Intelligent Systems (AI-2)

Computer Science cpsc422, Lecture 11

Oct, 3, 2016

422 big picture: Where are we?

StarAI (statistical relational AI)

Hybrid: Det +Sto

Prob CFG

Prob Relational Models

Markov Logics

Deterministic

Logics

First Order Logics

Ontologies

Temporal rep.

- Full Resolution
- SAT

Stochastic

Belief Nets

Approx. : Gibbs

Markov Chains and HMMs

Forward, Viterbi….

Approx. : Particle Filtering

Undirected Graphical Models

Markov Networks

Conditional Random Fields

Markov Decision Processes and Partially Observable MDP

- Value Iteration
- Approx. Inference

Reinforcement Learning

Query

Planning

Applications of AI

Representation

Reasoning Technique

Lecture Overview

- **Recap of BNs** Representation and Exact Inference
- Start Belief Networks **Approx. Reasoning**
  - Intro to **Sampling**
  - First Naïve Approx. Method: **Forward Sampling**
  - Second Method: **Rejection Sampling**

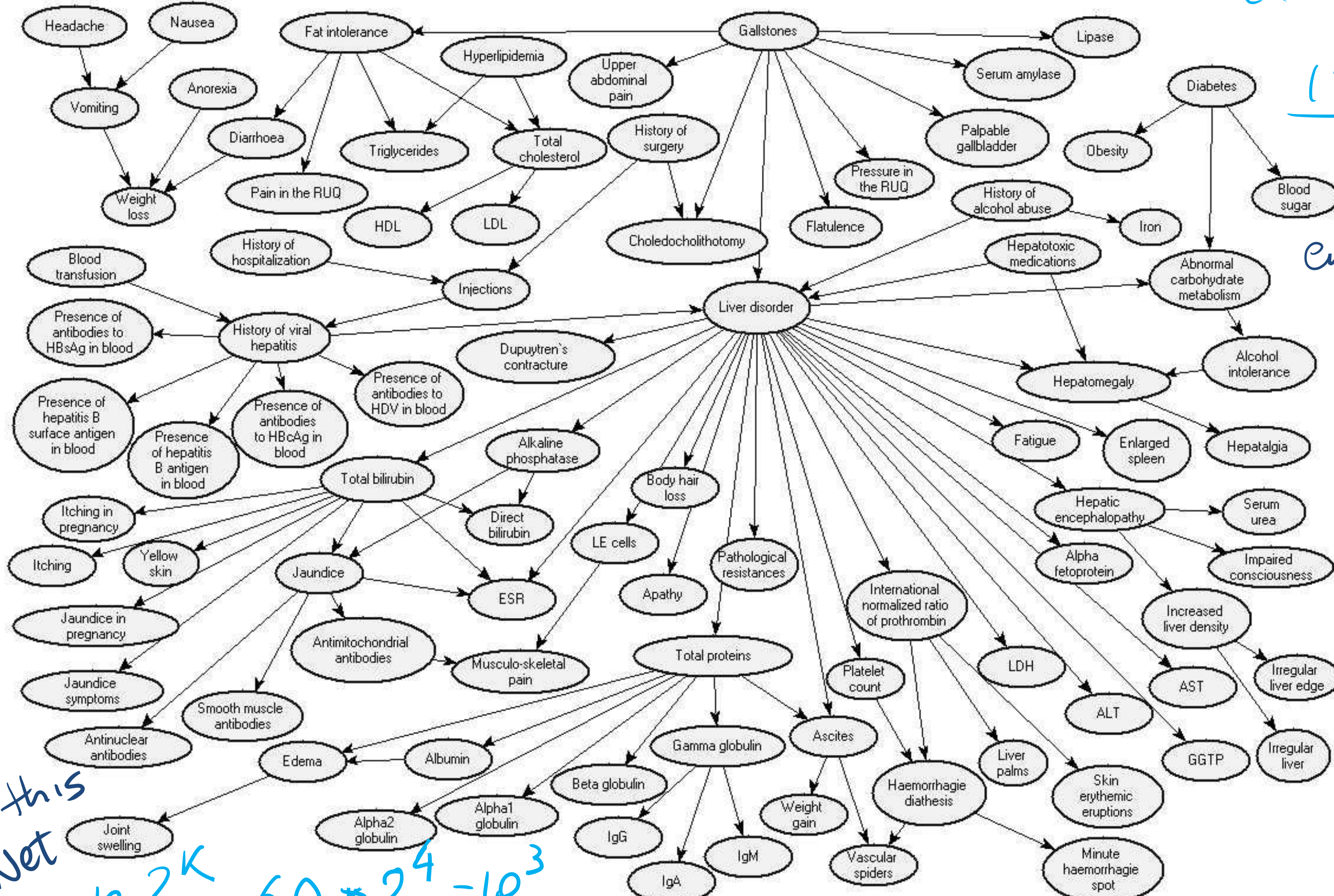
Realistic BNet: Liver Diagnosis

JPD

Source: Onisko et al., 1999

~ 60 nodes

$n = \sim 60$ $\sim 2^{60}$ $= (2^{10})^6$ $\approx 10^{18}$ entries

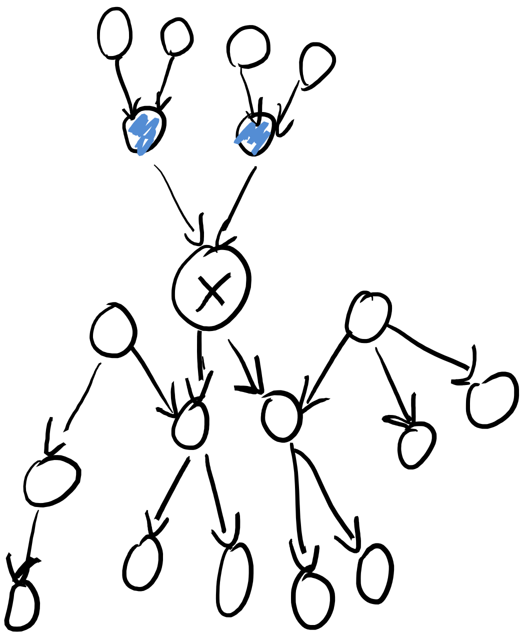


for this BNet

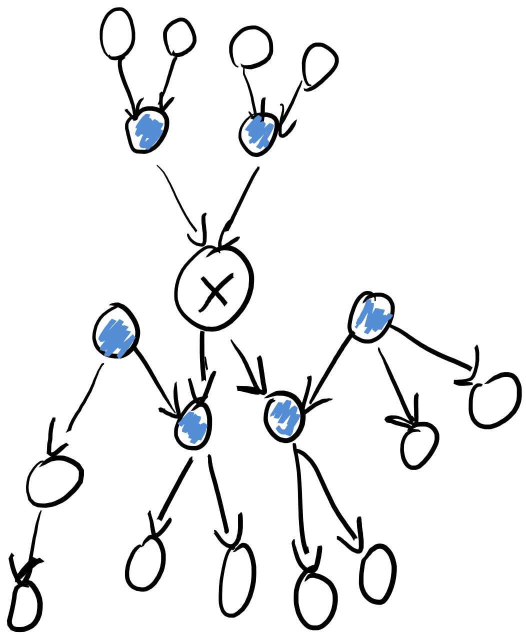
$\sim 2^K$ $60 * 2^4 = 10^3$

Revise (in)dependencies……

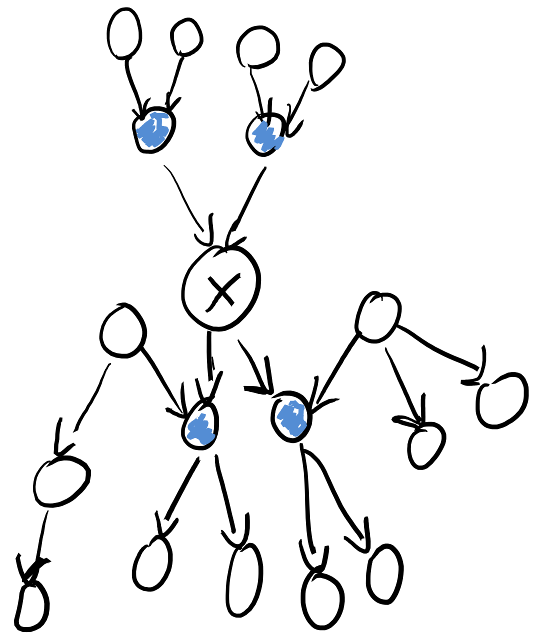
Independence (Markov Blanket)



A.



B.

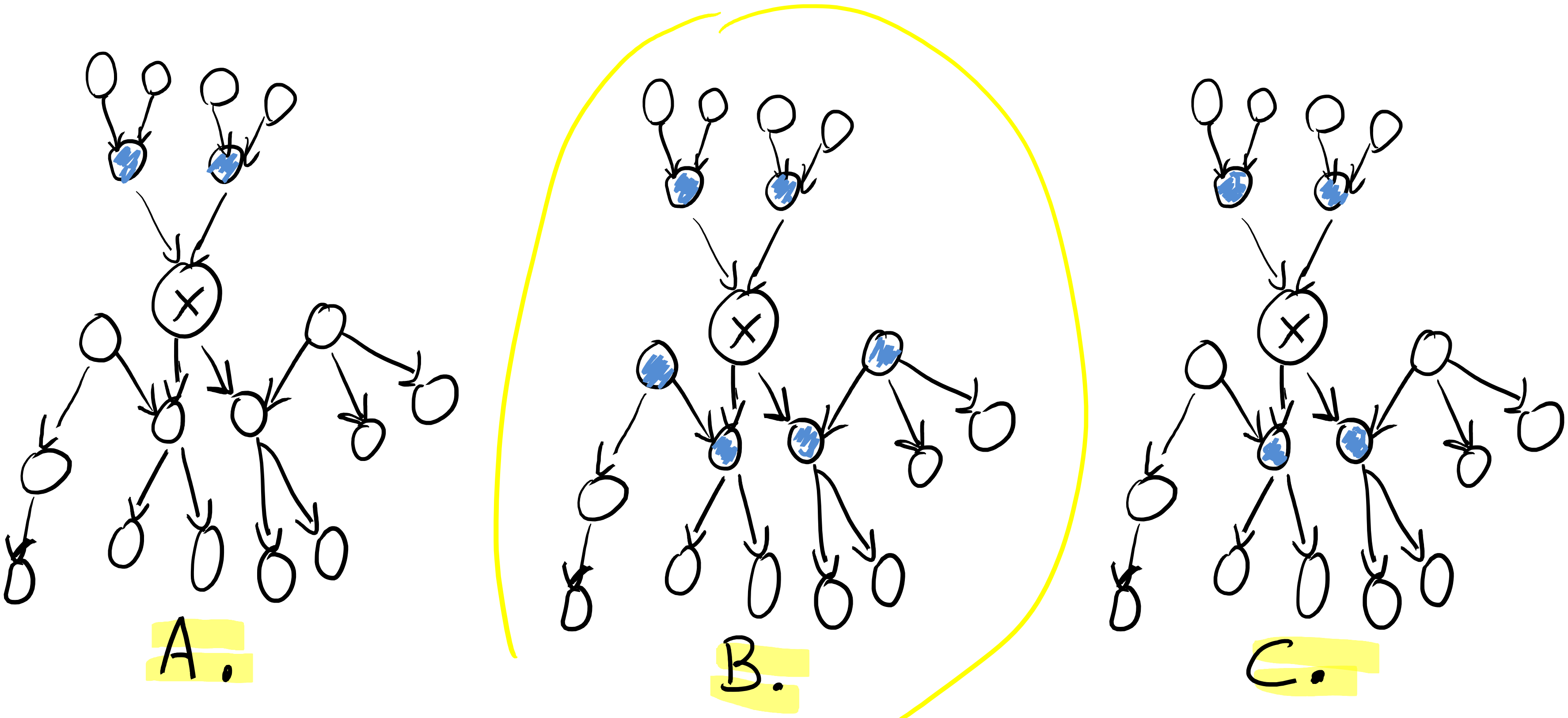


C.

What is the minimal set of nodes that must be observed in order to make **node X** independent from all the non-observed nodes in the network

iclicker.

Independence (Markov Blanket)



A node is conditionally independent from all the other nodes in the network, given its parents, children, and children's parents (i.e., its **Markov Blanket**) Configuration B

Variable elimination algorithm: Summary

$$P(Z,\ \cancel{Y_1,\cdots,Y_j}\ \ \cancel{Z_1,\cdots,Z_j}\ )$$

To compute $P(Z|\ Y_1=v_1\ ,\cdots\ ,Y_j=v_j\ )$**:**

1. Construct a factor for each conditional probability.
2. Set the observed variables to their observed values.
3. Given an elimination ordering, simplify/decompose sum of products
   - For all Z_i : Perform products and sum out Z_i
4. Multiply the remaining factors (all in ? Z)
5. Normalize: divide the resulting factor $f(Z)$ by $\sum_Z f(Z)$.

Variable elimination ordering

$$P(G,D{=}t) = \sum_{A,B,C,} f(A,G)\, f(B,A)\, f(C,G,A)\, f(B,C)$$

CBA

$$\sum_A f(A,G) \sum_B f(B,A) \sum_C f(C,G,A)\, f(B,C)$$

BCA

$$\sum_A f(A,G) \sum_C f(C,G,A) \sum_B f(B,C)\, f(B,A)$$

Complexity: Just Intuition…..

- **Tree-width of a network given an elimination ordering:** max number of variables in a factor created while running VE.
- **Tree-width of a belief network :** min tree-width over all elimination orderings (only on the graph structure and is a measure of the sparseness of the graph)

- **The complexity of VE is exponential in the tree-width** ☹ and linear in the number of variables.
- Also, finding the elimination ordering with minimum tree-width is NP-hard ☹ (but there are some good elimination ordering heuristics)

Lecture Overview

- **Recap of BNs** Representation and Exact Inference
- Start Belief Networks **Approx. Reasoning**
  - Intro to **Sampling**
  - First Naïve Approx. Method: **Forward Sampling**
  - Second Method: **Rejection Sampling**

Approximate Inference

Basic idea:

- Draw N samples from known prob. distributions
- Use those samples to estimate unknown prob. distributions

Why sample?

- Inference: getting a sample is faster than computing the right answer (e.g. with variable elimination)

We use *Sampling*

Sampling is a process to **obtain samples** adequate to estimate an **unknown probability**

How do we get samples?

Known prob. distribution(s)

Samples

Estimates for unknown (hard to compute) distribution(s)

Generating Samples from a Known Distribution

For a random variable X with

- values $\{x_1, \cdots, x_k\}$
- Probability distribution $P(X) = \{P(x_1), \cdots, P(x_k)\}$

Partition the interval [0, 1] into k intervals p_i , one for each x_i , with length $P(x_i)$

To generate one sample

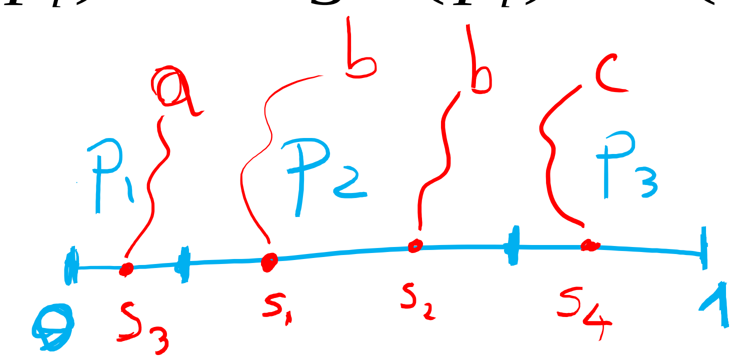
✓ Randomly generate a value y in [0, 1] (i.e. generate a value from a uniform distribution over [0, 1].

✓ Select the value of the sample based on the interval p_i that includes y

From probability theory: $$P(y \subset p_i) = Length(p_i) = P(x_i)$$

$X \{a, b, c\}$

x	$P(x)$
a	.1
b	.6
c	.3



From Samples to Probabilities

X	*count*
x_1	n_1
$\vdots$	$\vdots$
x_k	n_k
total	m

$\leftrightarrow$

X	*probability*
x_1	n_1/m
$\vdots$	$\vdots$
x_k	n_k/m

X	Count
00	4342
10	258
01	301
11	2299
total	7200

e.g. $P(01) = \frac{301}{7200}$

Count total number of samples m

Count the number n_i of samples x_i

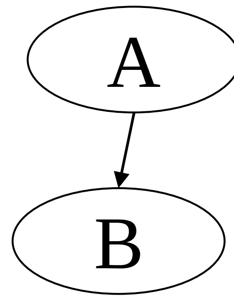
Generate the frequency of sample x_i as n_i / m

This frequency is your estimated probability of x_i

Sampling for Bayesian Networks (N)

- Suppose we have the following BN with two binary variables

A	P(B=1|A)	
1	0.7	.3
0	0.1	.9



P(A=1)	P(A=0)
0.3	0.7

Known distributions

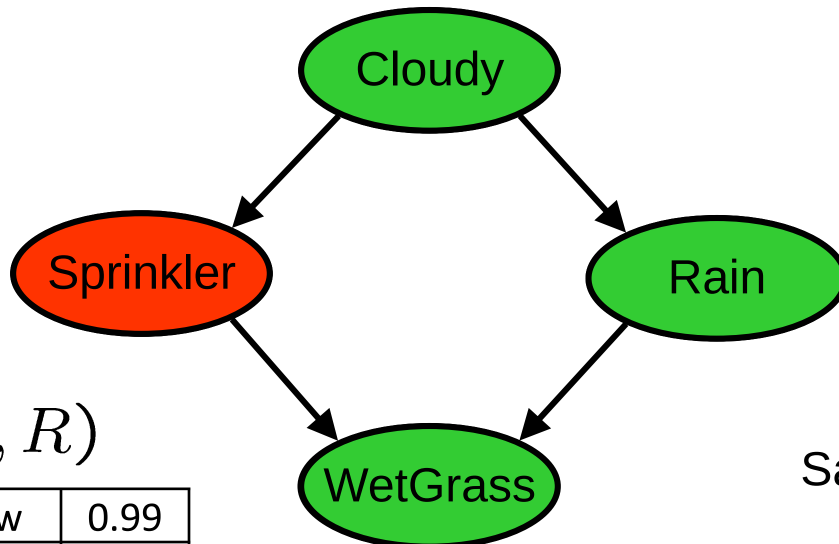
- It corresponds to the joint probability distribution
  - P(A,B) =P(B|A)P(A)
- To sample from P(A,B) i.e., unknown distribution
  - we first sample from P(A). Suppose we get A = 0.
  - In this case, we then sample from…. P(B|A=0)
  - If we had sampled A = 1, then in the second step we would have sampled from P(B|A=1)

A=0 B=1
A=0 B=0
A=1 B=1

Prior (Forward) Sampling

$P(C)$

+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

Samples:

+c, -s, +r, +w

-c, +s, -r, +w

…

Example

We'll get a bunch of samples from the BN:

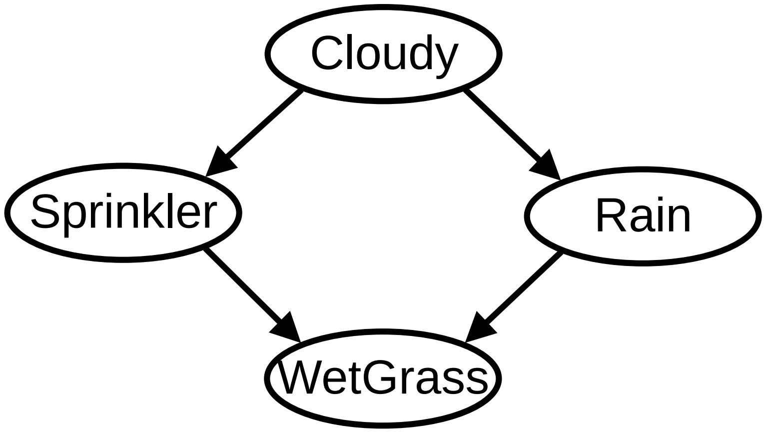
+c, −s, +r, +w

+c, +s, +r, +w

−c, +s, +r, −w

+c, −s, +r, +w

−c, −s, −r, +w



If we want to know P(W)

- We have counts <+w:4, −w:1>
- Normalize to get P(W) = <+w: .8 , −w: .2>
- This will get closer to the true distribution with more samples

Example

Can estimate anything else from the samples, besides P(W), P(R) , etc:

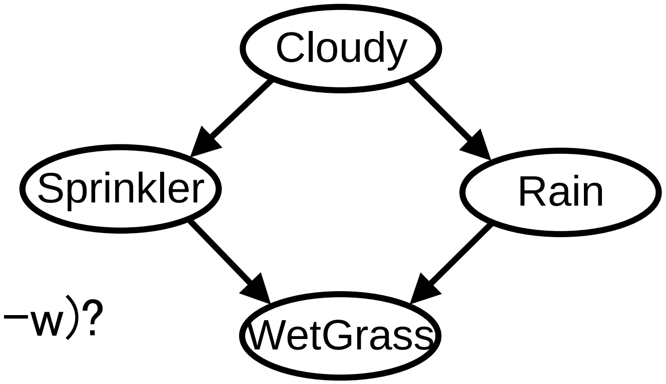
+c, −s, +r, +w

+c, +s, +r, +w

−c, +s, +r, −w

+c, −s, +r, +w

−c, −s, −r, +w



- What about P(C| +w)? P(C| +r, +w)? P(C| −r, −w)?

A. $\begin{array}{cc} +c & -c \\ \hline \end{array}$ $\begin{bmatrix} 0 & 1 \end{bmatrix}$

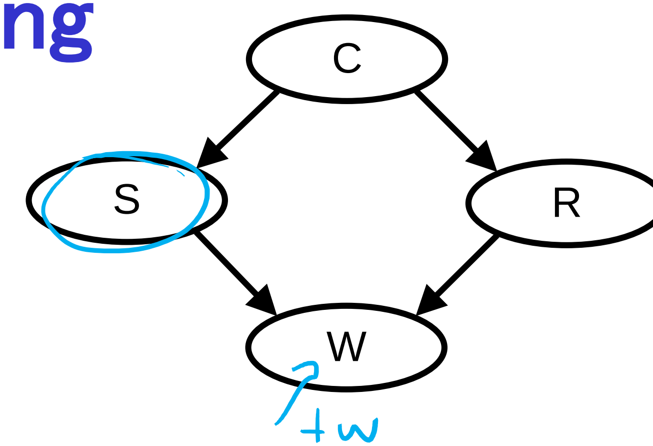
B. $\begin{array}{cc} +c & -c \\ \hline \end{array}$ $\begin{bmatrix} .5 & .5 \end{bmatrix}$

C. $\begin{array}{cc} +c & -c \\ \hline \end{array}$ $\begin{bmatrix} 1 & 0 \end{bmatrix}$

D. None of the above

Can use/generate fewer samples when we want to estimate a probability conditioned on evidence?

Rejection Sampling



Let's say we want P(S| +w)

- Ignore (reject) samples which don't have W=+w
- This is called rejection sampling
- It is also consistent for conditional probabilities (i.e., correct in the limit)

See any problem as the number of evidence vars increases?

+c, -s, ~~+r, +w~~
+c, +s, ~~+r, +w~~
-c, +s, ~~+r, -w~~
+c, -s, ~~+r, +w~~
-c, -s, ~~-r, +w~~

Hoeffding's inequality

- Suppose p is the true probability and s is the sample average from n independent samples.

$$P(|s-p|>\varepsilon) \le 2e^{-2n\varepsilon^2} < \delta$$

- p above can be the probability of any event for random variable $X = \{X_1, \cdots X_n\}$ described by a Bayesian network

- If you want an infinitely small probability of having an error greater than ε, you need infinitely many samples
- But if you settle on something less than infinitely small, let's say δ, then you just need to set

$$2e^{-2n\varepsilon^2} < \delta$$

- So you pick
  - the error ε you can tolerate,
  - the frequency δ with which you can tolerate it
- And solve for n, i.e., the number of samples that can ensure this performance

$$n > \frac{-\ln\frac{\delta}{2}}{2\varepsilon^2} \qquad (1)$$

Hoeffding's inequality

- Examples:
  - You can tolerate an error greater than 0.1 only in 5% of your cases
  - Set ε =0.1, δ = 0.05
  - Equation (1) gives you n > 184

$$n > \frac{-\ln\frac{\delta}{2}}{2\varepsilon^2} \qquad (1)$$

can rewrite as

$$n > \frac{\ln\frac{2}{\delta}}{2\varepsilon^2}$$

- If you can tolerate the same error (0.1) only in 1% of the cases, then you need 265 samples

- If you want an error greater than 0.01 in no more than 5% of the cases, you need 18,445 samples

so it should be clear that

$\varepsilon \downarrow$

$\delta \downarrow$

$n \uparrow$

$\downarrow$ goes down

$\uparrow$ goes up

Learning Goals for today's class

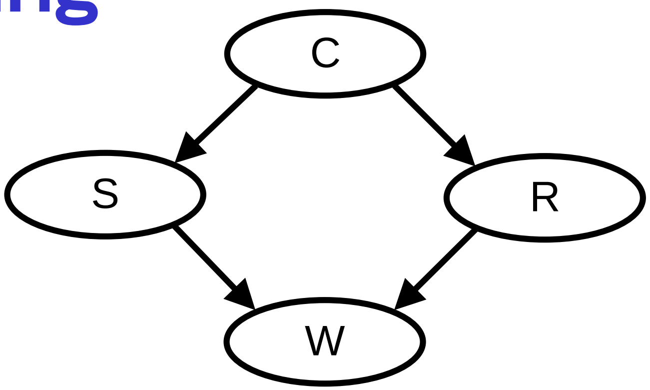
➢ **You can:**

- Motivate the need for approx inference in Bnets
- Describe and compare Sampling from a single random variable
- Describe and Apply Forward Sampling in BN
- Describe and Apply Rejection Sampling
- Apply Hoeffding's inequality to compute number of samples needed

TODO for Wed

- Read textbook 6.4.2
- Assignment-2 will be out today: Start working on it
- Next research paper will be this coming Fri

Rejection Sampling



Let's say we want P(C)

- No point keeping all samples around
- Just tally counts of C as we go

Let's say we want P(C| +s)

- Same thing: tally C outcomes, but ignore (reject) samples which don't have S=+s
- This is called rejection sampling
- It is also consistent for conditional probabilities (i.e., correct in the limit)

+c, -s, +r, +w
+c, +s, +r, +w
-c, +s, +r, -w
+c, -s, +r, +w
-c, -s, -r, +w