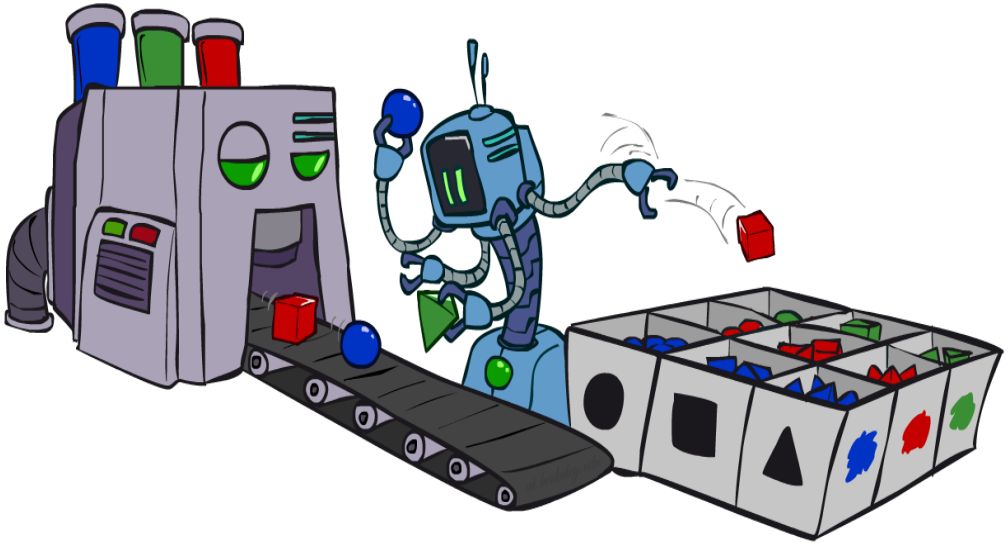
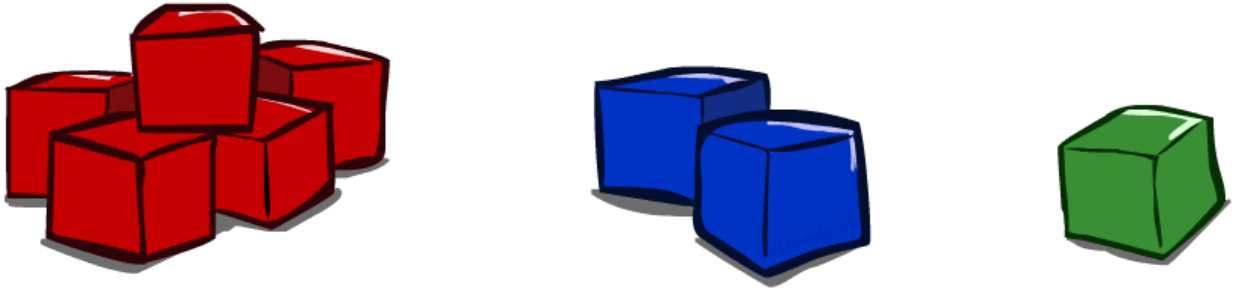
人工智能导论

贝叶斯网络: 近似推理APPROXIMATE INFERENCE



近似推理：采样

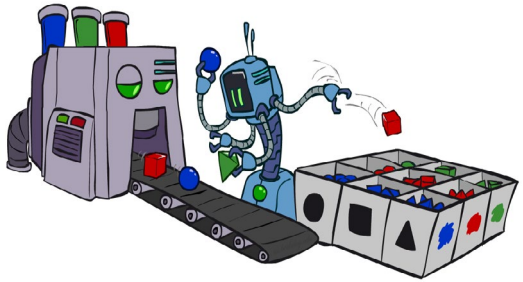


采样（Sampling）

采样很像重复的模拟

- 基本思想
  - 抽取 *N* 样本，形成一个采样分布 *S*
  - 计算一个近似后验概率
  - 证明可以收敛到真实的概率 *P*

- 为什么采样？
  - 通常很快得到一个好的近似解
  - 算法简单而且通用 (很容易应用在不同的概率模型上)
  - 算法只需很少的存储空间 (O(n))
  - 可以应用于大的模型上；对比准确算法（比如变量消除法）



从一个离散分布中采样

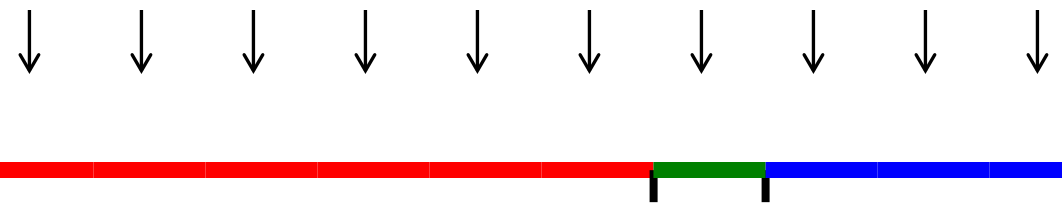
- 步骤 1: 获取一个采样 *u* 从均匀分布 [0, 1)
  - 例如 random()
- 步骤 2: 把这个采样值 *u* 转化成一个给定分布的输出结果。（通过关联每个输出结果 *x* 和一个 *P*(*x*)-大小的在[0,1) 上的一个子区间）

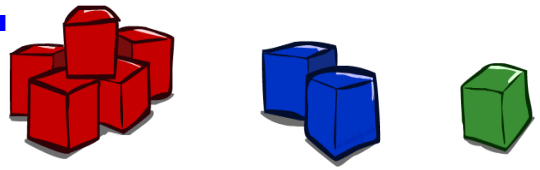
- 举例

C	P(C)
red	0.6
green	0.1
blue	0.3

$0.0 \le u < 0.6, \rightarrow C=red$
$0.6 \le u < 0.7, \rightarrow C=green$
$0.7 \le u < 1.0, \rightarrow C=blue$

- 如果 random() 返回 $u = 0.83$, 那么采样为 C = blue
- 再例如, 在8 次采样以后有:





贝叶斯网络里的采样

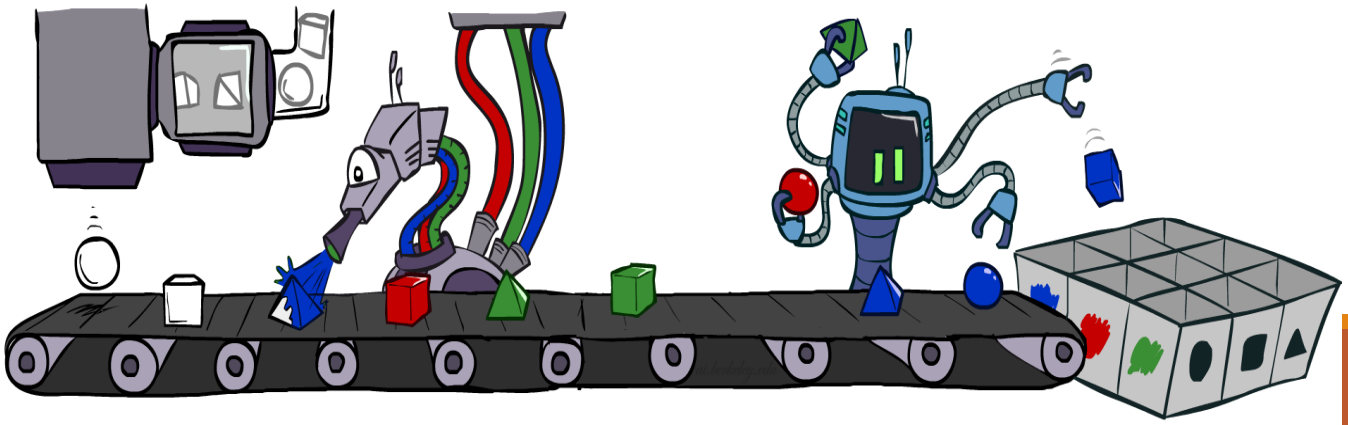
- 先验采样(Prior Sampling)
- 拒绝抽样(Rejection Sampling)
- 似然性/可能性加权(Likelihood Weighting)
- 吉布斯采样(Gibbs Sampling)

先验采样

For i=1, 2, …, n (按拓扑顺序)

- 采样 X_i 从 $P(X_i \mid parents(X_i))$

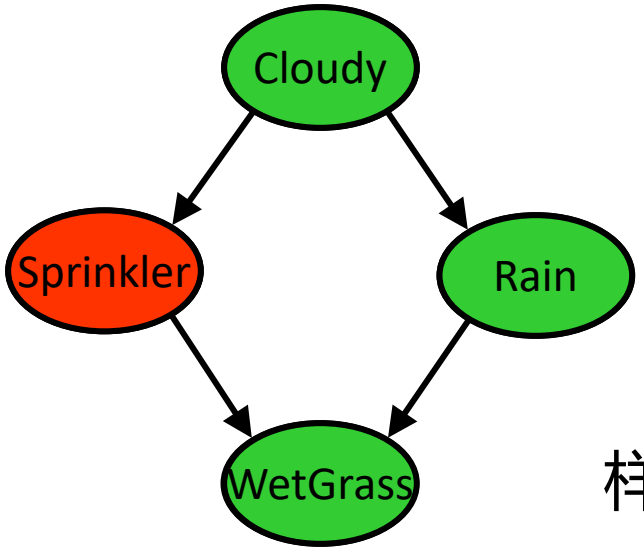
Return $(x_1, x_2, \ldots, x_n)$



先验采样

$P(C)$

c	0.5
¬c	0.5



$P(S|C)$

c	s	0.1
	¬s	0.9
¬c	s	0.5
	¬s	0.5

$P(R|C)$

c	r	0.8
	¬r	0.2
¬c	r	0.2
	¬r	0.8

$P(W|S,R)$

s	r	w	0.99
		¬w	0.01
	¬r	w	0.90
		¬w	0.10
¬s	r	w	0.90
		¬w	0.10
	¬r	w	0.01
		¬w	0.99

样本:

c, ¬s, r, w(这个例子里)

¬c, s, ¬r, w

...

先验采样

- 这个过程产生这样的样本的概率是:

$$S_{PS}(x_1 \ldots x_n) = \prod_{i=1}^{n} P(x_i | \text{Parents}(X_i)) = P(x_1 \ldots x_n)$$

…即 是贝叶斯网络的联合概率

- 让 一个事件的样本数为 $N_{PS}(x_1 \ldots x_n)$
- 那么

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

- 即, 这个采样过程是 一致的/连续的(consistent)

例如

我们从这个贝叶斯网络里获得一系列的样本:

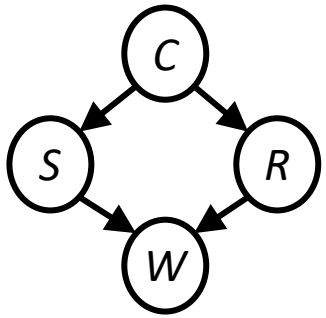
c, ¬s, r, w

c, s, r, w

¬c, s, r, ¬w

c, ¬s, r, w

¬c, ¬s, ¬r, w



如果我们想知道： P(*W*)

- 我们可以数出 <w:4, ¬w:1>
- 正规化后得到 P(*W*) = <w:0.8, ¬w:0.2>
- 样本越多，越接近真实的分布
- 还可以估计其他的概率量
- 比如, 想查询概率 P(*C*| *r, w*)， 使用 P(*C*| *r, w*) = α P(*C*, *r, w*)

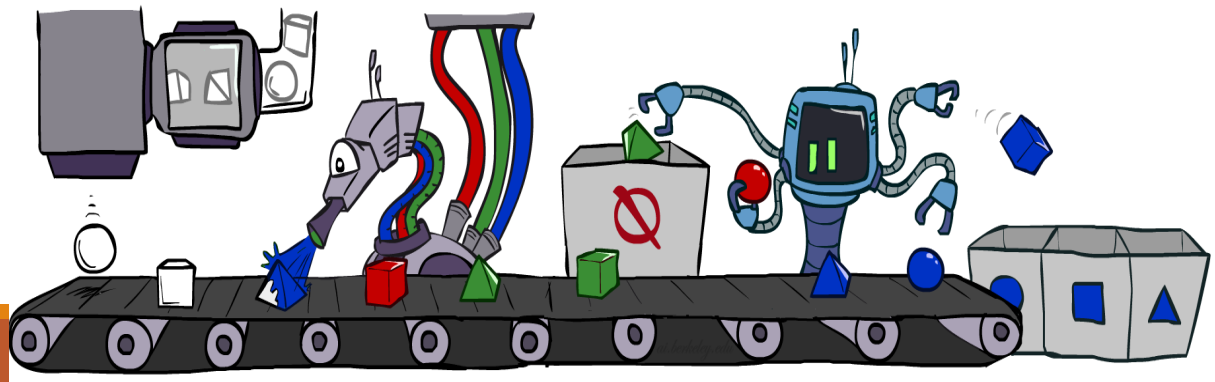
拒绝采样Rejection Sampling

输入: 观察值 $e_1,..,e_k$

For i=1, 2, …, n

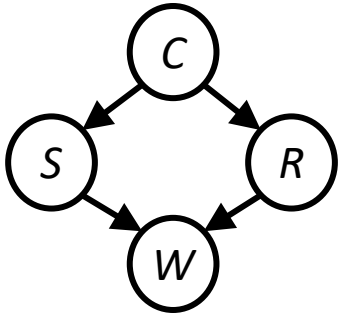
- 采样 X_i 从 $P(X_i \mid parents(X_i))$
- 如果 x_i 和观察值不一致
  - 拒绝这个样本: Return, 则在这个循环里没有样本产生

Return $(x_1, x_2, \ldots, x_n)$



拒绝采样

- 为了计算条件概率，对先验采样进行简单修改
- 假如我们想计算 P(*C*| *r, w*)
- 计算采样中 C 的结果, 但是忽略（拒绝） 那些不含有 *R*=true, *W*=true的样本
  - 这就叫做拒绝采样
  - 对于条件概率的估计，也是满足一致性的 (即, N趋于无限大时，等于理论真值)



c, ¬s, r, w

~~c, s, ¬r~~

~~¬c, s, r, ¬w~~

~~c, -s, ¬r~~

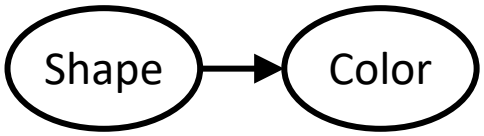
¬c, ¬s, r, w

似然性加权（采样）

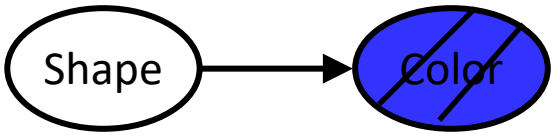
拒绝采样法的问题:

- 有可能拒绝许多样本，尤其当观察变量很多时
- 采样时没有利用已被观察变量的值
- 比如考虑 P(*Shape*|*Color=blue*)

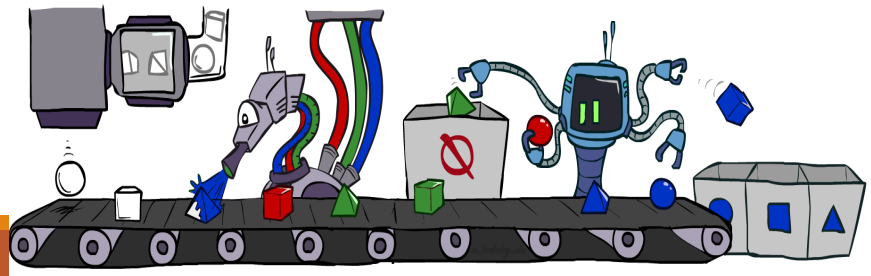
- 想法: 固定观察变量的值，对其他变量值进行采样
  - 问题: 样本分布与理论分布不一致!
  - 解决办法: ***权重*** 每个样本，通过使用观察变量给定父变量的概率

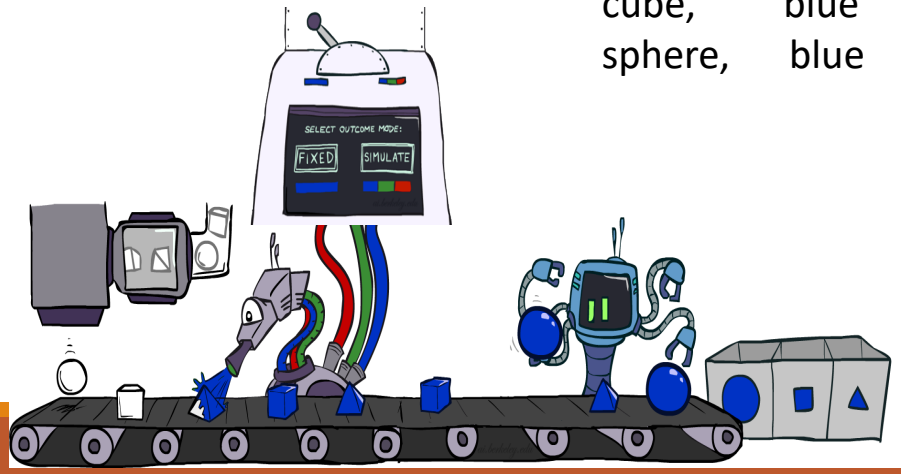


~~pyramid, green~~
~~pyramid, red~~
sphere, blue
~~cube, red~~
~~sphere, green~~



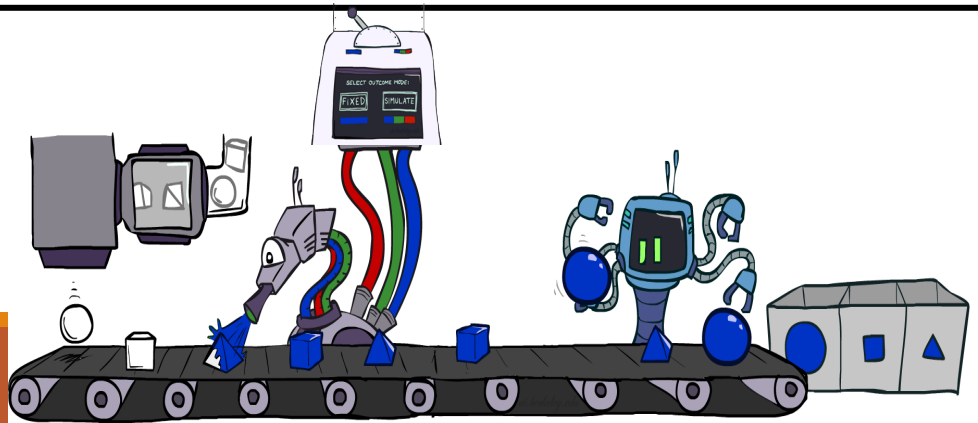
pyramid, blue
pyramid, blue
sphere, blue
cube, blue
sphere, blue





似然性加权采样

- 输入: 观察值 $e_1,..,e_k$
- w = 1.0
- for i=1, 2, …, n
  - 如果 X_i 是已观察的变量（evidence variables）
    - x_i = 观察到的 value$_i$ for X_i
    - 让 w = w * P(x_i | Parents(X_i))
  - 否则
    - 抽样 x_i 从 P(X_i | Parents(X_i))
- return (x_1, x_2, …, x_n), w

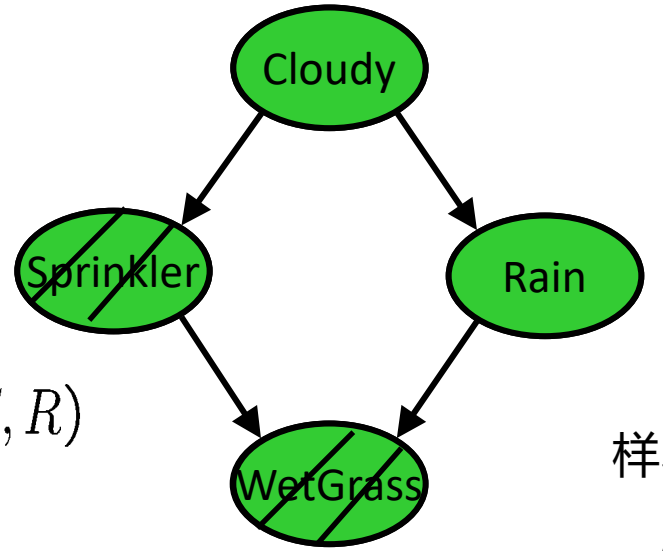


似然性加权（采样）

w 初始化为1.0；
拓扑排序： C, S, R,W
S, W 值固定为真

$P(C)$

c	0.5
¬c	0.5



$P(S|C)$

c	s	0.1
	¬s	0.9
¬c	s	0.5
	¬s	0.5

$P(R|C)$

c	r	0.8
	¬r	0.2
¬c	r	0.2
	¬r	0.8

$P(W|S,R)$

s	r	w	0.99
		¬w	0.01
	¬r	w	0.90
		¬w	0.10
¬s	r	w	0.90
		¬w	0.10
	¬r	w	0.01
		¬w	0.99

样本事件:

c, s, r, w

…

该样本事件的权值:

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

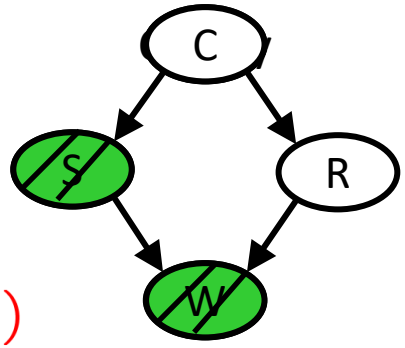
似然性加权采样

- 采样分布为（ z 为非观察变量的采样值 e 为固定的观察值）

$$S_{WS}(\mathbf{z},\mathbf{e}) = \prod_{i=1}^{l} P(z_i|\text{Parents}(Z_i))$$

- 现在, 每个样本都有权重

$$w(\mathbf{z},\mathbf{e}) = \prod_{i=1}^{m} P(e_i|\text{Parents}(E_i))$$



- 合起来, 加权的样本分布是具有一致性的，即：

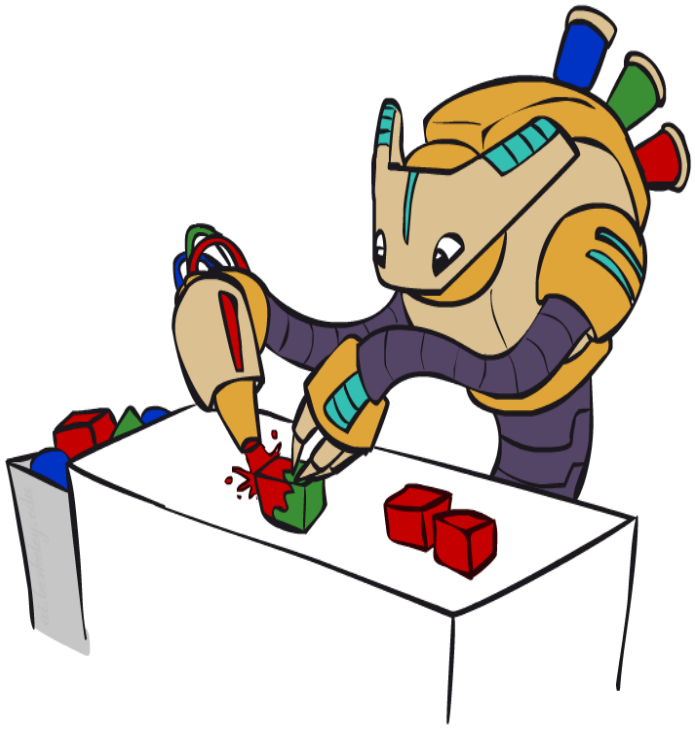
$$S_{\text{WS}}(z,e)\cdot w(z,e) = \prod_{i=1}^{l} P(z_i|\text{Parents}(z_i)) \prod_{i=1}^{m} P(e_i|\text{Parents}(e_i))$$

$$= P(\mathbf{z},\mathbf{e})$$

似然性加权

- 优点：
  - 可利用所有样本（加权的）
  - *下游* 变量的采样值会被 *上游* 已观察变量的值所影响（当观察变量处在网络上游的时候）
- 也有弱点：
  - *上游* 变量 的采样值 不受 *下游* 观察变量值的影响
  - 假设观察到的值在 k 个叶节点上，那么样本的权重可能为 $O(2^{-k})$
  - 随着观察变量的增多，而且如果这些变量出现在拓扑顺序的后面，那么许多样本的权值会很小，只有极少的幸运样本将有相对很大的权值，从而主导估计概率的结果
- 我们希望的是，每个变量都可以“看见”*所有* 已观察到的值！

吉布斯采样(Gibbs Sampling)



吉布斯采样 （Gibbs sampling）

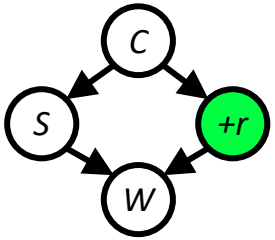
- 属于 MCMC 家族一类
  - 状态是对所有变量的完整的赋值
    - (对比局部搜索里的 模拟退火算法，属于同一算法家族!)
  - 观察（证据）变量的值固定，改变其他变量的值
  - 当产生下一个状态时, 选出一个变量，并对其采样一个值，采样的分布是条件于所有其他变量
    - $X_i' \sim P(X_i \mid x_1,..,x_{i-1},x_{i+1},..,x_n)$
    - 趋向于朝高概率发生的状态移动，但也可能移动到一个低概率的状态
    - 在贝叶斯网络里, $P(X_i \mid x_1,..,x_{i-1},x_{i+1},..,x_n) = P(X_i \mid 马可夫毯(X_i))$
- 定理: 吉布斯采样是具有一致性的
  - 如果假定吉布斯分布概率是远离0和1，并且变量选择是公平的

Gibbs Sampling吉布斯采样

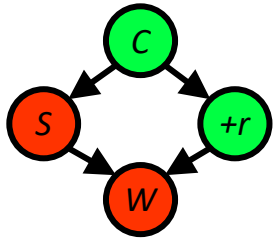
- *Procedure:* keep track of a full instantiation $x_1, x_2, \ldots, x_n$. 固定观察变量的值，给其他变量进行随机赋值。 每次选择一个未观察变量，并对它进行采样给定所有其他变量的赋值, 但是保持观察变量的值固定. 重复这一过程很长时间.
- *Property:* in the limit of repeating this infinitely many times 结果的样本将与条件于观察值的分布相一致.
- *Rationale*: 对上游和下游的变量的采样都将基于观察值.
- In contrast: likelihood weighting only conditions on upstream evidence, and hence weights obtained in likelihood weighting can sometimes be very small. Sum of weights over all samples is indicative of how many "effective" samples were obtained, so we want high weight.

Gibbs Sampling Example: P(S | +r)

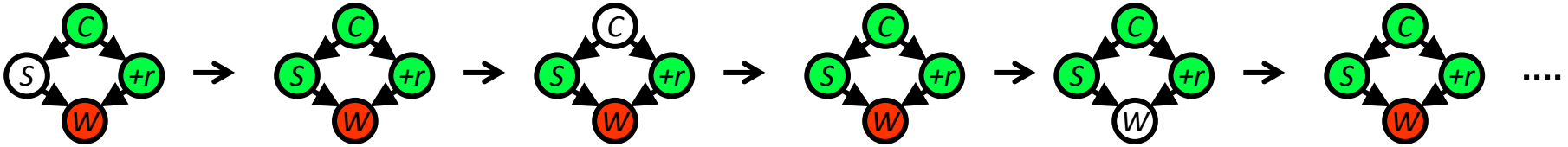
- Step 1: Fix evidence
  - R = +r



- Step 2: Initialize other variables
  - Randomly



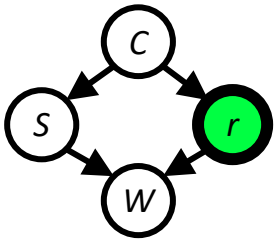
- Steps 3: Repeat
  - Choose a non-evidence variable X
  - Resample X from P(X | all other variables)



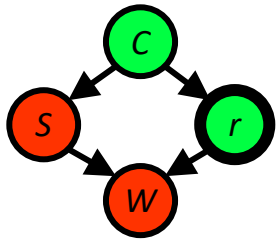
Sample from $P(S|+c,-w,+r)$ Sample from $P(C|+s,-w,+r)$ Sample from $P(W|+s,+c,+r)$

吉布斯采样举例: $P(S \mid r)$

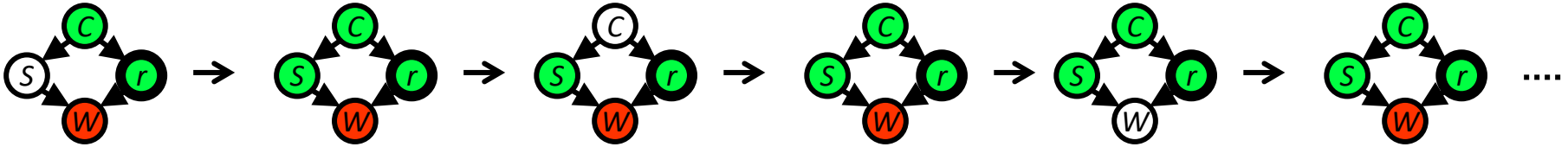
Step 1: 固定观察值
- R = true



- Step 2: 初始化其他变量
  - 随机地



Step 3: 重复以下
- 选择一个非证据变量 X
- 重采样 X 从 $P(X \mid 马可夫毯(X))$

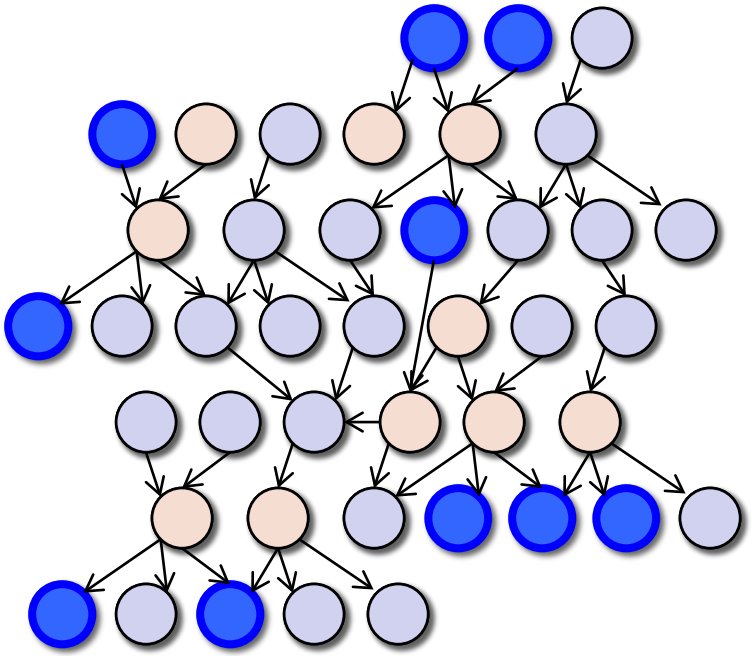


采样 $S \sim P(S \mid c, r, \neg w)$

采样 $C \sim P(C \mid s, r)$

采样 $W \sim P(W \mid s, r)$

为什么这样做？



采样很快开始反应网络里所有的观察值（已观察节点的值对其他变量值的采样施加影响）

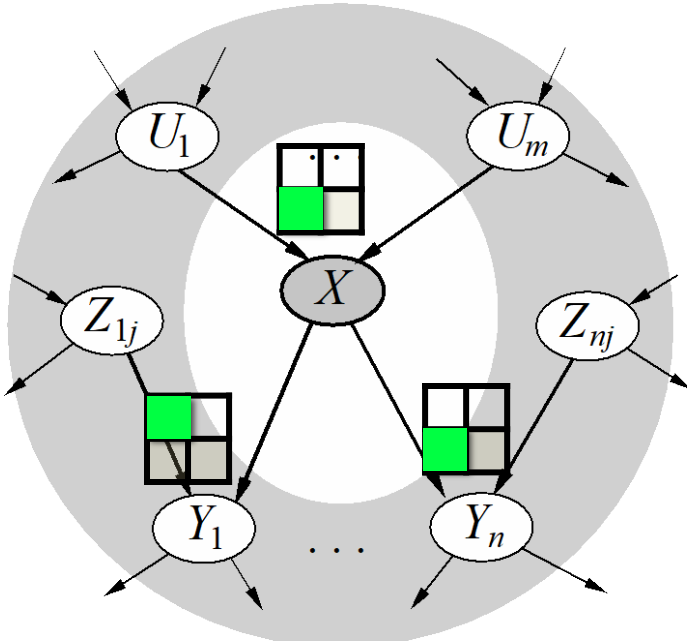
最终样本将从真实的后验概率分布上抽取!

Gibbs Sampling

- How is this better than sampling from the full joint?
  - In a Bayes' Net, sampling a variable given all the other variables (e.g. P(R|S,C,W)) is usually much easier than sampling from the full joint distribution
    - Only requires a join on the variable to be sampled (in this case, a join on R)
    - **The resulting factor only depends on the variable's parents, its children, and its children's parents (this is often referred to as its Markov blanket)**

如何进行采样？

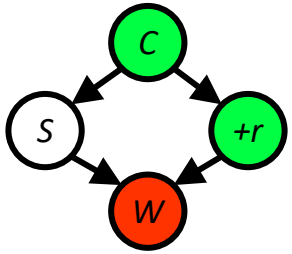
- 重复以下多次：
  - 对一个非观察到的变量 X_i 进行采样，从概率分布：
  - $P(X_i \mid x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) = P(X_i \mid \text{马尔科夫毯}(X_i))$
  - $= \alpha P(X_i \mid u_1, \ldots, u_m) \prod_j P(y_j \mid parents(Y_j))$



对一个变量的快速重采样

- Sample from P(S | +c, +r, -w)

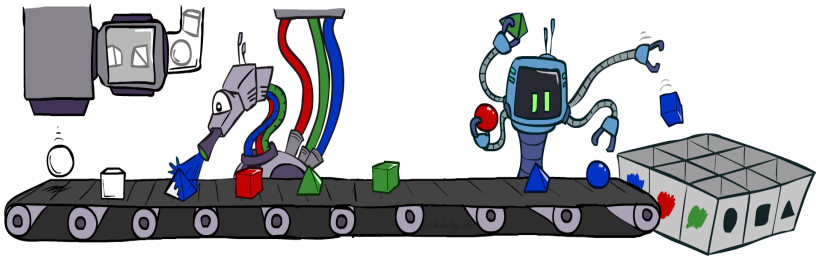
$$
\begin{aligned}
P(S|+c,+r,-w) &= \frac{P(S,+c,+r,-w)}{P(+c,+r,-w)} \\
&= \frac{P(S,+c,+r,-w)}{\sum_s P(s,+c,+r,-w)} \\
&= \frac{P(+c)P(S|+c)P(+r|+c)P(-w|S,+r)}{\sum_s P(+c)P(s|+c)P(+r|+c)P(-w|s,+r)} \\
&= \frac{P(+c)P(S|+c)P(+r|+c)P(-w|S,+r)}{P(+c)P(+r|+c)\sum_s P(s|+c)P(-w|s,+r)} \\
&= \frac{P(S|+c)P(-w|S,+r)}{\sum_s P(s|+c)P(-w|s,+r)}
\end{aligned}
$$



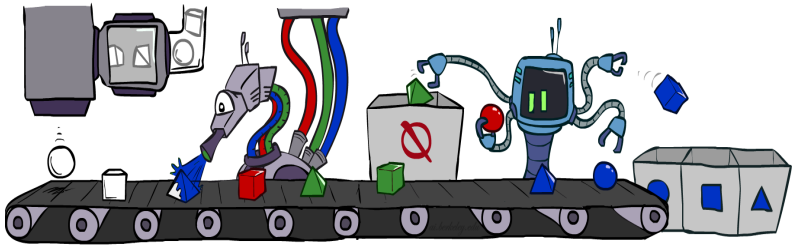
- Many things cancel out – only CPTs with S remain!
- More generally: only CPTs that have resampled variable need to be considered, and joined together

贝叶斯网络采样技术小结

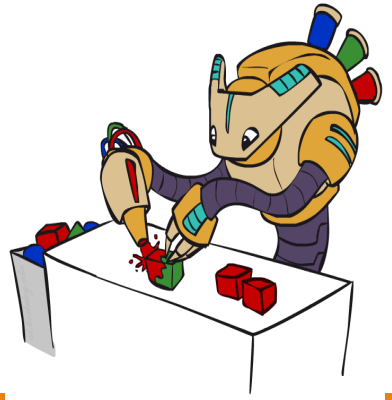
- 先验采样 P



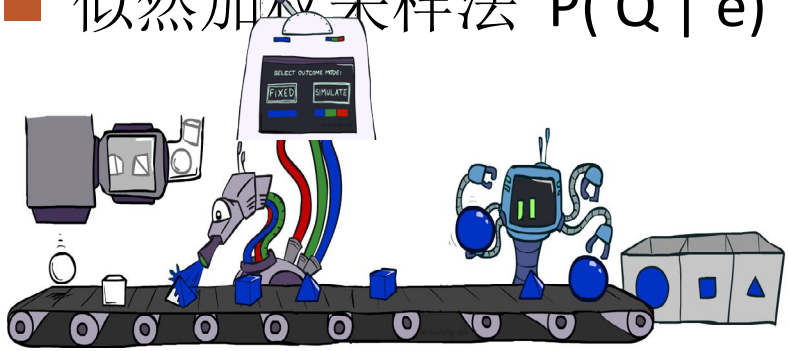
- 拒绝采样法 P(Q | e)



- 吉布斯采样 P(Q | e)



- 似然加权采样法 P(Q | e)



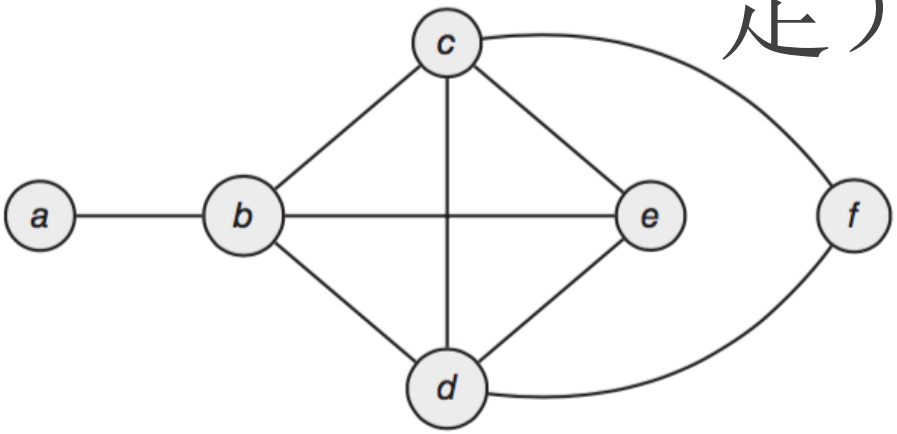
马尔科夫蒙特卡洛 理论 Markov Chain Monte Carlo

- MCMC 属于随机算法家族，用于在一个很大的状态空间里，近似估计某些感兴趣的量
  - 马尔科夫链 = 一序列随机选择的状态 (“随机漫步random walk”), 每个状态的选择是条件取决于前一个状态
  - 蒙特卡洛理论 Monte Carlo = 一种算法 (通常基于随机采样)，存在产生一个不正确解答的可能性（概率）
- MCMC = 随机漫步一会，平均化你所观察到的情况

为什么这种方法有效?

- 假定运行这种方法很长一段时间，并预测在时刻 t 到达任何一个状态的概率为： $\pi_t(x_1,\ldots,x_n)$ or $\pi_t(\mathbf{x})$
- 对每个吉布斯采样步骤 (挑一个变量, 重采样它的值) 当它应用到一个状态 $\mathbf{x}$ 时，有一个概率 $q(\mathbf{x}' \mid \mathbf{x})$ 移动到下个状态 $\mathbf{x}'$
- 所以 $\pi_{t+1}(\mathbf{x}') = \sum_{\mathbf{x}} q(\mathbf{x}' \mid \mathbf{x})\, \pi_t(\mathbf{x})$ 或, 用矩阵或向量形式表示: $\pi_{t+1} = Q\pi_t$
- 当这一动态过程处于平衡，即 $\pi_{t+1} = \pi_t$ ，所以 $Q\pi_t = \pi_t$
- 这种情况下有一个唯一解，即 $\pi_t = P(x_1,\ldots,x_n \mid e_1,\ldots,e_k)$
- 所以当时刻 t 足够大时，下一个样本将会从真实的后验条件概率分布上被采集

随机漫步（随机游走）(Random Walk)



$$\boldsymbol{P} = \begin{array}{c} \\ a \\ b \\ c \\ d \\ e \\ f \end{array} \begin{array}{c} \begin{array}{cccccc} a & b & c & d & e & f \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1/4 & 0 & 1/4 & 1/4 & 1/4 & 0 \\ 0 & 1/4 & 0 & 1/4 & 1/4 & 1/4 \\ 0 & 1/4 & 1/4 & 0 & 1/4 & 1/4 \\ 0 & 1/3 & 1/3 & 1/3 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 & 0 & 0 \end{pmatrix} \end{array}.$$

随机漫步（随机游走）(Random Walk)

- 随机漫步的过程
- 强连接的图
- $\mathbf{p}^T(t)\mathrm{P} = \mathbf{p}^T(t+1)$
- 基本性质
- 统计学里，也叫马科夫链(Markov Chain)
- 例如：WWW的搜索算法 PageRank
  - 网页的排序根据它们的静态概率

Further Reading on Gibbs Sampling*

- Gibbs sampling produces sample from the query distribution P(Q | e) in limit of re-sampling infinitely often
- Gibbs sampling is a special case of more general methods called Markov chain Monte Carlo (MCMC) methods
  - Metropolis-Hastings is one of the more famous MCMC methods (in fact, Gibbs sampling is a special case of Metropolis-Hastings)
- You may read about Monte Carlo methods – they're just sampling