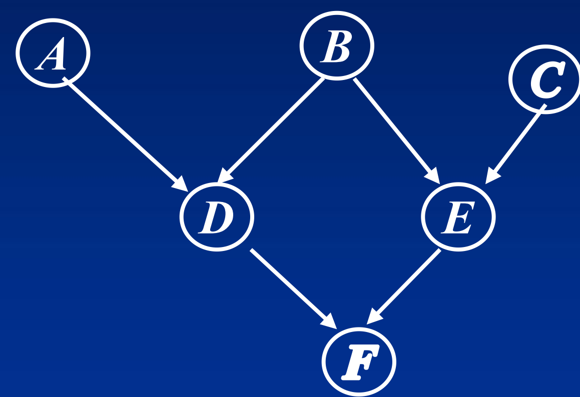
我们把 $f_{X=x}(Y)$ 记为 $f(X=x, Y)$。

- 设 Q 是 Y 的一个子集合，逐个消去所有在 Y 中但不在 Q 中的变量，得到一个证据集合，表示为 $P(Q, E=e)$。将所有 $E=e$ 的条件概率相加，就可以得到 $P(E=e)$ 。按照条件概率的定义，可得：

$$P(Q|E=e) = \frac{P(Q, E=e)}{P(E=e)}$$

其中 $P(E=e) = \sum_{Q} P(Q, E=e)$。此过程即为变量消元法，简称VE算法。

- 例6.2：如图所示贝叶斯网中，已知证据为$\{F=0\}$，用VE算法计算 $P(A \mid F=0)$。



- 变量消元顺序 $\rho = \langle C, E, B, D \rangle$ 。上图所给出的贝叶斯网的分解为：$T=\{P(A), P(B), P(C), P(D|A,B), P(E|B,C), P(F|D,E)\}$。

$$F = \left\{ \begin{matrix} \mathrm{P(A),P(B),}\underline{\mathrm{P(C)}}\mathrm{,P(D|A,B),} \\ \underline{\mathrm{P(E|B,C)}}\mathrm{,P(F=0|D,E)} \end{matrix} \right\}$$

$$\psi_1(B,E) = \sum_C P(C)P(E|B,C)$$

$$F = \left\{ \begin{matrix} \mathrm{P(A),P(B),P(D|A,B),} \\ \underline{\mathrm{P(F=0|D,E)}}, \underline{\psi_1(B,E)} \end{matrix} \right\}$$

$$\psi_2(B,D) = \sum_E P(F=0|D,E)\,\psi_1(B,E)$$

$$F = \left\{ \mathrm{P(A)}, \underline{\mathrm{P(B)}}, \underline{\mathrm{P(D|A,B)}}, \underline{\psi_2(B,D)} \right\}$$

$$\psi_3(A,D) = \sum_B P(B)P(D|A,B)\,\psi_2(B,D)$$

$$F = \left\{ \mathrm{P(A)}, \underline{\psi_3(A,D)} \right\}$$

$$\psi_4(A) = \sum_D \psi_3(A,D)$$

$$F = \left\{ \mathrm{P(A)}, \underline{\psi_4(A)} \right\}$$

$$P(A|F=0) = \frac{P(A)\psi_4(A)}{\sum_A P(A)\psi_4(A)}$$

例6.2：用一个社会调查研究的例子来说明贝叶斯网络的计算方法。通过对某地区的中学生进行调查，找出以下几个变量因素对学生的就学情况产生影响：

- 性别 X_1：male, female
- 智商 X_2：low, lower middle, upper middle, high
- 家庭经济 X_3：low, lower middle, upper middle, high
- 家庭鼓励 X_4：low, high
- 是否打算上大学 X_5：yes, no

- 表6.1是对10318名学生的统计结果。表中的第1格数据表示X_1=male,X_2=low,X_3= low,X_4= low,X_5=yes的学生个数为4，第2格数据表示X_1=male,X_2= low,X_3= low,X_4= low,X_5= no的学生个数为349，……，依此类推，在表的下半部分(即5~8行)，X的取值都为female。

表6.1 学生就学情况调查表

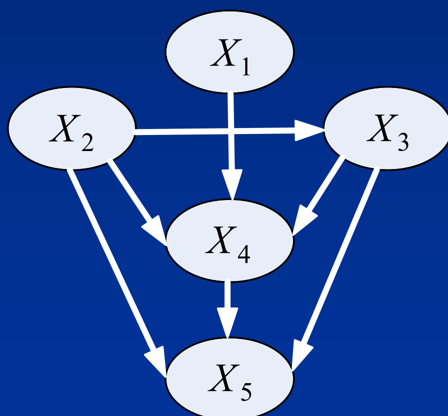
4	349	13	64	9	207	33	72	12	126	38	54	10	67	49	43
2	232	27	84	7	201	64	95	12	115	93	92	17	79	119	59
8	166	47	91	6	120	74	110	17	92	148	100	6	42	198	73
4	48	49	57	5	47	123	90	9	41	224	65	8	17	414	54
5	454	39	44	5	312	14	47	8	216	20	35	13	96	28	24
11	285	29	61	19	236	47	88	12	164	62	85	15	113	72	50
7	163	36	72	13	193	75	90	12	174	91	100	20	81	142	77
6	50	36	58	5	70	110	76	12	48	123	81	13	49	360	98

- 用贝叶斯网络作数据挖掘就是要找出这此变量之间的因果关系，具体计算过程如下：

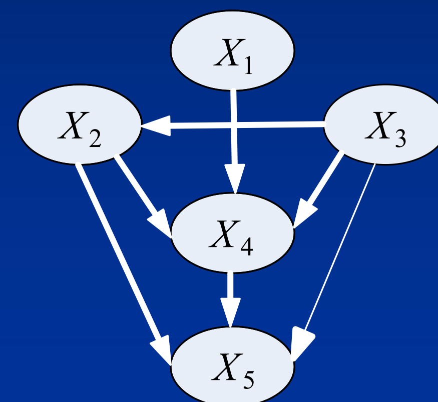
1) 根据先验知识，选择合适的网络结构

对于有 n 个变量的数据样本，可能组成的网络结构有 $n!$ 种，要对每一个网络结构进行计算是不可能的。利用现有的专家知识，就可以排除大量的不合理组合。例如，在本例中，学生的性别和家庭的经济状况是没有关系的，所以在贝叶斯网络中，不会有 X_1和 X_3之间的联系。

- 在下面的计算中，我们只选择了如图6.1(*a*)、(*b*)所示的 S_1和 S_2两种网络结构。它们唯一的区别是学生的智商与家庭经济的因果关系不同。



(*a*) 网络结构



(*b*) 网络结构

图6.1 两种可能的贝叶斯网络结构

2) Dirichlet分布超参数α_{ijk}及$\alpha_{ij\cdot}$的取值：

- Cooper和Herskovits提出的K2算法中，假设所有参数先验分布都是均匀分布。在此例中，先验分布的超参数α_{ijk}及$\alpha_{ij\cdot}$都用1来估计。

3) 根据贝叶斯网结构评分函数：

$$\log p(S,D)=\log p(D\mid S)+\log p(S)$$

分别计算两种不同结构S_1和S_2下的后验概率$p(S\mid D)$。

决定$\log p(S \mid D)$值大小的边缘似然函数$p(D \mid S)$，记为$L(S \mid D)$，其计算公式为：

$$L(S \mid D) = \prod_{i=1}^{n}\prod_{j=1}^{q_i}\frac{\Gamma(\alpha_{ij\cdot})}{\Gamma(\alpha_{ij\cdot}+m_{ij\cdot})}\cdot\prod_{k=1}^{r_i}\frac{\Gamma(\alpha_{ijk}+m_{ijk})}{\Gamma(\alpha_{ijk})}$$

将此例的样本数据代入上式，分别对上述两个网络结构进行计算，得出：

$$L(S_1 \mid D) \cong 1.0$$

$$L(S_2 \mid D) \cong 1.2\times10^{-12}$$

由此得出，网络结构 S_1 更能反映变量之间的因果关系。同时注意到，虽然 S_1 和 S_2 的结构差别不大，但计算的结果截然相反，这说明贝叶斯网络有较好的敏感性。

贝叶斯网工具箱BNT：

- 基于MATLAB 的贝叶斯网络工具箱BNT是Kevin P. Murphy 基于MATLAB语言开发，提供了许多贝叶斯网络学习的底层基础函数库, 支持多种类型的结点(概率分布)、精确推理和近似推理、参数学习和结构学习、静态模型和动态模型。

BNT下载安装步骤：

1、解压FullBNT-1.0.4.zip，将整个目录FullBNT-1.0.4复制到MATLAB的安装目录的TOOLBOX目录下，如D:\MATLAB7\toolbox\

2、打开Matlab，在MATLAB命令窗口中输入以下命令：

```
>> cd('D:\MATLAB7\toolbox\FullBNT-1.0.4')
>> addpath(genpath
('D:\MATLAB7\toolbox\FullBNT-1.0.4'))
>>
```

将TOOLBOX下新加的BNT工具箱加到MATLAB的搜索路径中去。

3、为了永久保存上面的路径，以免下次重启MATLAB时重新添加，在MATLAB命令窗口下使用下面的命令：

```
>> savepath
>>
```

4、检验是否成功设置的方法：

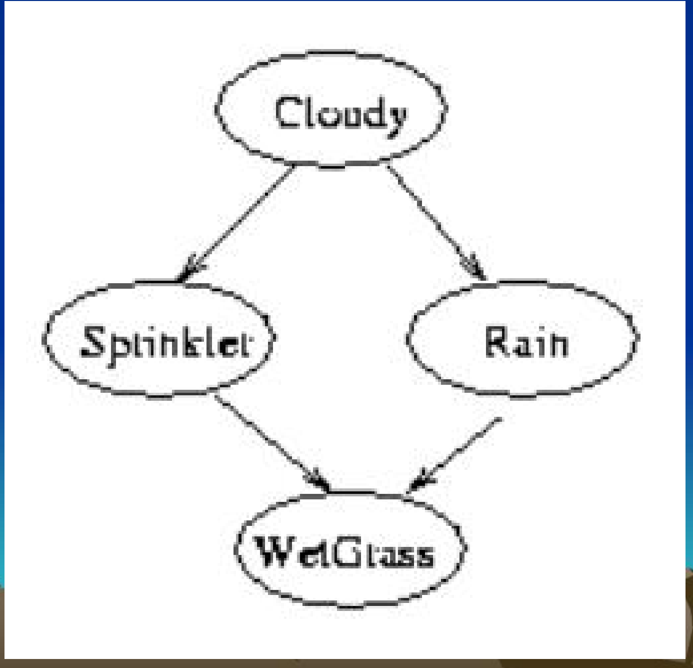
在命令窗口中输入以下命令：which test_BNT.m（可以为所加工具箱的任一个M文件名称），如果显示正确，就说明上面的设置成功。

5、检查路径

>>path 可以看到BNT工具的路径已加入。

BNT 中贝叶斯网络的表示方式：

- 在贝叶斯软件包BNT 中, 采用矩阵的形式表示贝叶斯网络的结构。即若结点i到结点j有一条弧, 则对应矩阵中(i, j)值为 1, 否则为 0：



$$\begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

BNT 中贝叶斯网络参数学习函数:

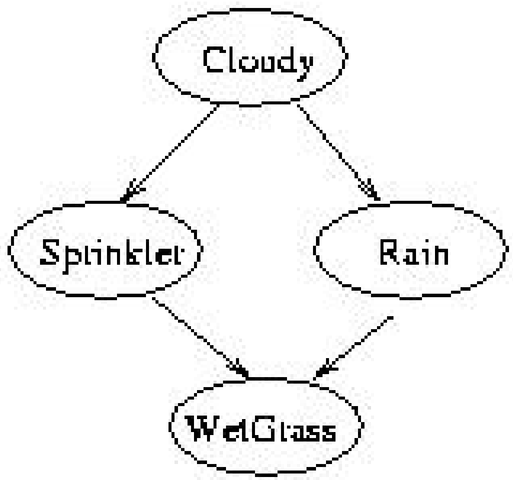
- 最大似然性估计learn_params( ) ;
- 贝叶斯方法bayes_update_params( ) 。

BNT 中贝叶斯网络结构学习函数

- K2 算法learn_struct_K2( ) ;
- 贪婪搜索GS(Greedy Search) 算法learn_struct_gs ( );
- 爬山HC (Hill Climbing) 算法learn_struct_hc( ) 。

BNT用法举例: 草地潮湿原因

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

建立贝叶斯网络结构:

- N = 4; %四个节点分别是cloudy,sprinkler,rain,wetgrass
- dag = zeros(N,N);   %网络连接矩阵初始化
- C = 1; S = 2; R = 3; W = 4; % 初始化节点顺序
- dag(C,[R S]) = 1;   % 定义节点之间的连接关系
  dag(R,W) = 1;
  dag(S,W) = 1;
- discrete_nodes = 1:N;   %离散节点
- node_sizes = 2*ones(1,N);   %节点状态数

建立网络架构的语句:

- **bnet =mk_bnet(dag,node_sizes,'names',{'cloudy','sprinkler’,'rain','wetgrass'},'discrete',discrete_nodes);**

BNT网络参数学习及结果

- 最大似然方法

```
bnet1 = learn_params(bnet, data);
```

- 贝叶斯估计方法

```
bnet2=bayes_update_params(bnet,data);
```

```
data =

     2     2     2     1     1     1     1     2     1     2     2     2     1     2     2     1     2     1     2     2
     1     1     1     1     1     2     2     1     2     1     1     1     1     1     1     1     1     2     1     1
     2     1     2     1     1     2     2     2     1     2     2     2     2     2     1     1     2     1     2     2
     2     1     2     1     1     2     2     2     2     2     2     2     2     1     1     1     2     2     2     2

>>
```

BNT的结构学习：

- 参数初始化：

需要事先确定的最大父节点数目、节点次序

```
order=[1 2 3 4];        % 节点次序
ns=[2 2 2 2];           % 节点属性值的个数
max_fan_in=2;           % 最大父节点数目
```

参数初始化：

- 需要事先确定的最大父节点数目、节点次序

```
order=[1 2 3 4];        % 节点次序
ns=[2 2 2 2];           % 节点属性值的个数
max_fan_in=2;           % 最大父节点数目
```

K2算法：

- dag = learn_struct_K2(data,ns,order,‘max_fan_in’,max_fan_in);　% 结构学习语句

习题：服务器数据传送

- 现有两台服务器(S_1, S_2)，都会单向向用户 U 传送数据。服务器 S_1 和 S_2 之间也会有数据通讯，但无法确定它们之间的数据流向。数据包的传送只取两种可能值：T=1 (成功) 或 F=2 (失败)。

- 假设贝叶斯网络由S_1、S_2和U这三个节点构成，现采集了100条该网络的数据传送样本，如文件 **server_data.txt** 所给出。该文件中，每行代表一个三节点网络的样本， 试利用贝叶斯算法学习得到该网络的结构和参数。

下载：Server_data. txt