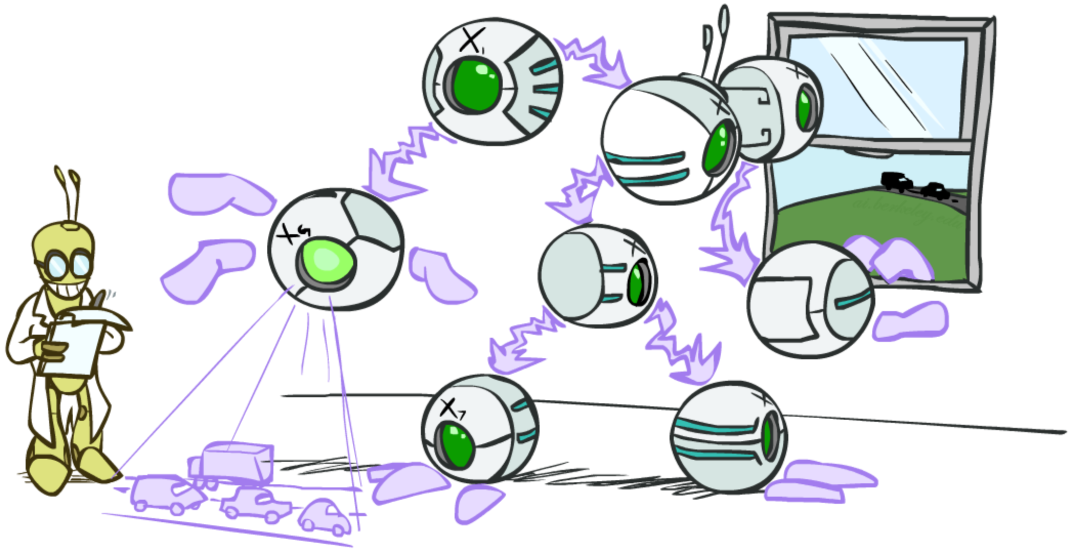
COMS W4701: Artificial Intelligence

Lecture 19: Inference and Sampling



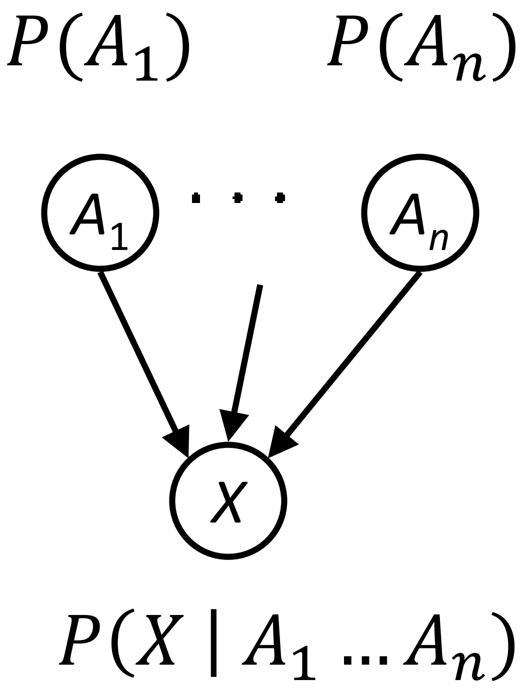
Instructor: Tony Dear

*Lecture materials derived from UC Berkeley's AI course at ai.berkeley.edu

Review: Bayesian Networks

- Joint distribution: directed acyclic graph
- Nodes: Random variables (with domains)
- Arcs: Correlation or influence between variables

- Each node encodes a conditional probability distribution based on its parents



$$P(x_i \mid x_1, \ldots, x_{i-1}) = P(x_i \mid parents(X_i))$$

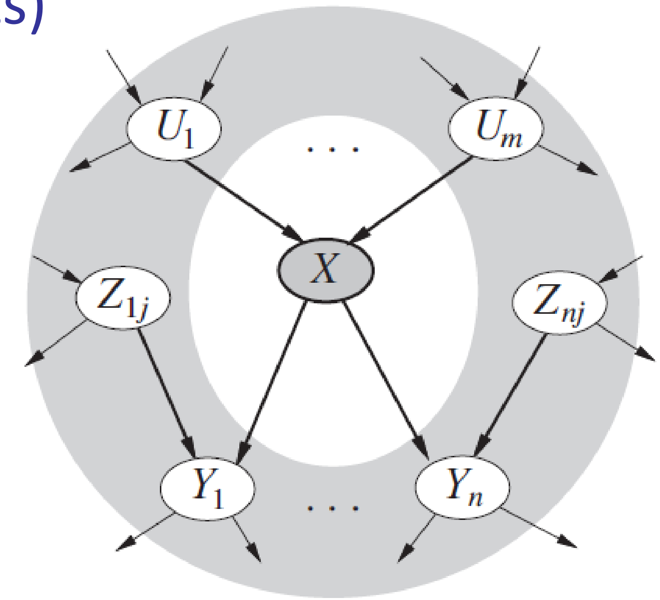
$$P(x_1, \ldots, x_n) = P(x_1) \prod_{i=2}^{n} P(x_i \mid x_1, \ldots, x_{i-1}) = \prod_{i=1}^{n} P(x_i \mid parents(X_i))$$

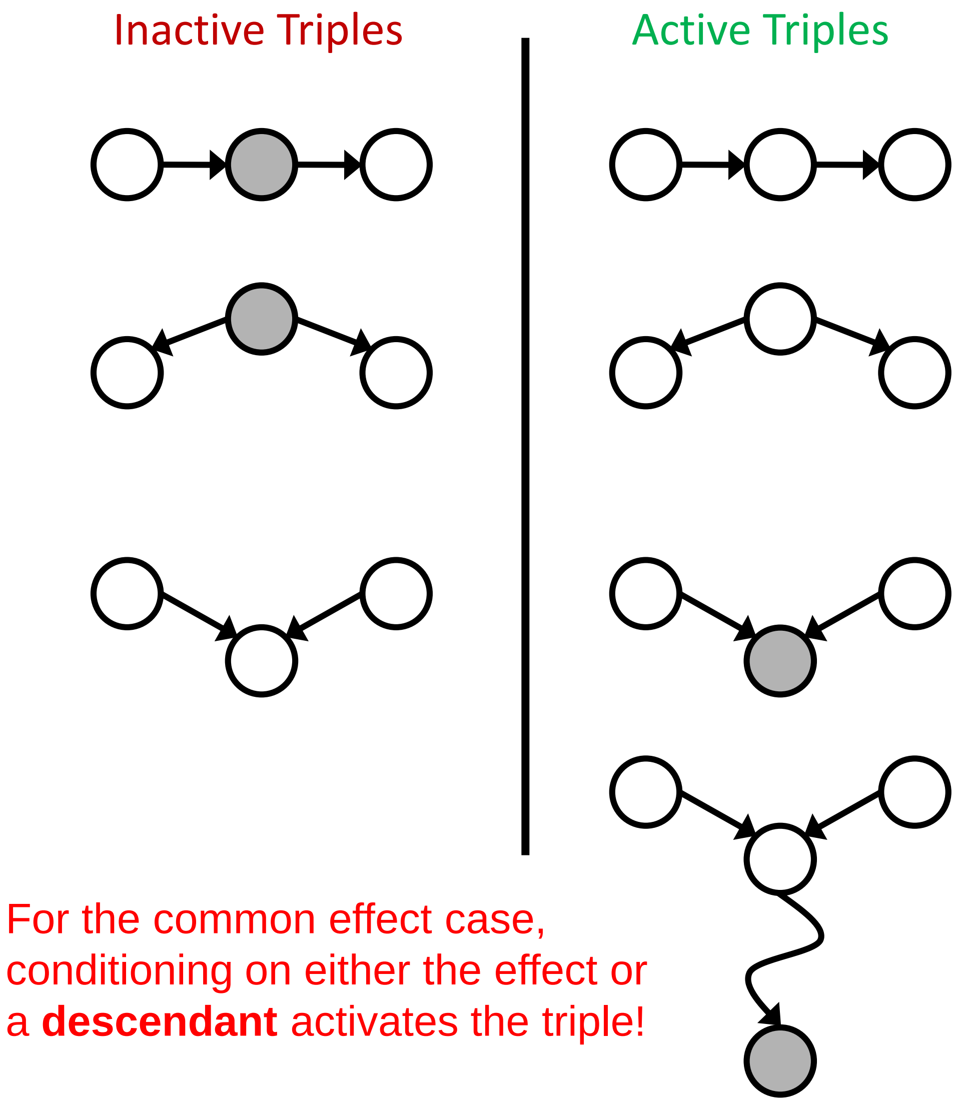
Conditional Independence in BNs

- Two RVs are (conditionally) independent if *all paths* between their nodes contain **inactive triples**

- Corollary: A RV is conditionally independent of the rest of the BN given its **Markov blanket** (parents, children, children's parents)

- $\text{Parents}(X) = \{U_k\}$
- $\text{Children}(X) = \{Y_i\}$
- $\text{Parents}(Y_i) = \{Z_{ij}\}$





Today

- Exact inference: Inference by enumeration

- Approximate inference: Monte Carlo methods
- Rejection sampling
- Likelihood weighting
- Gibbs sampling

Exact Inference

- We want to find $P(\boldsymbol{X} \mid \boldsymbol{e})$
- Query variables $\boldsymbol{X}$; evidence variables $\boldsymbol{e}$; hidden variables $\boldsymbol{Y}$

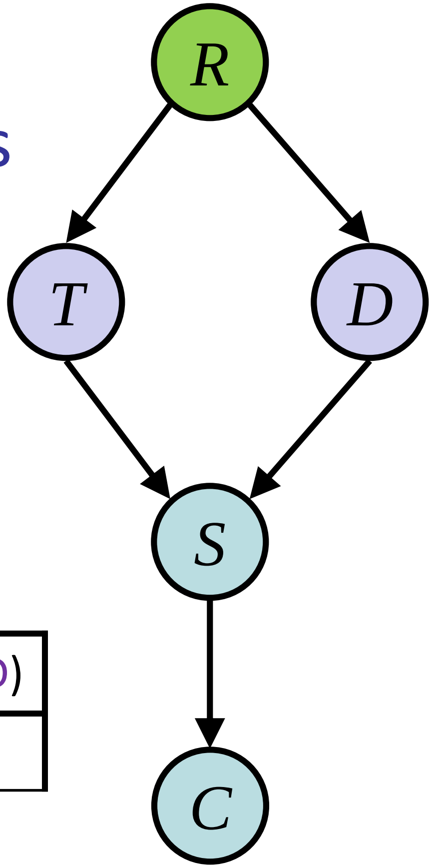
- Enumeration strategy: Construct joint distributions using **chain rule**, apply **conditional independences**, and **marginalize** out hidden variables

$$P(\boldsymbol{X} \mid \boldsymbol{e}) = \alpha P(\boldsymbol{X}, \boldsymbol{e}) = \alpha \sum_{\boldsymbol{y}} P(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{e})$$

- This is not an easy task!!!
- Intermediate joint distributions grow exponentially

Example: Exact Inference

- Query variables $\boldsymbol{X}$; evidence variables $\boldsymbol{e}$; hidden variables $\boldsymbol{Y}$
- When *joining* distributions, *pointwise multiply* matching rows



$$P(R|s,c) = P(R|s) \propto P(R,s) = \sum_{t,d} P(R,t,d,s)$$

Conditional independence

$$= \sum_{t,d} P(R)P(t|R)P(d|R)P(s|t,d)$$

R	P(R)
2 rows	

×

R	T	P(T|R)
4 rows		

×

R	D	P(D|R)
4 rows		

×

T	D	P(s|T,D)
4 rows		

→

R	T	D	P(R,T,D,s)
8 rows			

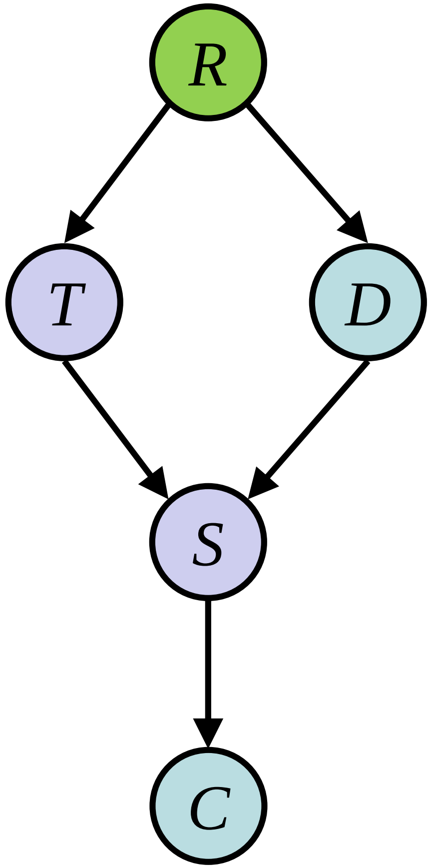
$\sum_{t,d}$ →

R	P(R,s)
2 rows	

Example: Exact Inference

$$P(R|+c,+d) \propto \sum_{t,s} P(R)P(t|R)P(+d|R)P(s|t,+d)P(+c|s)$$

R	P(R)
+r	0.5
-r	0.5



R	T,D	P(T|R), P(D|R)
+r	+t,+d	0.7
-r	+t,+d	0.6

T	D	S	P(S|T,D)
+t	+d	+s	0.1
+t	-d	+s	0.4
-t	+d	+s	0.2
-t	-d	+s	0.9

S	C	P(C|S)
+s	+c	0.8
-s	+c	0.3

R	T	S	P(R,T,+d,S,+c)
+r	+t	+s	(0.5)(0.7)(0.7)(0.1)(0.8)
+r	+t	-s	(0.5)(0.7)(0.7)(0.9)(0.3)
+r	-t	+s	(0.5)(0.3)(0.7)(0.2)(0.8)
+r	-t	-s	(0.5)(0.3)(0.7)(0.8)(0.3)
-r	+t	+s	(0.5)(0.6)(0.6)(0.1)(0.8)
-r	+t	-s	(0.5)(0.6)(0.6)(0.9)(0.3)
-r	-t	+s	(0.5)(0.4)(0.6)(0.2)(0.8)
-r	-t	-s	(0.5)(0.4)(0.6)(0.8)(0.3)

$$\sum_{t,s}$$

R	P(R,+c,+d)
+r	0.12775
-r	0.111

Need to construct table with 2^3 rows and lots of repetitive information!

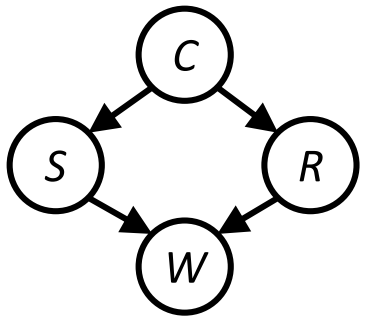
Approximate Inference: Sampling

- Complexity of computing probabilities grows with number of variables

- **Monte Carlo** methods: Repeated sampling from *known* probability distribution (e.g., Bayes net model probabilities) to estimate *unknown* distribution

- A sample from a joint distribution assigns a value to each RV—how to do so consistently?

- Idea: *Order* the RVs s.t. we can use all $P(X_i \mid parents(X_i))$!



Ordering: C, S, R, W

1. Assign C using $P(C)$
2. Assign S using $P(S|c)$
3. Assign R using $P(R|c)$
4. Assign W using $P(W|s, r)$

Ordering C, R, S, W also works

Prior Sampling

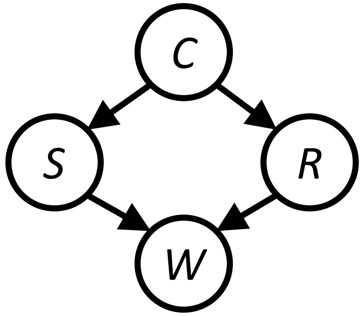
Order all nodes such that all parents of X_i occur before X_i

for $i = 1{:}n$

  Sample x_i from $P(X_i \mid \text{parents}(X_i))$ We have these!

return $\text{sample} = (x_1, x_2, \ldots, x_n)$

- Once sufficient samples are generated, inferences can be computed by simply counting samples corresponding to the query



- (+c, -s, +r, +w)
- (+c, +s, +r, +w)
- (-c, +s, +r, -w)
- (+c, -s, +r, +w)
- (-c, -s, -r, +w)

$\hat{P}(R)$

+r	0.8
-r	0.2

This is because +r occurs 4 times, and -r occurs once

+c, +w occur 3 times etc

$\hat{P}(C, W)$

+c	+w	0.6
	-w	0
-c	+w	0.2
	-w	0.2

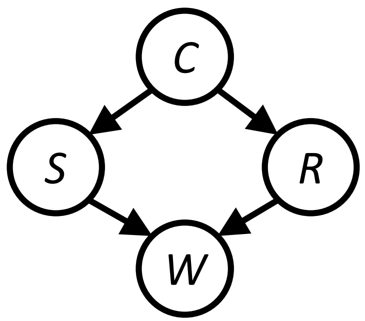
$\hat{P}(S|W)$

+w	+s	0.25
	-s	0.75
-w	+s	1
	-s	0

Rejection Sampling

- Counting samples can be done while generating them instead of all at the end
- If query contains evidence, many samples will often be irrelevant
- E.g., want $P(A|+b)$, all samples with $-b$ are useless to us!

- Idea: Discard irrelevant samples as they come and only count *consistent* ones



1. (+c, -s, +r, +w)
2. (+c, +s, +r, +w)
3. (-c, +s, +r, -w)
4. (+c, -s, +r, +w)
5. (-c, -s, -r, +w)

$P(C|+s)$

+c	0.5
-c	0.5

Reject 1, 4, 5

$P(R|-c)$

+r	0.5
-r	0.5

Reject 1, 2, 4

$P(S|+r,+w)$

+s	0.3
-s	0.7

Reject 3, 5

Rejection Sampling

Order all nodes such that all parents of X_i occur before X_i

for $i = 1{:}n$

  Sample x_i from $P(X_i \mid \text{parents}(X_i))$ We have these!

  if x_i not consistent with evidence:

    reject and **return** (no sample generated)

return $\text{sample} = (x_1, x_2, \ldots, x_n)$

- Problem: Lots of potentially wasted work due to rejected samples!
- As we condition on more and more evidence variables, fraction of consistent samples drops *exponentially*

Likelihood Weighting

Order all nodes such that all parents of X_i occur before X_i
Instantiate all evidence variables, weight $w = 1.0$
for $i = 1{:}n$
    if X_i is evidence variable:
        $w = w * P(x_i|parents(X_i))$ We have these!
    else: **Sample** x_i from $P(X_i \mid \text{parents}(X_i))$
return $\text{sample} = (x_1, x_2, \dots, x_n), w$

Update weight instead of sampling evidence

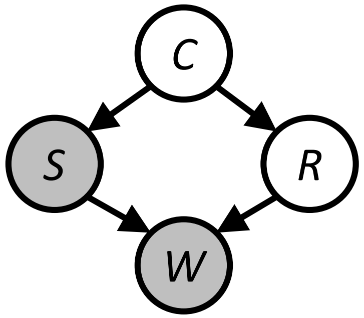
- Idea: **Fix** evidence variables to the values that we want
- But we don't want to purposely bias our samples, so compensate by **weighting** each sample using probability of evidence given parents
- Weights are *cumulative products* for each evidence variable

Example: Likelihood Weighting

- Suppose we want $P(C, R \mid +s, +w)$
- Fix $+s$ and $+w$
- We will require $P(+s|parents(S))$ and $P(+w|parents(W))$

C	P(+s|C)
+c	0.1
-c	0.5

R	P(+w|+s,R)
+r	0.99
-r	0.90



- (+c, +s, +r, +w) $0.1 \times .99 = .099$
- (+c, +s, +r, +w) $0.1 \times .99 = .099$
- (+c, +s, -r, +w) $0.1 \times 0.9 = 0.09$
- (-c, +s, -r, +w) $0.5 \times 0.9 = 0.45$

- Traverse nodes in order C, S, R, W
- Do not sample S and W; instead, update weight by multiplying prob of $+s/+w$
- When counting, **sum up the *weights* of each sample**, and then normalize

$\hat{P}(C, R, +s, +w)$

+c	+r	0.198
	-r	0.09
-c	+r	0
	-r	0.45

$\propto$

$\hat{P}(C, R \mid +s, +w)$

+c	+r	0.268
	-r	0.122
-c	+r	0
	-r	0.610

Sampling as Local Search

- Drawback of likelihood weighting: With lots of evidence, weights become small and tallies are dominated by a few samples with larger weights

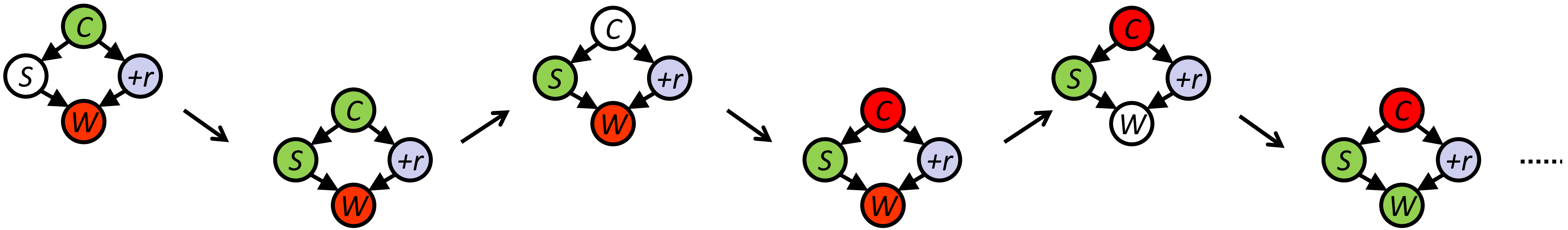
- Fixed evidence only affects sampling of variables that occur later
- Both “upstream” and “downstream” RVs should condition on evidence

- Can we also condition on evidence variables’ descendants?
- Idea: Instead of generating each new sample from scratch, make small change to current one (just like local search!)

Gibbs Sampling

- **Gibbs sampling**: Fix evidence and start with random sample (state) of non-evidence RVs $\boldsymbol{X}$. Generate next sample by sampling one X_i conditioned on all *current* $\boldsymbol{X}$. Repeat for different X_i in order.

- Example: Evidence $+r$. Start (randomly) with $(+c, -w, +r)$ and sample S.



Sample from $P(S \mid +c, +r, -w)$ and obtain $+s$

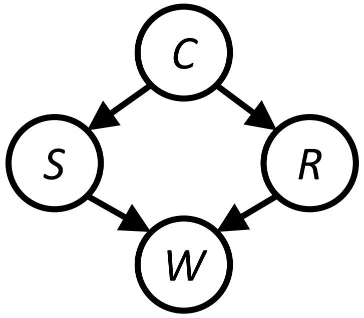
Sample from $P(C \mid +s, +r, -w)$ and obtain $-c$

Sample from $P(W \mid -c, +s, +r)$ and obtain $+w$

Gibbs Sampling

- Problem: How do we sample from $P(X_i \mid all\ other\ nodes\ in\ the\ BN)$?
- This is exactly equal to $P(X_i \mid MarkovBlanket(X_i))$!
- Easy to compute analytically; size is same as size of marginal $P(X_i)$

$$P(x_i' \mid mb(X_i)) = \alpha\, P(x_i' \mid parents(X_i)) \times \prod_{Y_j \in Children(X_i)} P(y_j \mid parents(Y_j))$$



$$P(C \mid s, r, w) = P(C \mid s, r) \propto P(C)P(s|C)P(r|C)$$

$$P(S \mid c, r, w) \propto P(c)P(S|c)P(r|c)P(w|S,r) \propto P(S|c)P(w|S,r)$$

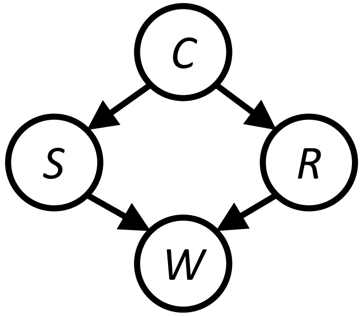
$$P(R \mid c, s, w) \propto P(c)P(s|c)P(R|c)P(w|s,R) \propto P(R|c)P(w|s,R)$$

$$P(W \mid c, s, r) = P(W \mid s, r)$$

Example: Gibbs Sampling

C	P(+s|C)
+c	0.1
-c	0.5

C	P(C)
+c	0.5



C	P(+r|C)
+c	0.8
-c	0.2

S	R	P(+w|S,R)
+s	+r	0.99
+s	-r	0.90
-s	+r	0.90
-s	-r	0

$$P(S \mid +c, +r, -w) \propto P(S|+c)P(-w|S, +r)$$

S	P(S,+c+r,-w)
+s	0.001
-s	0.09

=

S	P(S|+c)
+s	0.1
-s	0.9

×

S	P(-w|S,+r)
+s	0.01
-s	0.1

$$P(C \mid +s, +r, -w) \propto P(C)P(+s|C)P(+r|C)$$

C	P(C,+s,+r)
+c	0.04
-c	0.05

=

C	P(C)
+c	0.5
-c	0.5

×

C	P(+s|C)
+c	0.1
-c	0.5

×

C	P(+r|C)
+c	0.8
-c	0.2

$$P(W \mid -c, +s, +r) = P(W \mid +s, +r)$$

W	P(W|+s,+r)
+w	0.99
-w	0.01

Markov Chain Monte Carlo

- Gibbs sampling is a **Markov chain Monte Carlo (MCMC)** method

- We can think of Gibbs sampling as a Markov chain in the space of RVs
- Next sample depends only on current one
- Transition probability: likelihood of the next sample

- The joint distribution of the BN, conditioned on the evidence, is the *stationary distribution* of this Markov chain!

Summary

- Exact inference involves alternating between generating joint probabilities and then marginalizing them
- Can be improved using efficient ordering but still NP-hard

- Inference can be approximated via Monte Carlo methods
  - Prior sampling
  - Rejection sampling
  - Likelihood weighting
  - Gibbs sampling