Bayes Nets III: Inference

Hal Daumé III

Computer Science
University of Maryland

me@hal3.name

CS 421: Introduction to Artificial Intelligence

10 Apr 2012

Many slides courtesy of
Dan Klein, Stuart Russell,
or Andrew Moore

Announcements

- Midterms graded
  - Grades posted, pick up after class, complain soon :)
  - Grade distribution:
- Projects:
  - P3 solution is posted
  - P4 (a combined P4/P5 is posted)
    - If you do well on P4, it's worth 14%
    - otherwise, it's worth 8.75%

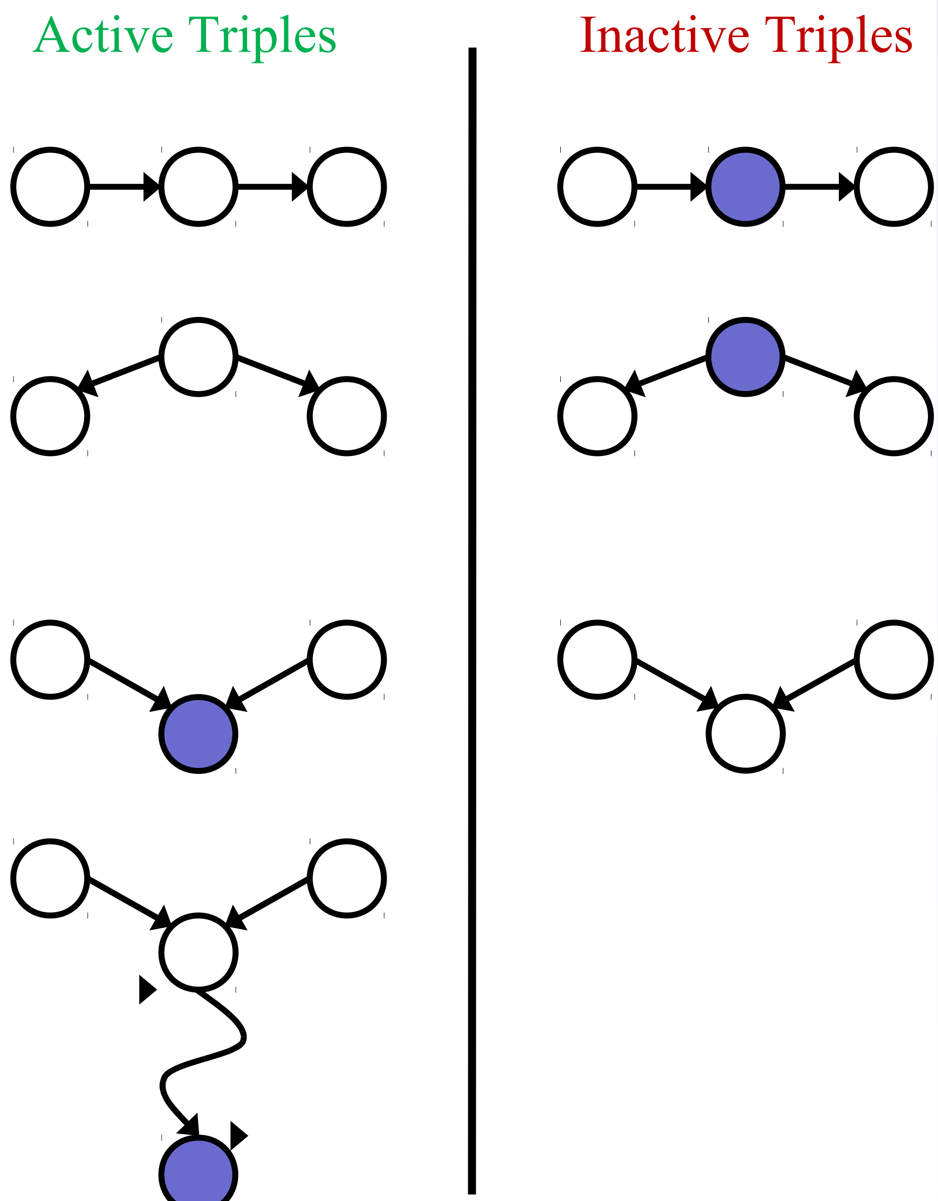
Contest

- Capture-the-flag style pacman
  - Tight connection to P4
  - Completely optional, team based (<=3 students)
- Deadline: 8 May
- Prizes:
  - Worth a few points on the final exam
  - See web page for prize details

Reachability (D-Separation)

- Question: Are X and Y conditionally independent given evidence variables {Z}?
  - Look for “active paths” from X to Y
  - No active paths = independence!
- A path is active if each triple is either a:
  - Causal chain $A \rightarrow B \rightarrow C$ where B is unobserved (either direction)
  - Common cause $A \leftarrow B \rightarrow C$ where B is unobserved
  - Common effect (aka v-structure)

    $A \rightarrow B \leftarrow C$ where B or one of its descendents is observed



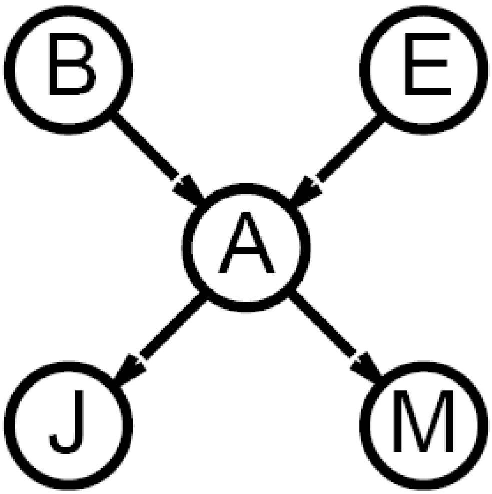
Causality?

- When Bayes' nets reflect the true causal patterns:
  - Often simpler (nodes have fewer parents)
  - Often easier to think about
  - Often easier to elicit from experts
- BNs need not actually be causal
  - Sometimes no causal net exists over the domain
  - E.g. consider the variables *Traffic* and *Drips*
  - End up with arrows that reflect correlation, not causation
- What do the arrows really mean?
  - Topology may happen to encode causal structure
  - Topology only guaranteed to encode conditional independencies

Inference by Enumeration

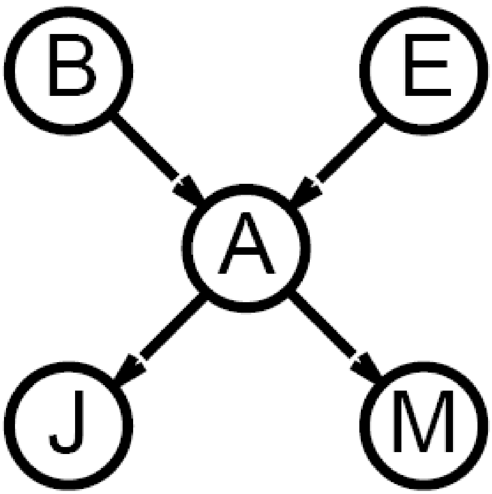
- Given unlimited time, inference in BNs is easy
- Recipe:
  - State the marginal probabilities you need
  - Figure out ALL the atomic probabilities you need
  - Calculate and combine them
- Example:

$$P(b|j,m) = \frac{P(b,j,m)}{P(j,m)}$$



Example

$$P(b|j,m) = \frac{P(b,j,m)}{P(j,m)}$$



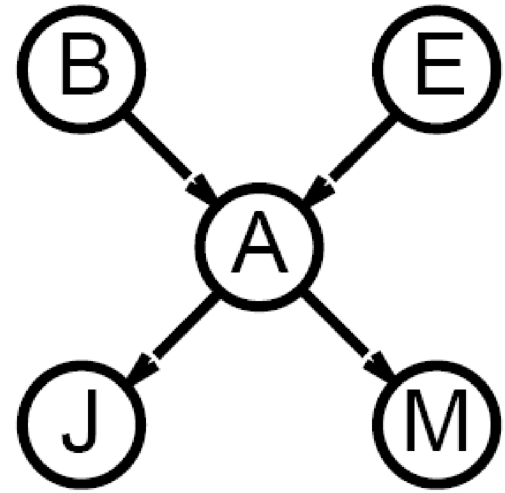
$$\begin{aligned} P(b,j,m) = \; & P(b,e,a,j,m)+ \\ & P(b,\bar{e},a,j,m)+ \\ & P(b,e,\bar{a},j,m)+ \\ & P(b,\bar{e},\bar{a},j,m) \end{aligned}$$

Where did we use the BN structure?

$$= \sum_{e,a} P(b,e,a,j,m)$$

We didn't!

Example

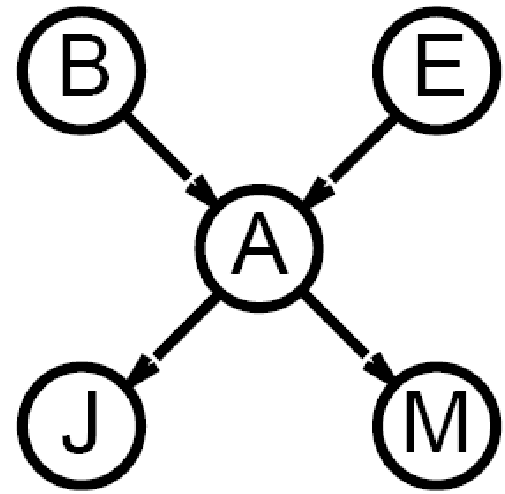


- In this simple method, we only need BN to synthesize the joint entries

$$
\begin{aligned}
P(b, j, m) = {} \\
& P(b)P(e)P(a|b,e)P(j|a)P(m|a) + \\
& P(b)P(e)P(\bar{a}|b,e)P(j|\bar{a})P(m|\bar{a}) + \\
& P(b)P(\bar{e})P(a|b,\bar{e})P(j|a)P(m|a) + \\
& P(b)P(\bar{e})P(\bar{a}|b,\bar{e})P(j|\bar{a})P(m|\bar{a})
\end{aligned}
$$

Normalization Trick

$$P(B|j,m) = \frac{P(B,j,m)}{P(j,m)}$$

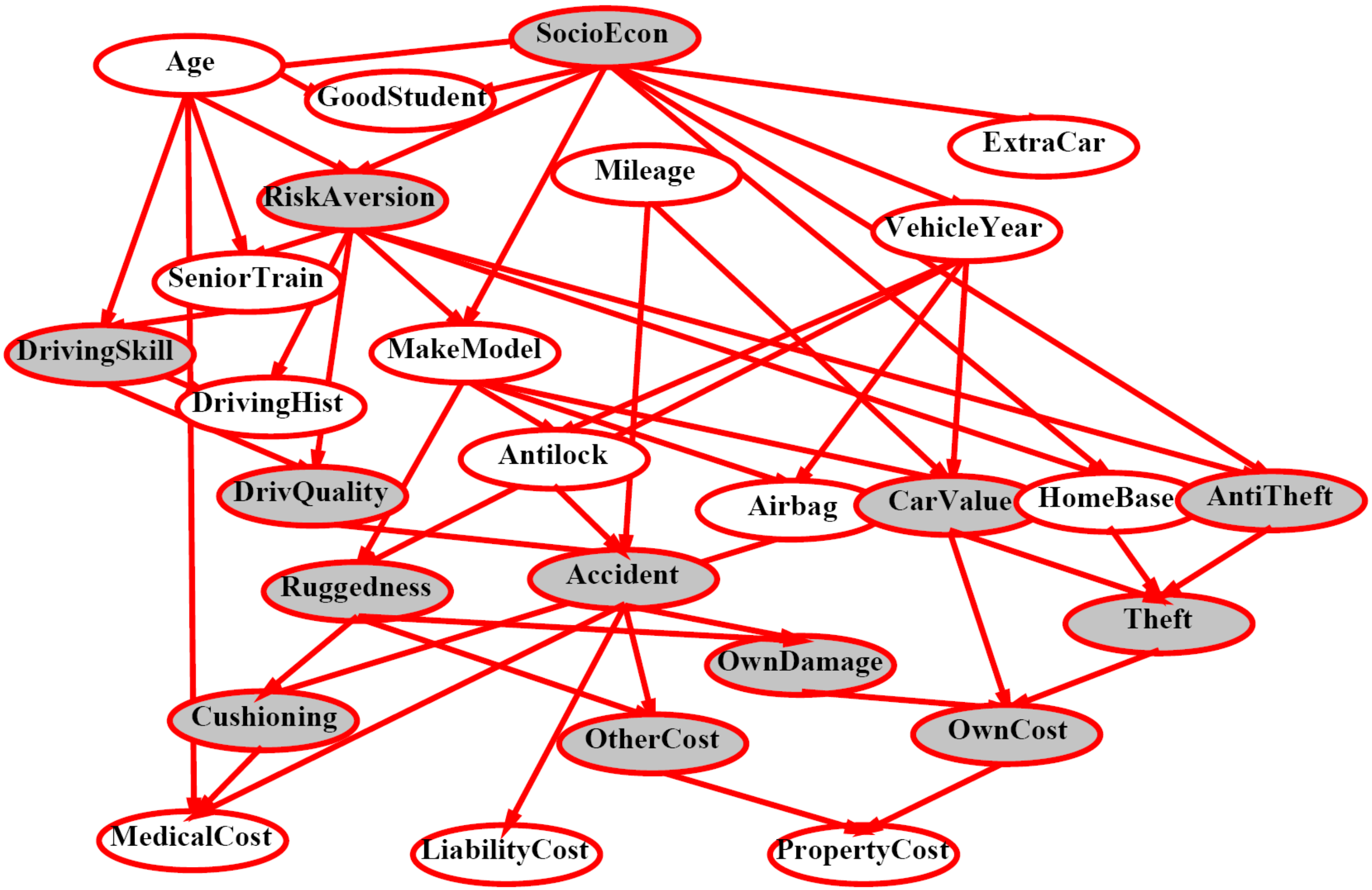


$$P(b,j,m) = \sum_{e,a} P(b,e,a,j,m)$$

$$P(\bar{b},j,m) = \sum_{e,a} P(\bar{b},e,a,j,m)$$

$$\begin{pmatrix} P(b,j,m) \\ P(\bar{b},j,m) \end{pmatrix} \xrightarrow{\text{Normalize}} \begin{pmatrix} P(b|j,m) \\ P(\bar{b}|j,m) \end{pmatrix}$$

Inference by Enumeration?



Variable Elimination

- Why is inference by enumeration so slow?
  - You join up the whole joint distribution before you sum out the hidden variables
  - You end up repeating a lot of work!
- Idea: interleave joining and marginalizing!
  - Called “Variable Elimination”
  - Still NP-hard, but usually much faster than inference by enumeration
- We’ll need some new notation to define VE

Factor Zoo I

Joint distribution: P(X,Y)

- Entries P(x,y) for all x, y
- Sums to 1

$P(T, W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

- Selected joint: P(x,Y)
  - A slice of the joint distribution
  - Entries P(x,y) for fixed x, all y
  - Sums to P(x)

$P(cold, W)$

T	W	P
cold	sun	0.2
cold	rain	0.3

Factor Zoo II

- Family of conditionals: P(X |Y)
  - Multiple conditionals
  - Entries P(x | y) for all x, y
  - Sums to |Y|

$$P(W|T)$$

T	W	P
hot	sun	0.8
hot	rain	0.2
cold	sun	0.4
cold	rain	0.6

$P(W|hot)$

$P(W|cold)$

- Single conditional: P(Y | x)
  - Entries P(y | x) for fixed x, all y
  - Sums to 1

$$P(W|cold)$$

T	W	P
cold	sun	0.4
cold	rain	0.6

Factor Zoo III

- Specified family: P(y | X)
  - Entries P(y | x) for fixed y, all x
  - Sums to … who knows!

$P(rain|T)$

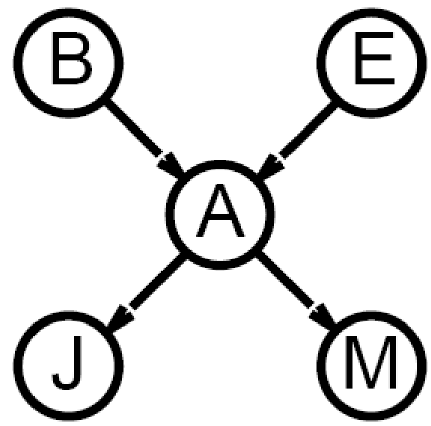
T	W	P
hot	rain	0.2
cold	rain	0.6

$P(rain|hot)$

$P(rain|cold)$

- In general, when we write $P(Y_1 \ldots Y_N \mid X_1 \ldots X_M)$
  - It is a "factor," a multi-dimensional array
  - Its values are all $P(y_1 \ldots y_N \mid x_1 \ldots x_M)$
  - Any unassigned X or Y is a dimension missing (selected) from the array

Basic Objects



- Track objects called factors
- Initial factors are local CPTs
  - One per node in the BN

$$P(B) \quad P(E) \quad P(J|A) \quad P(M|A) \quad P(A|B,E)$$

- Any known values are specified
  - E.g. if we know J = j and E = ¬e, the initial factors are

$$P(B) \quad P(\neg e) \quad P(j|A) \quad P(M|A) \quad P(A|B,\neg e)$$

- VE: Alternately join and marginalize factors

Basic Operation: Join

- First basic operation: join factors
- Combining two factors:
  - Just like a database join
  - Build a factor over the union of the variables involved
- Example:

$$P(A|B) \quad \times \quad P(B|C) \quad \Longrightarrow \quad P(A,B|C)$$

  - Computation for each entry: pointwise products

$$\forall a,b,c: \qquad P(a,b|c) = P(a|b) \cdot P(b|c)$$

Basic Operation: Join

- In general, we join on a variable
  - Take all factors mentioning that variable
  - Join them all together
- Example:

$$P(B) \quad P(\neg e) \quad P(j|A) \quad P(M|A) \quad P(A|B,\neg e)$$

  - Join on A:
  - Pick up these:

$$P(j|A) \quad P(M|A) \quad P(A|B,\neg e)$$

  - Join to form: $P(j,M,A|B,\neg e)$

Basic Operation: Eliminate

- Second basic operation: marginalization
- Take a factor and sum out a variable
  - Shrinks a factor to a smaller one
  - A projection operation
- Example:

$$P(A, b|C) \xrightarrow{\text{sum } A} P(b|C)$$

- Definition:

$$\forall c: \quad P(b|c) = \sum_a P(a, b|c)$$

General Variable Elimination

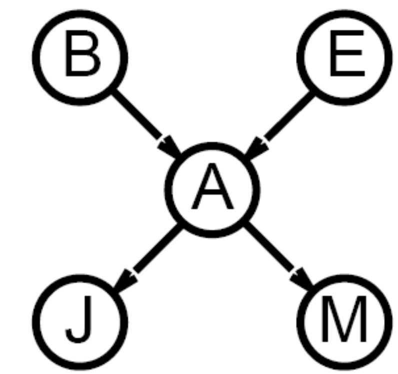
- Query: $P(Q|E_1 = e_1, \ldots E_k = e_k)$

- Start with initial factors:
  - Local CPTs (but instantiated by evidence)

- While there are still hidden variables (not Q or evidence):
  - Pick a hidden variable H
  - Join all factors mentioning H
  - Project out H

- Join all remaining factors and normalize

Example



$$P(B|j,m) \propto P(B,j,m)$$

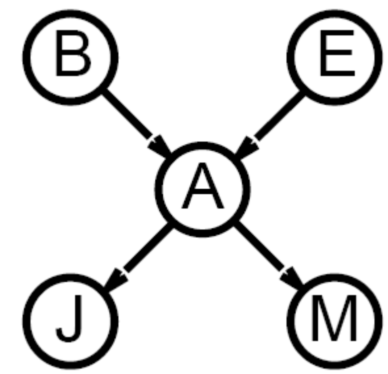
$$\boxed{P(B) \qquad P(E) \qquad P(A|B,E) \qquad P(j|A) \qquad P(m|A)}$$

Choose A

$$\left.\begin{array}{l} P(A|B,E) \\ P(j|A) \\ P(m|A) \end{array}\right. \xrightarrow{\times} P(j,m,A|B,E) \xrightarrow{\sum} P(j,m|B,E)$$

$$\boxed{P(B) \qquad P(E) \qquad P(j,m|B,E)}$$

Example



$$P(B) \qquad P(E) \qquad P(j,m|B,E)$$

Choose E

$$P(E) \quad P(j,m|B,E) \quad \xrightarrow{\times} \quad P(j,m,E|B) \quad \xrightarrow{\sum} \quad P(j,m|B)$$

$$P(B) \qquad P(j,m|B)$$

Finish with B

$$P(B) \quad P(j,m|B) \quad \xrightarrow{\times} \quad P(j,m,B) \quad \xrightarrow{\text{Normalize}} \quad P(B|j,m)$$

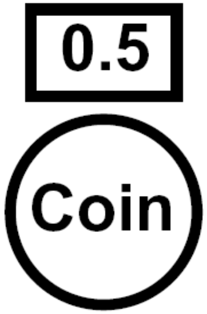
Variable Elimination

- What you need to know:
  - Should be able to run it on small examples, understand the factor creation / reduction flow
  - Better than enumeration: VE caches intermediate computations
  - Saves time by marginalizing variables as soon as possible rather than at the end
  - Polynomial time for tree-structured graphs – sound familiar?
- We will see special cases of VE later
  - You'll have to implement the special cases

- Approximations
  - Exact inference is slow, especially with a lot of hidden nodes
  - Approximate methods give you a (close, wrong?) answer, faster

Sampling

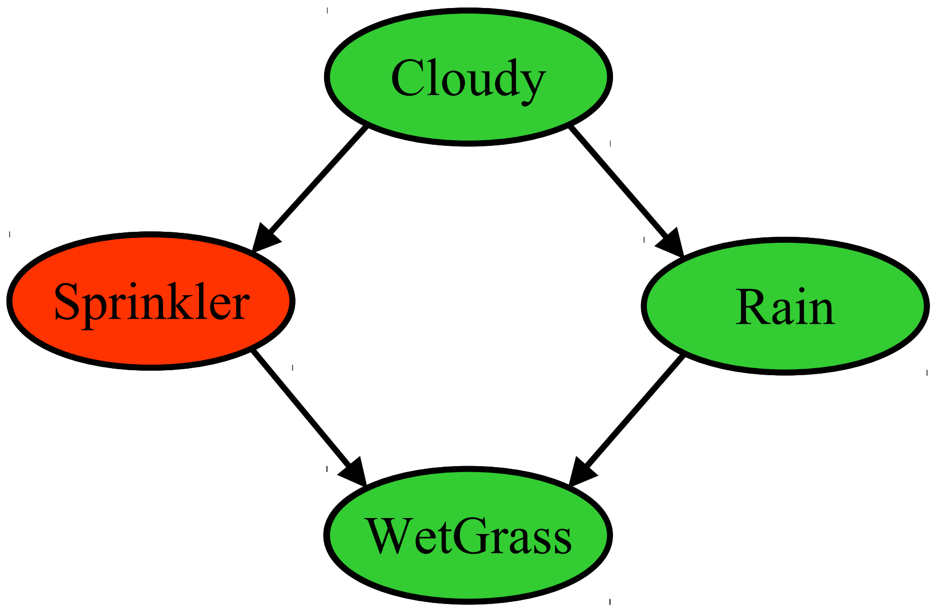
- Basic idea:
  - Draw N samples from a sampling distribution S
  - Compute an approximate posterior probability
  - Show this converges to the true probability P



- Outline:
  - Sampling from an empty network
  - Rejection sampling: reject samples disagreeing with evidence
  - Likelihood weighting: use evidence to weight samples

Prior Sampling

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

Prior Sampling

- This process generates samples with probability

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

…i.e. the BN's joint probability

- Let the number of samples of an event be $N_{PS}(x_1 \ldots x_n)$

- Then $\lim_{N\to\infty} \hat{P}(x_1, \ldots, x_n) = \lim_{N\to\infty} N_{PS}(x_1, \ldots, x_n)/N$

$$= S_{PS}(x_1, \ldots, x_n)$$

$$= P(x_1 \ldots x_n)$$

- I.e., the sampling procedure is consistent

Example

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

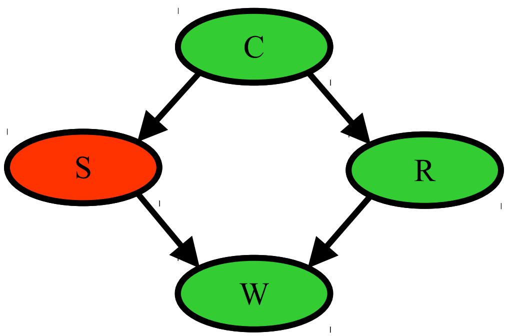
  c, ¬s, r, w

  c, s, r, w

  ¬c, s, r, ¬w

  c, ¬s, r, w

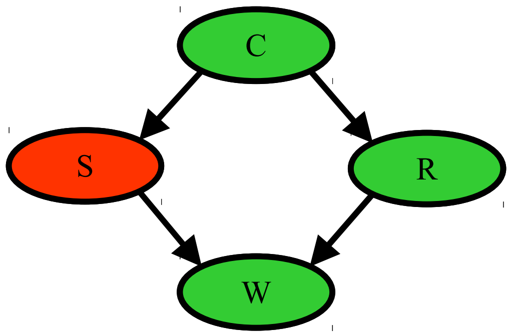
  ¬c, s, ¬r, w



- If we want to know P(W)
  - We have counts <w:4, ¬w:1>
  - Normalize to get P(W) = <w:0.8, ¬w:0.2>
  - This will get closer to the true distribution with more samples
  - Can estimate anything else, too
  - What about P(C| ¬r)? P(C| ¬r, ¬w)?

Rejection Sampling

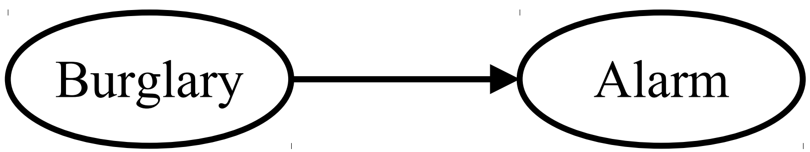
- Let's say we want P(C)
  - No point keeping all samples around
  - Just tally counts of C outcomes
- Let's say we want P(C| s)
  - Same thing: tally C outcomes, but ignore (reject) samples which don't have S=s
  - This is rejection sampling
  - It is also consistent (correct in the limit)



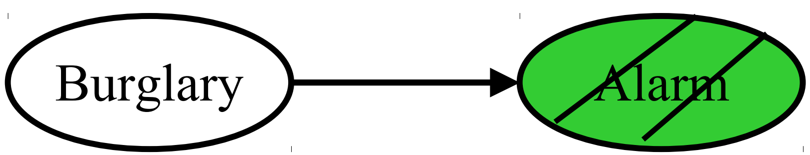
c, ¬s, r, w
c, s, r, w
¬c, s, r, ¬w
c, ¬s, r, w
¬c, s, ¬r, w

Likelihood Weighting

- Problem with rejection sampling:
  - If evidence is unlikely, you reject a lot of samples
  - You don't exploit your evidence as you sample
  - Consider P(B|a)



- Idea: fix evidence variables and sample the rest

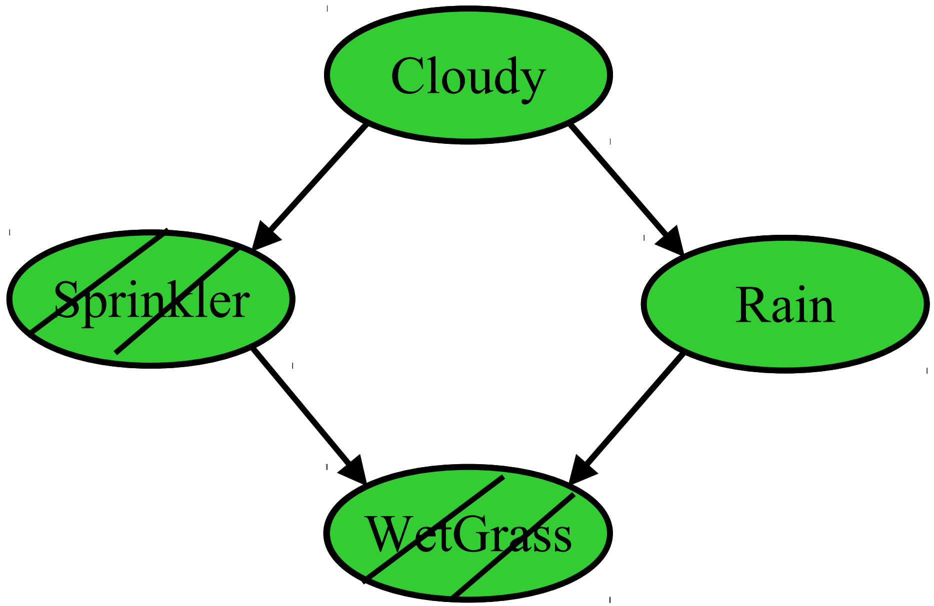


- Problem: sample distribution not consistent!
- Solution: weight by probability of evidence given parents

Likelihood Sampling

P(C)
.50

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



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

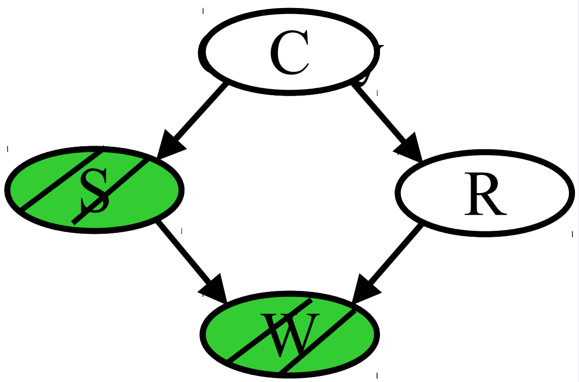
S	R	P(W|S,R)
T	T	.99
T	F	.90
F	T	.90
F	F	.01

w = 1.0 * 0.1 * 0.9

Likelihood Weighting

- Sampling distribution if z sampled and e fixed evidence

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



- Now, samples have weights

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

- Together, weighted sampling distribution is consistent

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

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

Likelihood Weighting

- Note that likelihood weighting doesn't solve all our problems
- Rare evidence is taken into account for downstream variables, but not upstream ones
- A better solution is Markov-chain Monte Carlo (MCMC), more advanced
- We'll return to sampling for robot localization and tracking in dynamic BNs