Artificial Intelligence

8.不确定知识与推理c

对应课本13-15章

贝叶斯网络推理

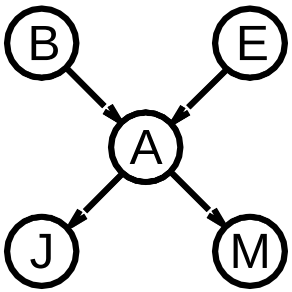
- 使用枚举进行精准推理
- 通过消除变量进行精准推理
- 随机模拟的逼近推理
- 马尔科夫链蒙特卡罗逼近推理

推理任务

- 简单查询：计算后验边缘概率 $\mathrm{P}(X_i|\mathrm{E} = \mathrm{e})$
  - 例如 $P(NoGas|Gauge = empty, Lights = on, Starts = false)$
- 连接查询：$\mathrm{P}(X_i, X_j|\mathrm{E} = \mathrm{e}) = \mathrm{P}(X_i|\mathrm{E} = \mathrm{e})\mathrm{P}(X_j|X_i, \mathrm{E} = \mathrm{e})$
- 最优决策：决策网络包括效用信息
  - 概率推理需要$P(outcome|action, evidence)$
- 信息价值：下一步找什么证据?
- 敏感性分析：哪些概率值是最关键的?
- 对答案的解释：为什么我需要一个新的启动器?

通过枚举的推理

- 这种方法，可以从节点的交叉中计算出变量，而不需要构造交叉状态的显式表示
- 简单查询入室行窃网络：



$\mathrm{P}(B|j,m)$
$= \mathrm{P}(B,j,m)/P(j,m)$
$= \alpha\mathrm{P}(B,j,m)$
$= \alpha\ \Sigma_e \Sigma_a \mathrm{P}(B,e,a,j,m)$

- 使用条件概率表(CPT)的乘积重写完整的联合项：

$\mathrm{P}(B|j,m)$
$= \alpha\ \Sigma_e \Sigma_a \mathrm{P}(B)P(e)\mathrm{P}(a|B,e)P(j|a)P(m|a)$
$= \alpha\mathrm{P}(B)\ \Sigma_e P(e)\ \Sigma_a \mathrm{P}(a|B,e)P(j|a)P(m|a)$

- 递归深度优先枚举
  - 时间复杂度： $O(n)$
  - 空间复杂度： $O(d^n)$

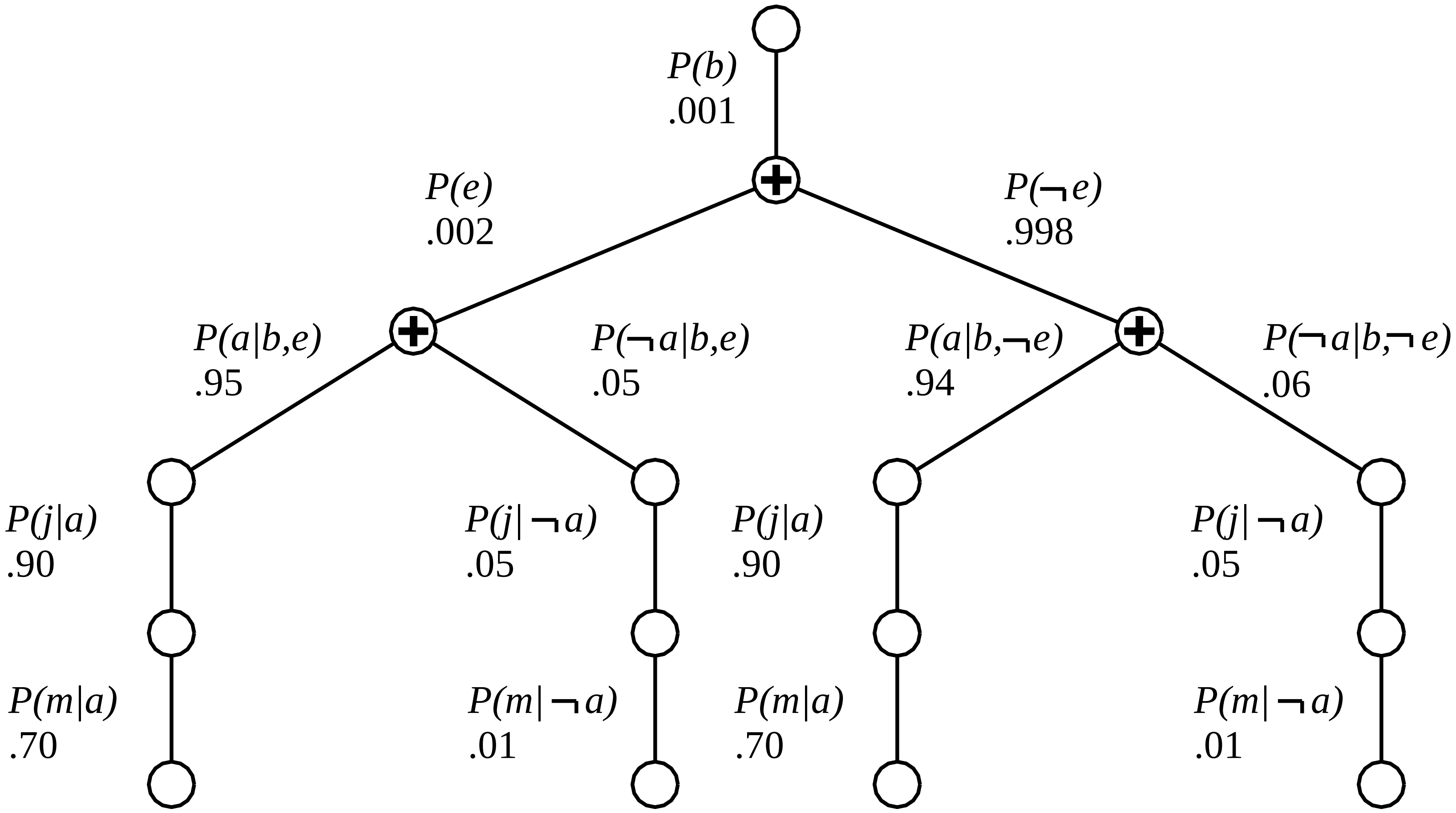
枚举算法

```
function Enumeration-Ask(X, e, bn) returns a distribution over X
    inputs: X,  the query variable
              e, observed values for variables E
              bn, a Bayesian network with variables { X } ∪E ∪Y

    Q(X) ← a distribution over X, initially empty
    for each value x_i of X do
        extend e with value x_i for X
    Q(x_i) ← Enumerate-All(Vars[bn],e) return
    Normalize(Q(X))


function Enumerate-All(vars, e) returns a real number
    if Empty?(vars) then return 1.0
    Y ← First(vars)
    if Y has value y in e
        then return P(y|Pa(Y)) × Enumerate-All(Rest(vars), e)
        else return Σ_y P(y|Pa(Y)) × Enumerate-All(Rest(vars), e_y)
            where e_y is e extended with Y = y
```

评价树



- 枚举效率低：需要重复计算每个值
  - 比如对于e的每个取值需要计算$P\ (j|a)P\ (m|a)$

变量消除推理

- 变量消除：从右到左进行求和，存储中间结果(因子)，避免重新计算

$$
\begin{aligned}
&\mathbf{P}(B|j,m) \\
&\quad = \alpha \underbrace{\mathbf{P}(B)}_{B} \textstyle\sum_e \underbrace{P(e)}_{E} \textstyle\sum_a \underbrace{\mathbf{P}(a|B,e)}_{A} \underbrace{P(j|a)}_{J} \underbrace{P(m|a)}_{M} \\
&\quad = \alpha \mathbf{P}(B) \textstyle\sum_e P(e) \textstyle\sum_a \mathbf{P}(a|B,e) P(j|a) f_M(a) \\
&\quad = \alpha \mathbf{P}(B) \textstyle\sum_e P(e) \textstyle\sum_a \mathbf{P}(a|B,e) f_J(a) f_M(a) \\
&\quad = \alpha \mathbf{P}(B) \textstyle\sum_e P(e) \textstyle\sum_a f_A(a,b,e) f_J(a) f_M(a) \\
&\quad = \alpha \mathbf{P}(B) \textstyle\sum_e P(e) f_{\bar{A}JM}(b,e) \ \text{(sum out } A\text{)} \\
&\quad = \alpha \mathbf{P}(B) f_{\bar{E}\bar{A}JM}(b) \ \text{(sum out } E\text{)} \\
&\quad = \alpha f_B(b) \times f_{\bar{E}\bar{A}JM}(b)
\end{aligned}
$$

变量消除：基本操作

- 从因子乘积中得出一个变量：
- 把常数移到求和之外
- 将剩余因子的点积中的子矩阵相加

$$\sum_x f_1 \times \cdots \times f_k = f_1 \times \cdots \times f_i \sum_x f_{i+1} \times \cdots \times f_k = f_1 \times \cdots \times f_i \times f_{\overline{x}}$$

- 假设 $f_1, \ldots, f_i$ 并不依赖于X
- 因此可以计算f_1的f_2的点积

$$f_1(x_1, \ldots, x_j, y_1, \ldots, y_k) \times f_2(y_1, \ldots, y_k, z_1, \ldots, z_l) \\ = f(x_1, \ldots, x_j, y_1, \ldots, y_k, z_1, \ldots, z_l)$$

- 如 $f_1(a, b) \times f_2(b, c) = f(a, b, c)$

变量消除算法

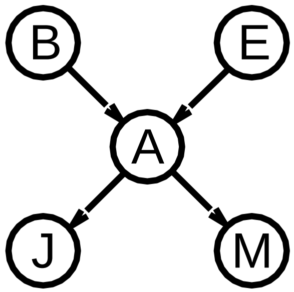
```
function Elimination-Ask(X, e, bn) returns a distribution over X
  inputs: X, the query variable
          e, evidence specified as an event
          bn, a belief network specifying joint distribution P(X1, ..., Xn)
  factors ← []; vars ← Reverse(Vars[bn])
  for each var in vars do
      factors ← [Make-Factor(var, e)|factors]
      if var is a hidden variable then factors ← Sum-Out(var, factors)
  return Normalize(Pointwise-Product(factors))
```

不相关的变量

- 考虑查询语句$P\,(JohnCalls|Burglary = true)$

$$P\,(J|b) = \alpha P\,(b)\Sigma_e P\,(e)\Sigma_a P\,(a|b,\, e)P\,(J|a)\Sigma_m P\,(m|a)$$

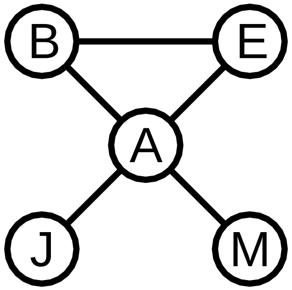
- 其中，m的和为1，而M的取值和查询语句无关
- Y是无关的，除非$Y \in Ancestors(\{X\} \cup E)$
  - 这里 $X = JohnCalls$, $E = \{Burglary\}$
  - 并且$Ancestors(\{X\} \cup E) = \{Alarm, Earthquake\}$
  - 所以$MaryCalls$是无关的
  - (将其与来自Horn子句KBs中的查询的反向链接进行比较)



不相关的变量

- 假设我们将A的父代连在一起，且取消所有的箭头
- 那么我们说A对于B被C隔开，当且仅当A也被C所隔开

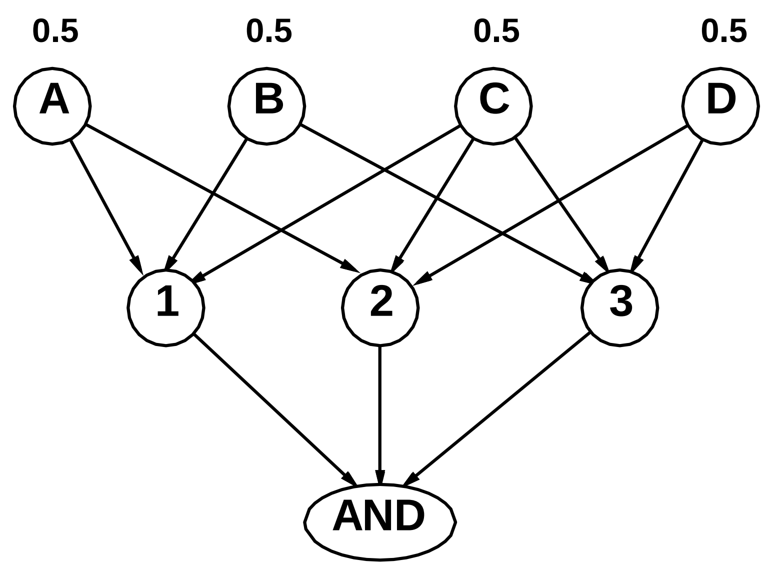
- 例如对 $P(JohnCalls|Alarm = true)$来说，$Burglary$ 和 $Earthquake$ 都是不相关的



精确推理的复杂性

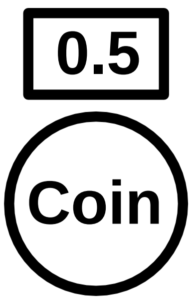
- 单连通网络(或多树)：
  - 任意两个节点最多通过一条(无向)路径连接
  - 变量消去的时间和空间代价为$O(d^k n)$
- 多连通网络:
  - 能不能把3SAT降低到np-hard精确推理
  - 相当于把3SAT模型算上#P-complete

- **1. A v B v C**
- **2. C v D v A**
- **3. B v C v D**



随机模拟推理

- 基本思想：
  - 从一个抽样分布S中抽取N个样本
  - 计算一个近似后验概率 $\hat{P}$
  - 证明这个收敛于真实概率 P
- 要点：
  - 从空网络中抽样
  - 拒绝式抽样：拒绝与证据不符的样品
  - 可能性加权：使用证据对样本进行加权
  - 马尔科夫链蒙特卡洛方法Markov chain Monte Carlo (MCMC)：随机过程的样本，其平稳分布是真实的后验

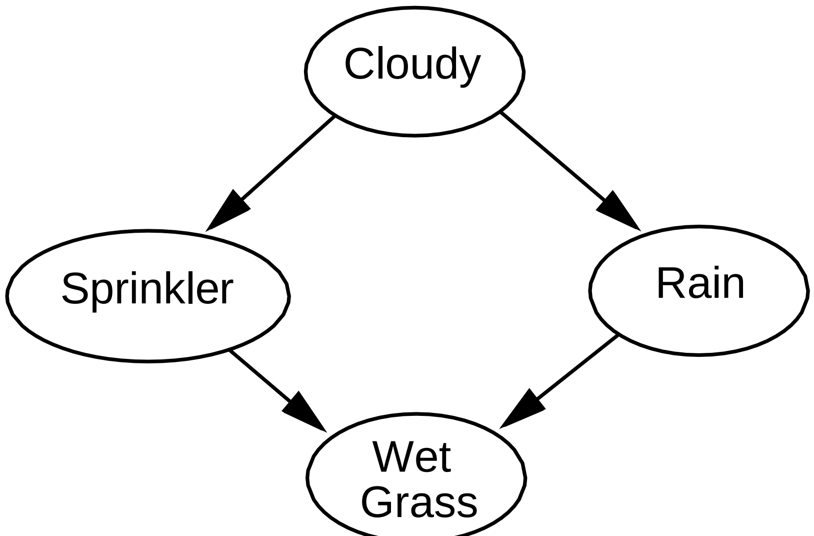


从空网络中采样

```
function Prior-Sample(bn) returns an event sampled from bn
    inputs: bn, a belief network specifying joint distribution P(X_1, ..., X_n)
    x ← an event with n elements
    for i = 1 to n do
        x_i ← a random sample from P(X_i | parents(X_i))
        given the values of Parents(X_i) in x
    return x
```

从空网络中采样实例

P(C)
.50

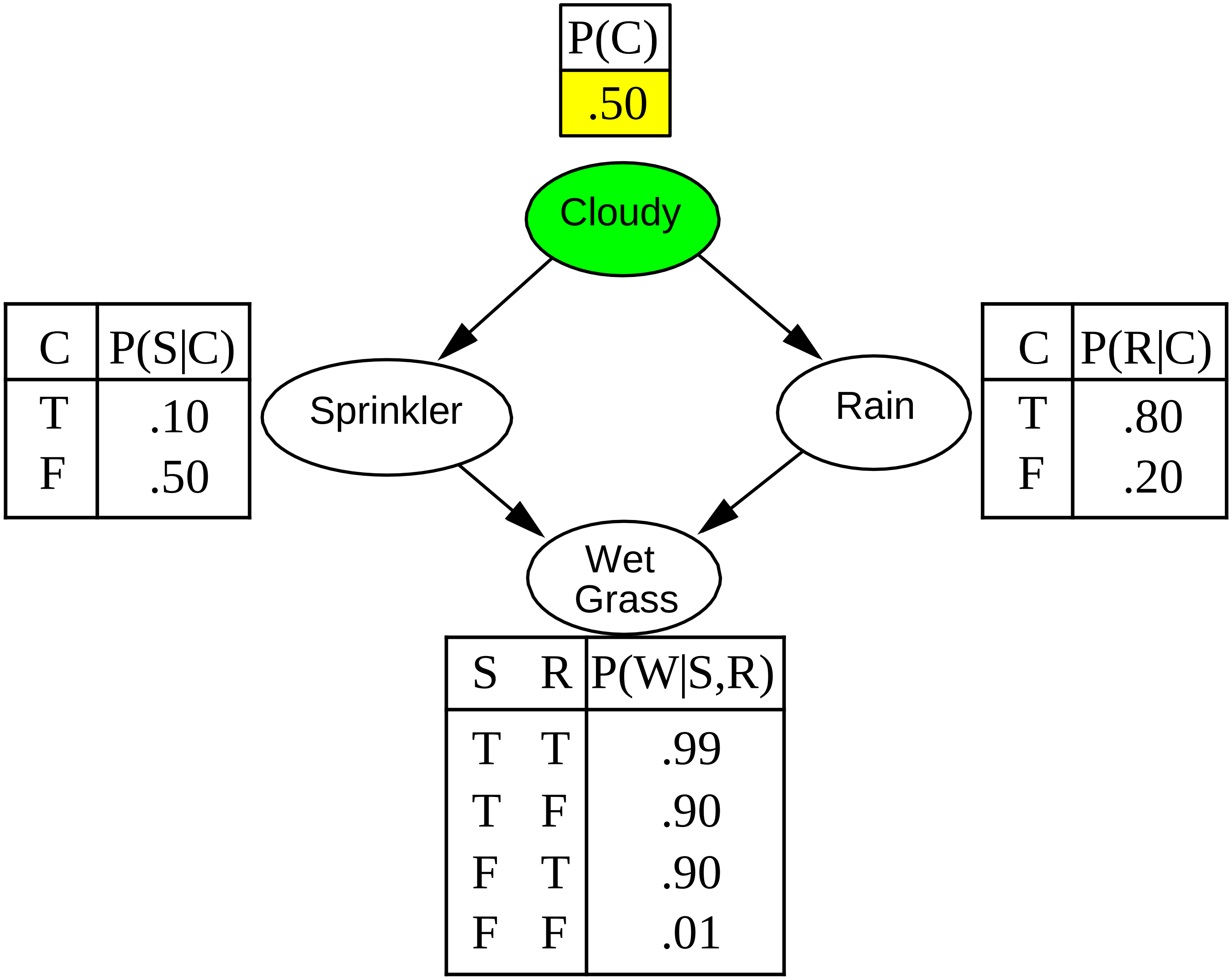


C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

从空网络中采样实例



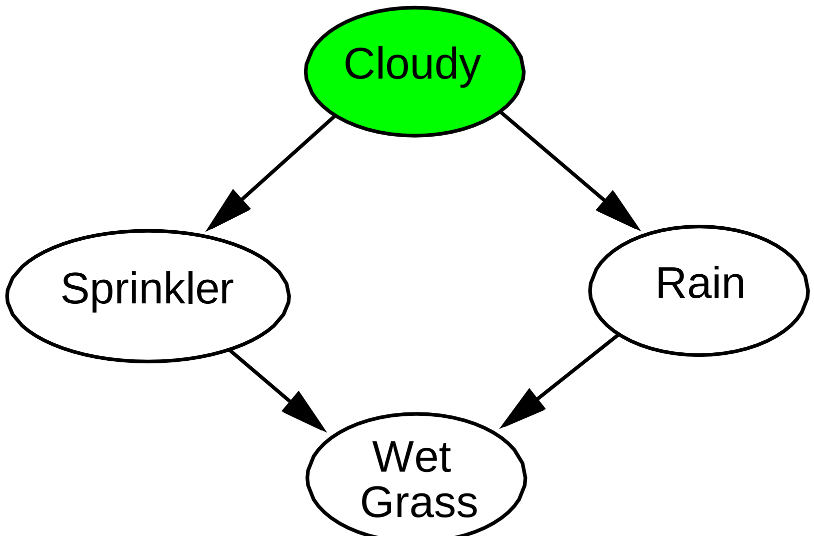
P(C)
.50

C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

从空网络中采样实例



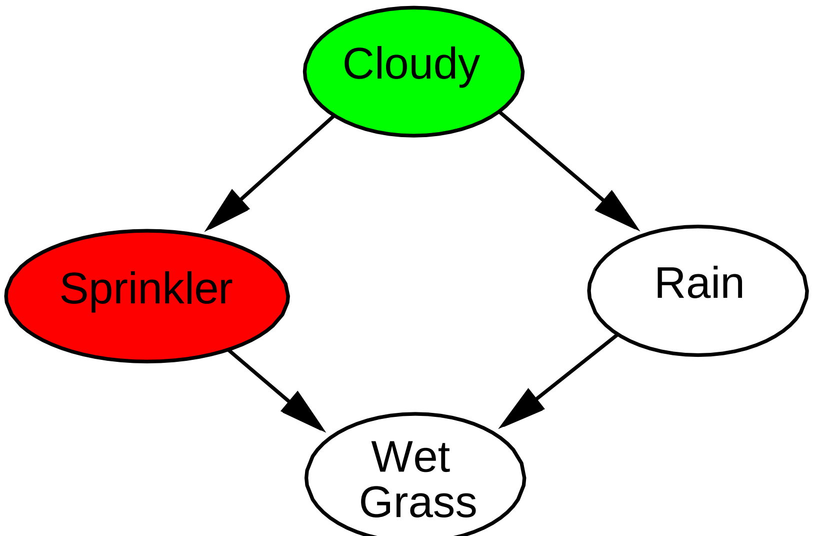
P(C)
.50

C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

从空网络中采样实例



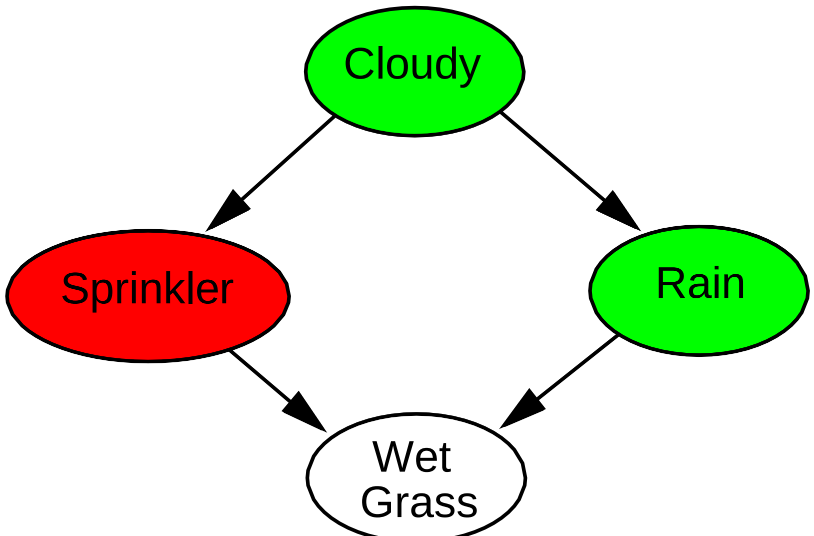
P(C)
.50

C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

从空网络中采样实例



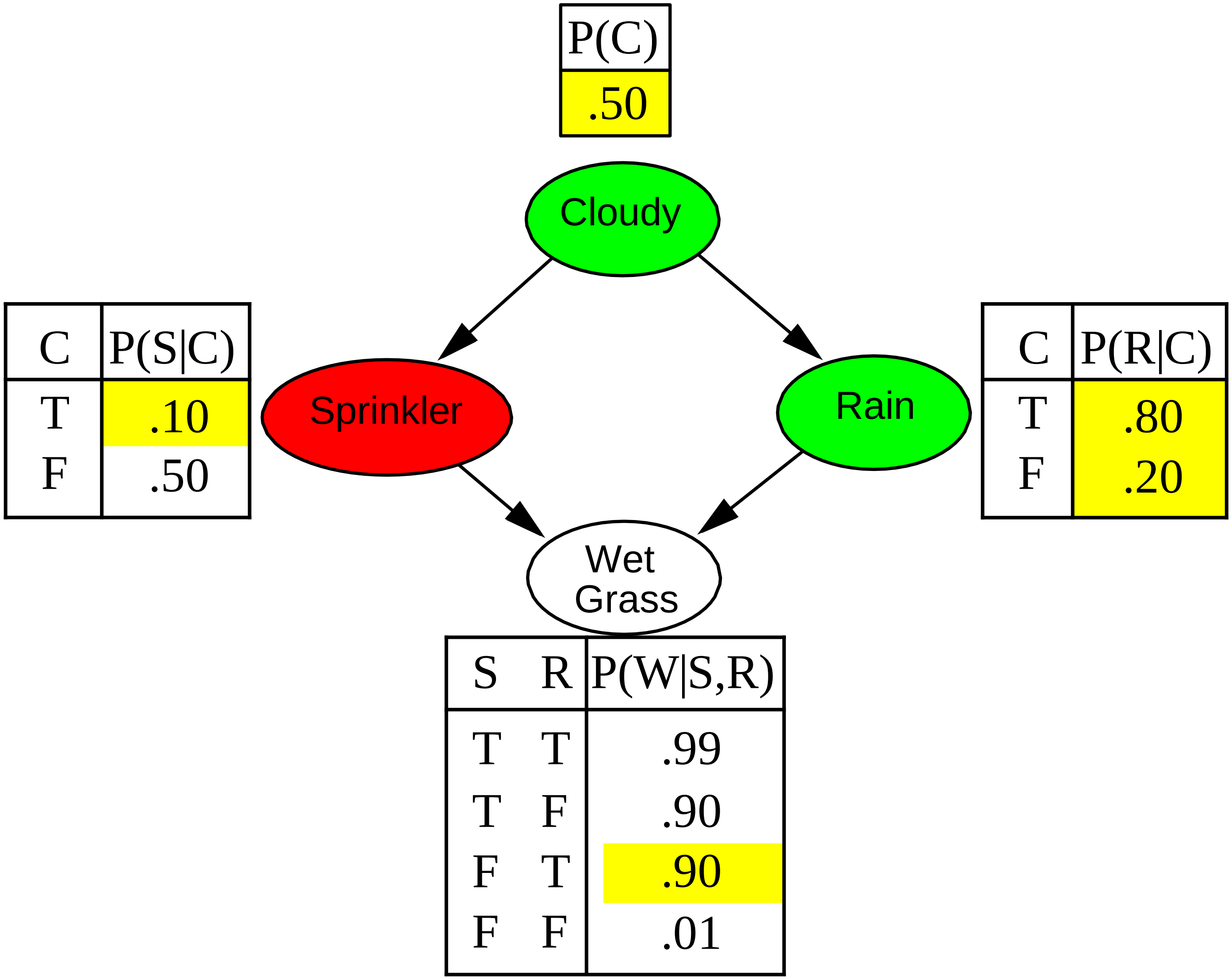
P(C)
.50

C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

从空网络中采样实例



P(C)
.50

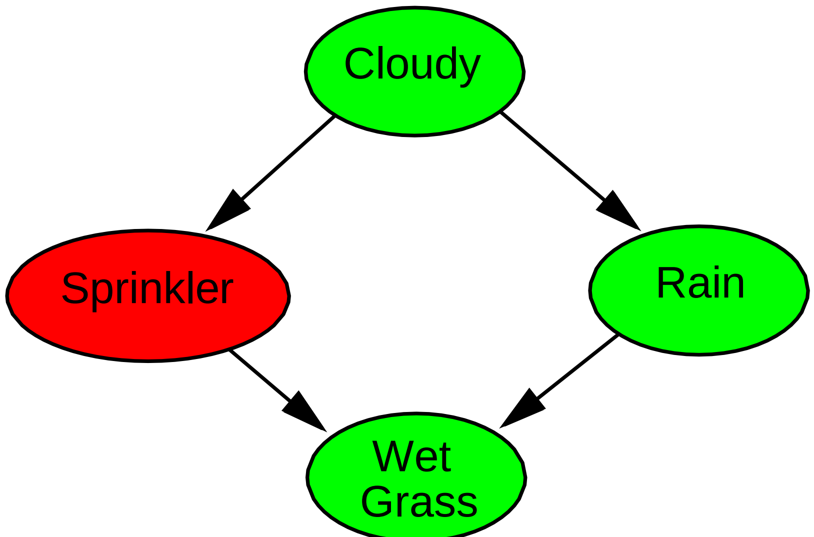
C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

从空网络中采样实例

P(C)
.50



C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

从空网络中采样

- 先验样本产生特定事件的概率

$$S_{ps}(x_1 \ldots x_1) = \prod_{i=1}^{n} P(x|parents(X)) = P(x_1 \ldots x_n)$$

- 即真实的先验概率
- 如$S_{PS}(t, f, t, t) = 0.5 \times 0.9 \times 0.8 \times 0.9 = 0.324 = P(t, f, t, t)$
- 令$N_{PS}(x_1 \ldots x_n)$作为采样所生成的事件$x_1, \ldots, x_n$
- 于是我们有

$$\begin{aligned}\lim_{n\to\infty} \hat{P}(x_1, \ldots, x_n) &= \lim_{N\to\infty} N_{PS}(x_1, \ldots, x_n)/N \\ &= S_{PS}(x_1, \ldots, x_n) \\ &= P(x_1 \ldots x_n)\end{aligned}$$

- 也就是说，通过先验样本的估计值和真实值是一致的

$$\hat{P}(x_1, \ldots, x_n) \approx P(x_1, \ldots, x_n)$$

拒绝采样

- $\hat{P}(X|e)$ 可以由样本估算，符合 e 即可

```
function Rejection-Sampling(X, e, bn, N) returns an estimate of P(X|e)
   local variables: N, a vector of counts over X, initially zero

   for j = 1 to N do
       x ← Prior-Sample(bn)
       if x is consistent with e then
           N[x] ← N[x]+1 where x is the value of X in x
   return Normalize(N[X])
```

- 如$P(Rain|Sprinkler = true)$使用了100个采样
  - 其中27个采样有$Sprinkler = true$
  - 其中8个$Rain = true$，19个$Rain = false$

  $\hat{P}(Rain|Sprinkler = true) = \text{Normalize}(\langle 8, 19\rangle) = \langle 0.296, 0.704\rangle$

- 类似于一个基本的真实世界的经验估计程序

拒绝抽样分析

$\widehat{\mathrm{P}}(X|\mathrm{e}) = \alpha \mathrm{N}_{ps}(X, e)$ 函数定义

$= \mathrm{N}_{PS}(X, \mathrm{e}) / N_{PS}(\mathrm{e})$ 通过 $N_{PS}(\mathrm{e})$ 实现归一化

$\approx \mathrm{P}(X, \mathrm{e}) / P(\mathrm{e})$ 先验采样的属性

$= \mathrm{P}(X|\mathrm{e})$ 条件概率的定义

- 因此，排斥抽样返回一致的后验估计
- 问题：如果 $P(\mathrm{e})$ 很小，就会非常昂贵
- $P(\mathrm{e})$ 随着证据变量的数量呈指数下降

可能性的权重

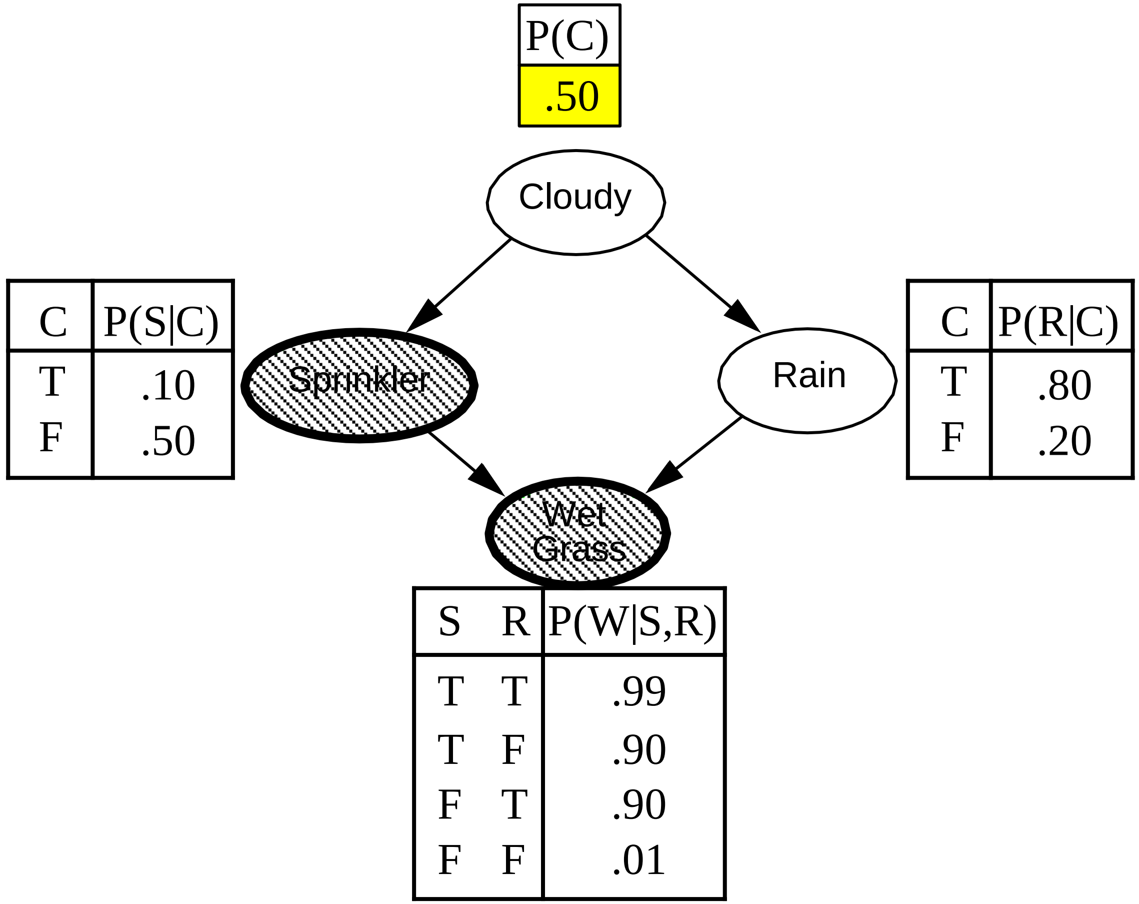
- 思路：固定证据变量，只采样非证据变量，并加权每个样本的可能性，使它符合证据

```
function Likelihood-Weighting(X, e, bn, N) returns an estimate of P(X|e)
    local variables: W, a vector of weighted counts over X, initially zero
    for j = 1 to N do
        x, w ← Weighted-Sample(bn)
        W[x] ← W[x] + w where x is the value of X in x
    return Normalize(W[X])

function Weighted-Sample(bn, e) returns an event and a weight
    x ← an event with n elements; w ← 1
    for i = 1 to n do
        if Xi has a value xi in e
            then w ← w × P(Xi = xi | parents(Xi))
            else xi ← a random sample from P(Xi | parents(Xi))
    return x, w
```

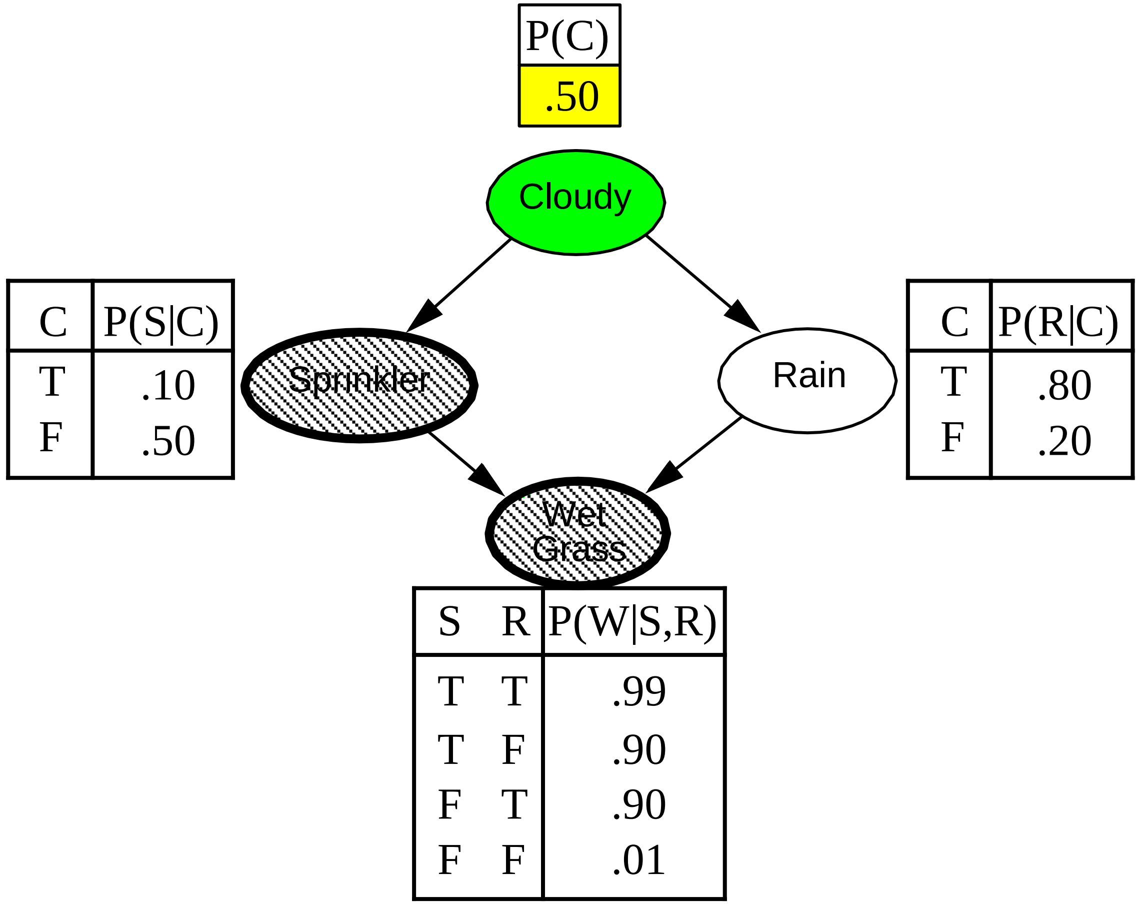
可能性权重的例子

- $w = 1.0$



可能性权重的例子

- w = 1.0



P(C)
.50

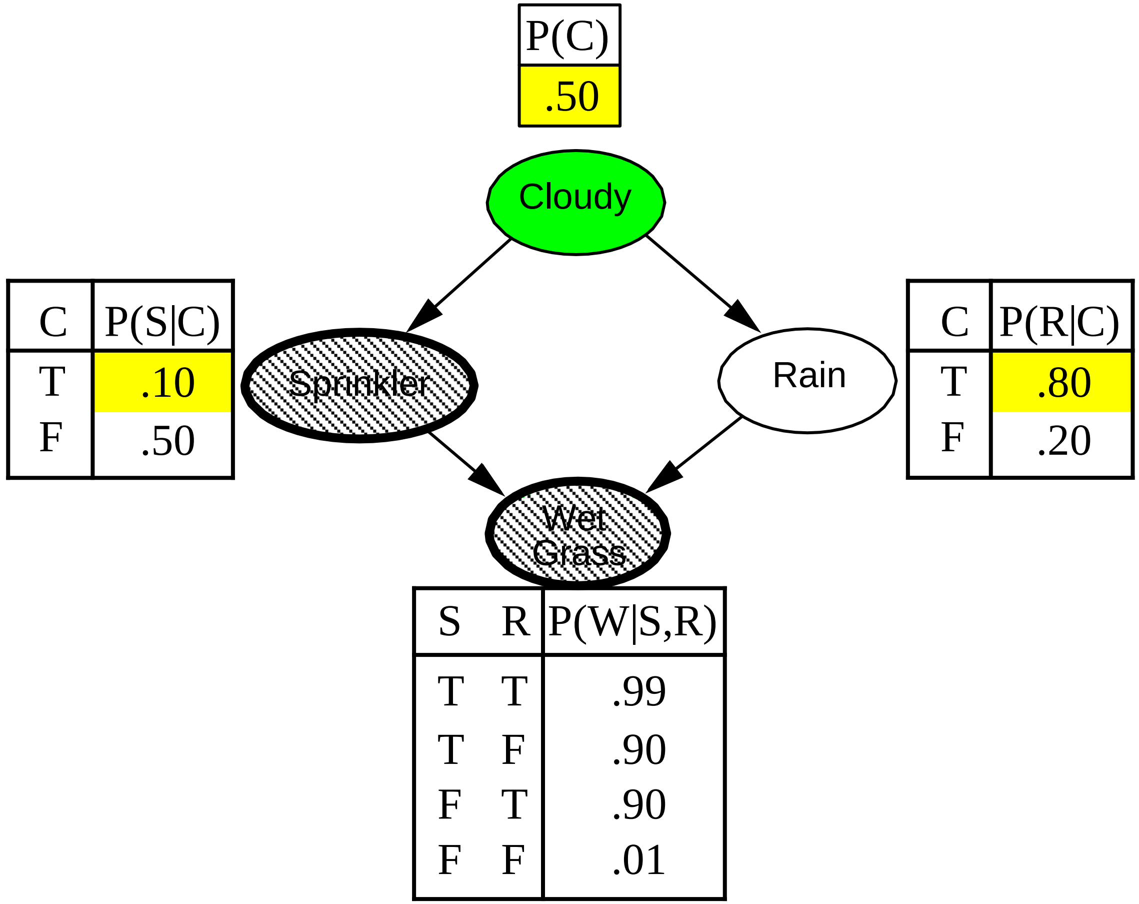
C	P(S|C)
T	.10
F	.50

C	P(R|C)
T	.80
F	.20

S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

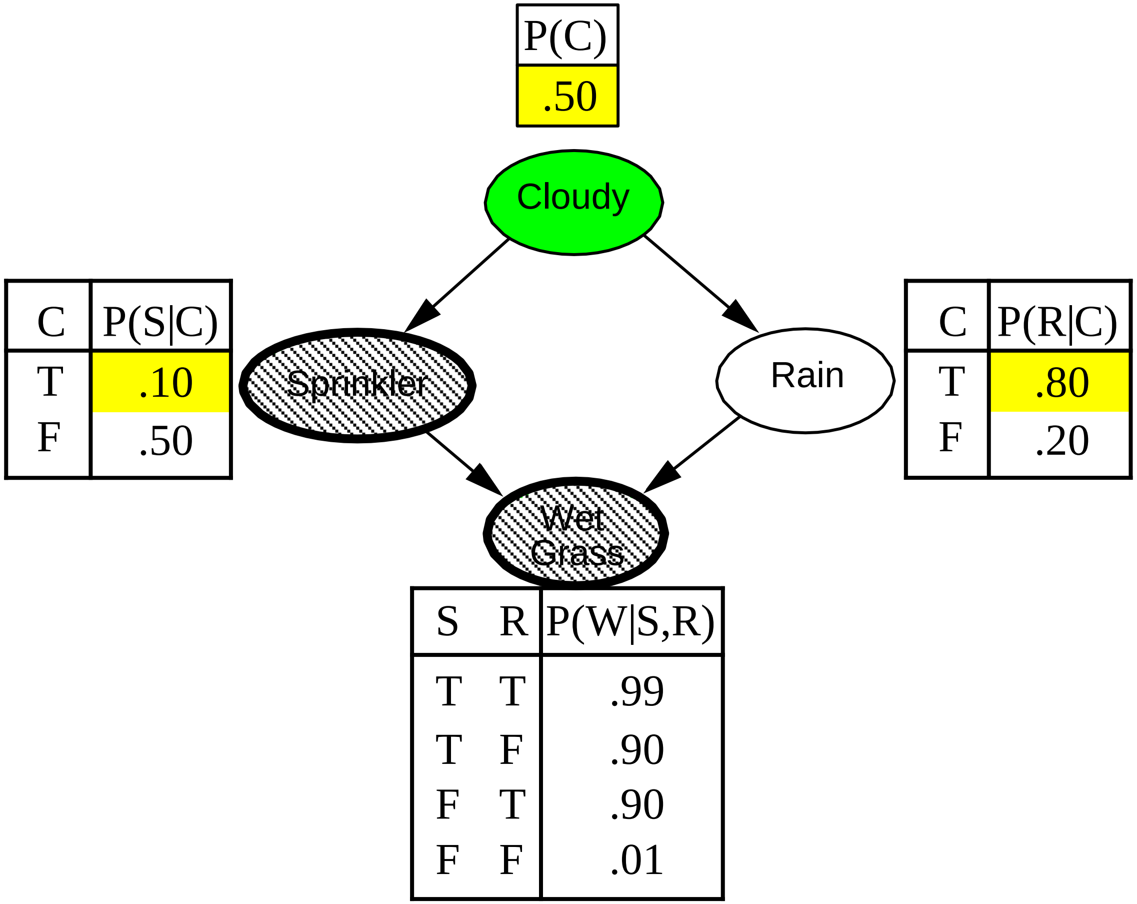
可能性权重的例子

- $w = 1.0$



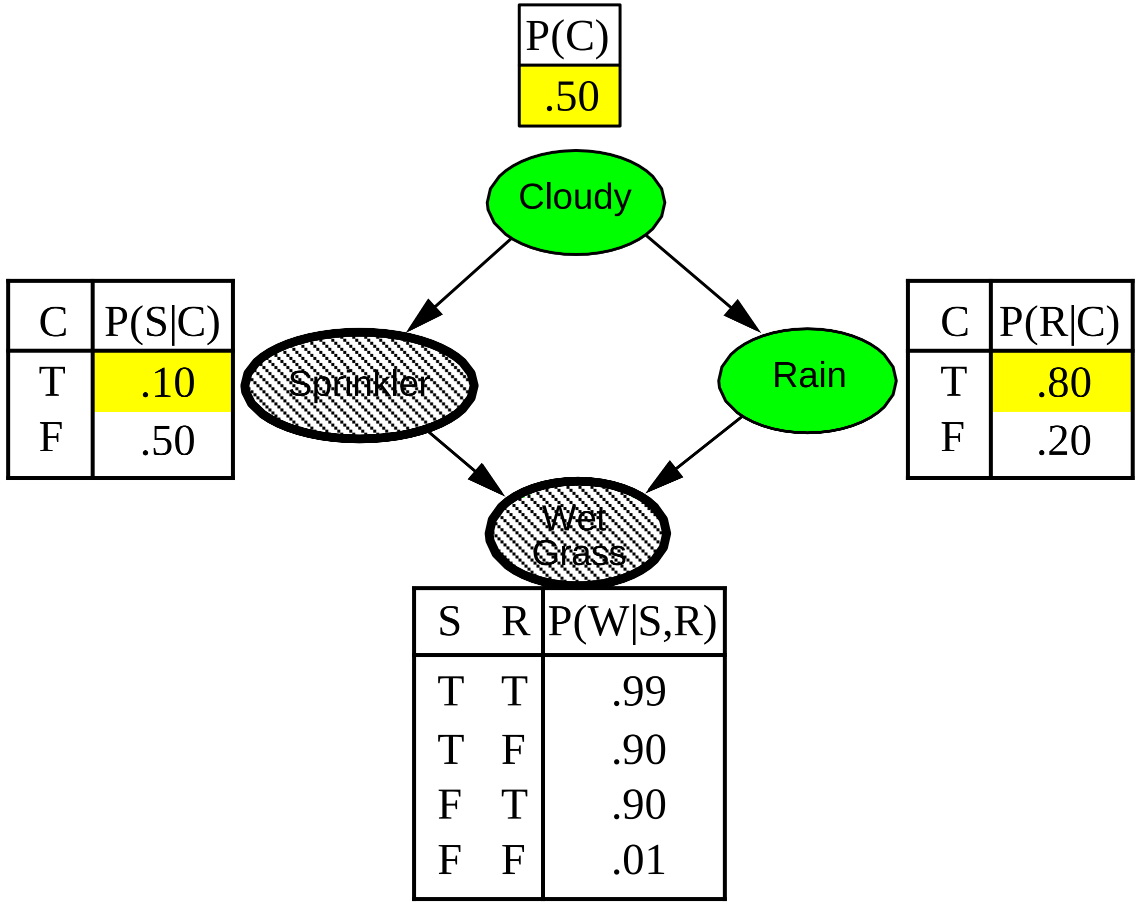
可能性权重的例子

- $w = 1.0 \times 0.1$



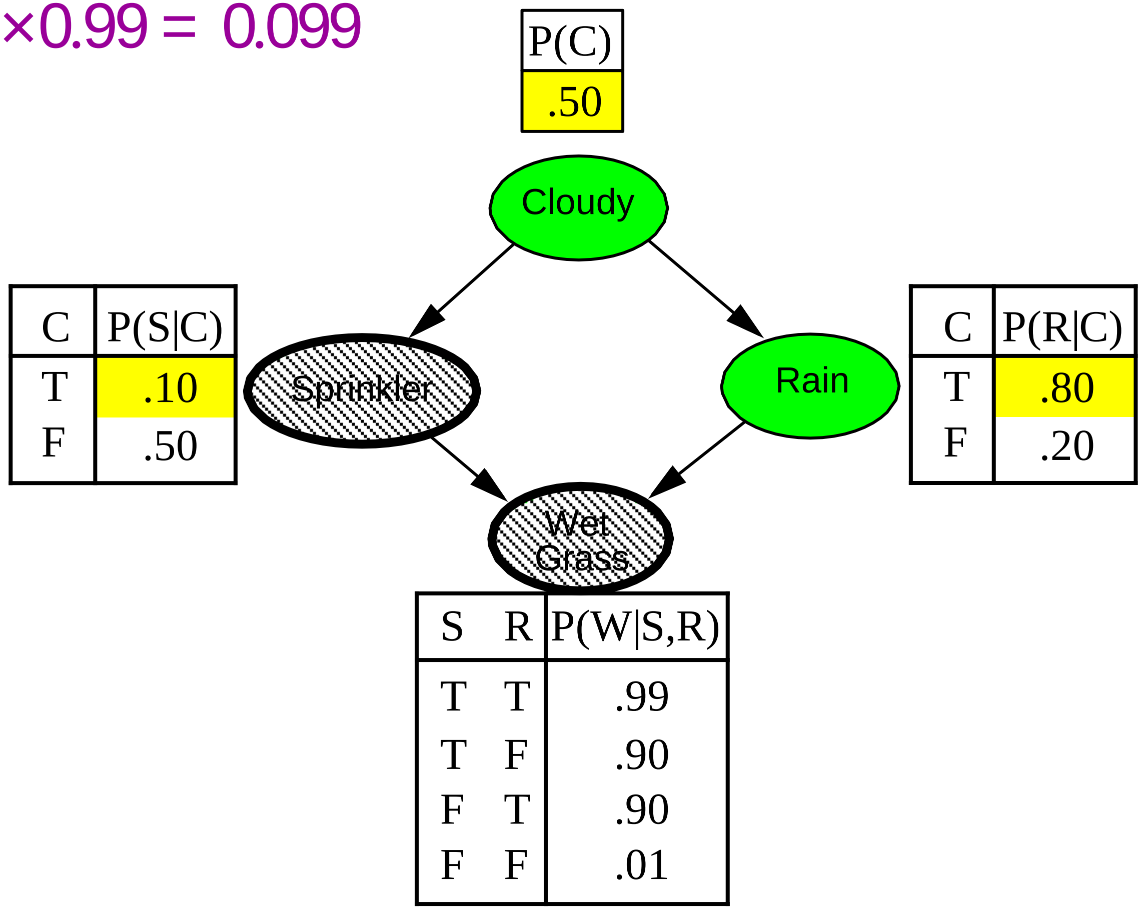
可能性权重的例子

- $w = 1.0 \times 0.1$



可能性权重的例子

- $w = 1.0 \times 0.1 \times 0.99 = 0.099$



可能性权重分析

- 加权样本的抽样概率为

$$S_{ws}(\mathrm{z,e}) = \prod_{i=1}^{l} P(z_i \mid parents(Z))$$

- 注意：只注意祖先的证据
  - 介于前后分布之间
- 对于每个给定的采样值z, e的权重是：

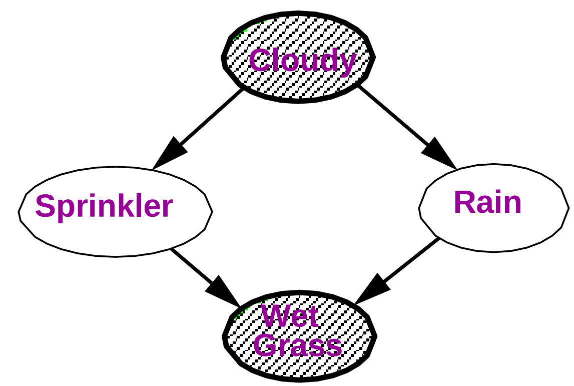
$$w(z,\mathrm{e}) = \prod_{i=1}^{m} P(e_i \mid parents(E_i))$$

- 加权抽样概率为

$$S_{ws}(\mathrm{z,e})\, w(z,\mathrm{e}) = \prod_{i=1}^{l} P(z_i \mid parents(Z)) \prod_{i=1}^{m} P(e_i \mid parents(E_i))$$

$= P(\mathrm{z, e})$ （通过网络的标准全局语义）

- 因此，可能性加权的估计值和实际值是一致的
- 但是，由于存在许多证据变量，性能仍然会下降
- 因为几个样品几乎占了全部w值



使用MCMC进行近似推理

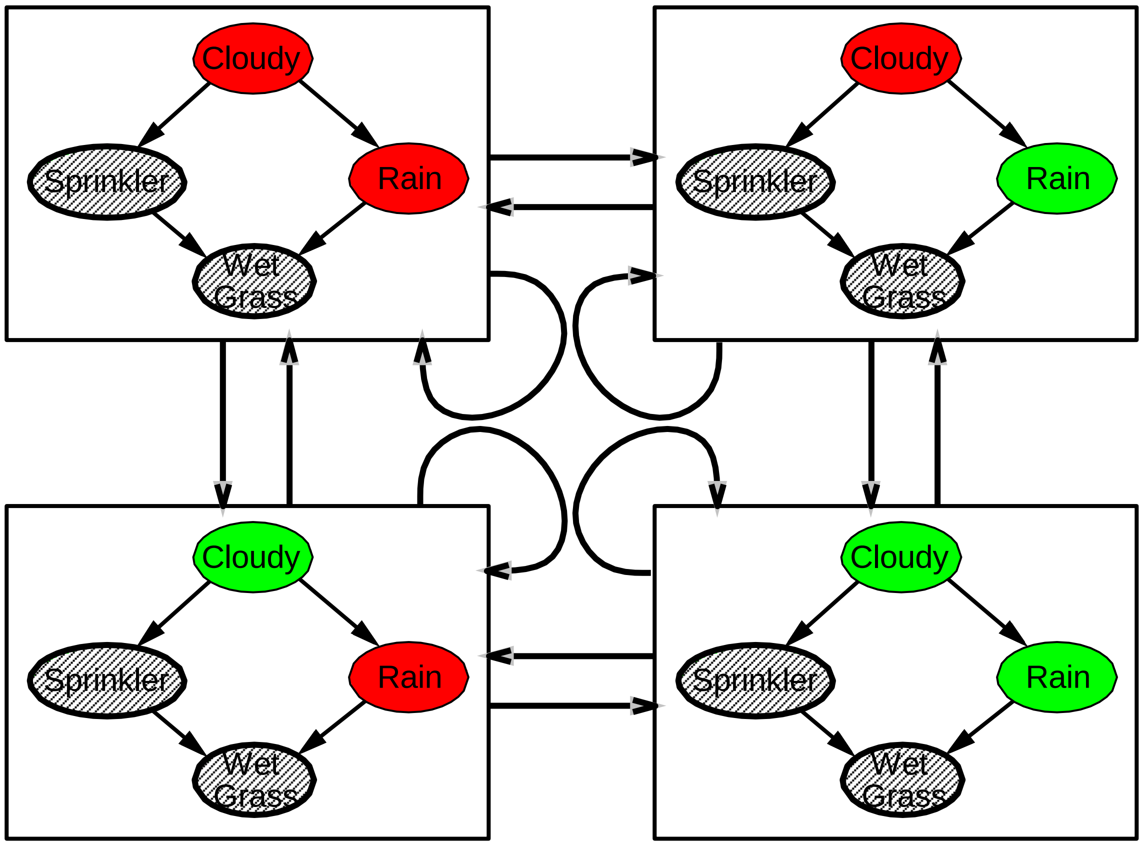
- 网络的“状态”=当前对所有变量的赋值。
- 通过对一个给定的变量进行马尔科夫毯抽样个，依次对每个变量进行抽样，保持证据固定，从而生成下一状态

```
function MCMC-Ask(X, e, bn, N) returns an estimate of P(X|e)
    local variables: N[X], a vector of counts over X, initially zero
                     Z, the nonevidence variables in bn
                     x, the current state of the network, initially copied from e
    initialize x with random values for the variables in Y
    for j = 1 to N do
        for each Z_i in Z do
            sample the value of Z_i in x from P(Z_i | mb(Z_i))
                given the values of MB(Z_i) in x
            N[x] ← N[x] + 1 where x is the value of X in x
    return Normalize(N[X])
```

- 每次还可以选择一个变量随机抽样

马尔科夫链

- 在$Sprinkler = true, WetGrass = true$下，有四种状态



MCMC实例

- 估计 $P(Rain|Sprinkler = true, WetGrass = true)$
- 对 $Cloudy$ 和 $Rain$ 反复采样可以得到马尔科夫毯
- 之后在样本中统计$Rain$的实际次数
- 例如访问100 个地区
  - 31 个$Rain = true$, 69 个$Rain = false$

$$P(Rain|Sprinkler = true, WetGrass = true)$$
$$= \mathrm{Normalize}((31, 69)) = (0.31, 0.69)$$

- 定理：链趋于平稳分布
- 长期运行后，每个地区统计结果的比值将和后验概率接近

马尔科夫毯抽样

- 关于$Cloudy$的马尔科夫毯是$Sprinkler$ 和 $Rain$
- 关于$Rain$的马尔科夫毯是$Cloudy$, $Sprinkler$, 和 $WetGrass$
- 马尔可夫毯的概率计算公式如下：

$$P(x_i'|mb(X_i)) = P(x_i'|parents(X_i)) \prod_{Z_j \in Children(X_i)} (z_j|parents(Z_j))$$

- 这种方法在并行传输系统下实现很容易
- 计算方面主要的问题有两个：
  - 难以判断计算是否完成
  - 如果马尔可夫毯很大，可能会造成浪费：$P\,(X_i|mb(X_i))$ 变化不大(大数定律)

小结

- 变量消除的精确推理：
  - 多重树（polytrees）上：多项式时间
  - 一般图上：NP-hard
  - 空间=时间，对拓扑结构非常敏感
- LW和MCMC的近似推理：
  - 当有大量(下游)证据时，LW表现不佳
  - LW, MCMC一般对拓扑结构不敏感
  - 收敛可能非常缓慢，概率接近1或0
  - 可以处理离散和连续变量的任意组合