Bayesian networks

AIMA2e Chapter 14

Outline

- ♢ Syntax
- ♢ Semantics
- ♢ Parameterized distributions

Bayesian networks

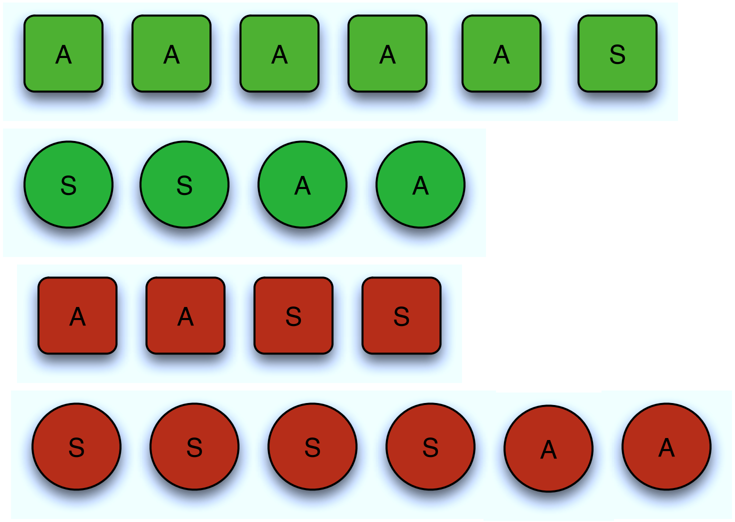
A simple, graphical notation for conditional independence assertions and hence for compact specification of full joint distributions

Syntax:

a set of nodes, one per variable
a directed, acyclic graph (link $\approx$ "directly influences")
a conditional distribution for each node given its parents:
$\mathbf{P}(X_i|Parents(X_i))$

In the simplest case, conditional distribution represented as a conditional probability table (CPT) giving the distribution over X_i for each combination of parent values

Independence

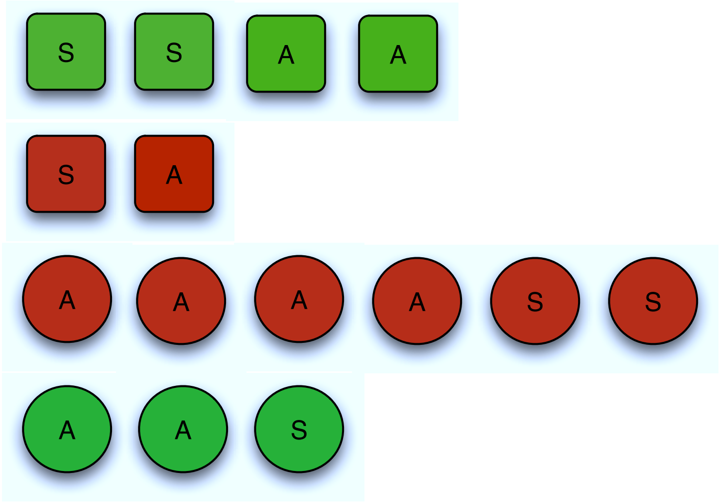


- 3 variables: **Color** (Red, Green), **Shape** (Circle, Square), **Mark** (A, S)
- Are shape and color independent?
  - $P(circle) = P(square) = P(red) = P(green) = 0.5$
  - But, $P(circle \mid red) = 0.6 \neq P(circle)$ and $P(circle \mid green) = 0.4 \neq P(circle)$
  - Similarly, $P(green \mid square) = 0.6 \neq P(green)$
  - Since background probs $\neq$ conditional probs, shape and color are not indep.

Independence (2)

- Are mark and color independent?
  - $P(s) = P(a) = 0.5 = P(s \mid red) = P(s \mid green) = P(a \mid red) = P(a \mid green)$
  - Yes, they are independent. Odds of getting a given mark are the same if we pick from the whole population or just from a particular color class.
- Are mark and shape independent?
  - $P(s) = P(a) = 0.5 = P(s \mid square) = P(s \mid circle) = P(a \mid circle) = P(a \mid square)$
  - Yes!
- Notice that independence is a symmetric property: If A is indep of B, then B is indep of A.
  - Proof: If A is indep of B, then, by definition, $P(A \mid B) = P(A)$.
  - We know that $P(A \wedge B) = P(A \mid B)P(B) = P(B \mid A)P(A)$
  - So $P(A)$ and $P(A \mid B)$ cancel in the rightmost equation.
  - Yielding $P(B) = P(B \mid A) \rightarrow$ B is indep of A.

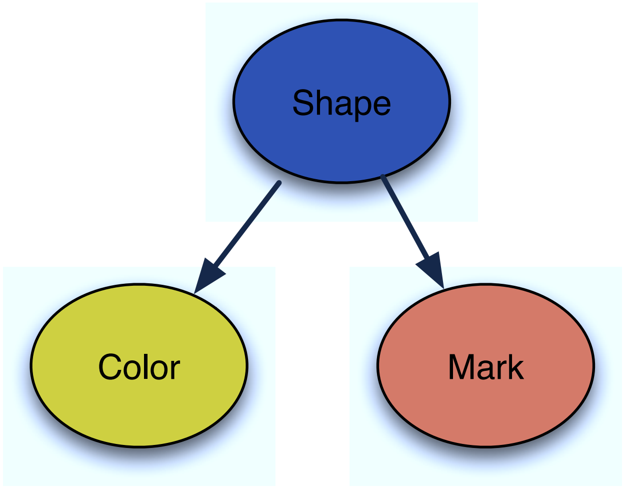
Conditional Independence

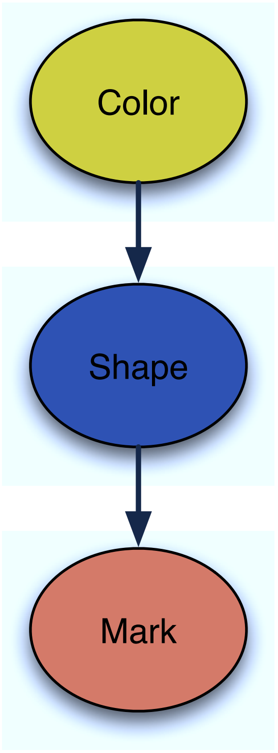


- Mark is not independent of color. E.g. $P(a) = 0.6 \neq P(a \mid red) = 0.625$
- And Mark is not indep of shape. $P(a \mid circle) = 0.666$.
- But Mark is conditionally independent of color, given shape:
  - $P(a \mid circle \wedge red) = P(a \mid circle \wedge green) = P(a \mid circle) = 0.666$
  - $P(a \mid square \wedge red) = P(a \mid square \wedge green) = P(a \mid square) = 0.5$

Conditional Independence and Causality

There may lie a causal relationship behind this:

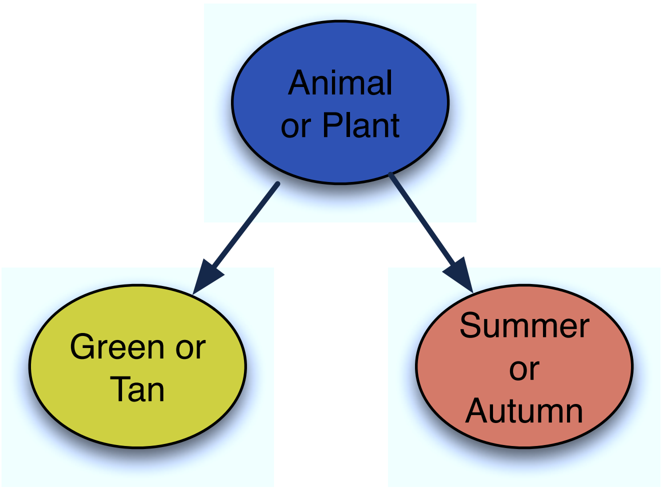




Conditional Independence and Causality (2)

But this only makes sense if the attributes have another meaning, such as:

- Shape: Square → Plant; Circle → Animal
- Color: Green → Brightly colored; Red → Tan or Dull colored
- Mark = Harvest/Slaughter time: S → Summer; A → Autumn

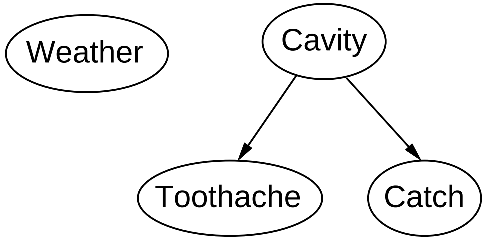


Conditional Independence and Causality (3)

1. Looking for independence and conditional independence in a large data set with many attributes and data vectors is a difficult task.
2. But we need to find them in order to reduce the number of necessary prior probabilities down to a reasonable size - i.e., linear (not exponential) in the number of attributes.
3. Via our **background knowledge** about the domain, we can see the raw data as more than just meaningless vectors of attribute values.
4. This will lead to good hypotheses about possible independences and causal independences.
5. These can be easily checked against the raw data.

Back to the Dentist's Office

Topology of network encodes conditional independence assertions:



- $Weather$ is independent of the other variables.
- $Toothache$ and $Catch$ are conditionally independent given Cavity.
- Once you know that there is (or is not) a cavity, then knowing whether or not there is a toothache does NOT give **additional** information as to whether or not the probe will Catch.
- $P(Catch \mid Toothache \wedge Cavity) = P(Catch \mid Cavity)$
- $P(Toothache \mid Catch \wedge Cavity) = P(Toothache \mid Cavity)$

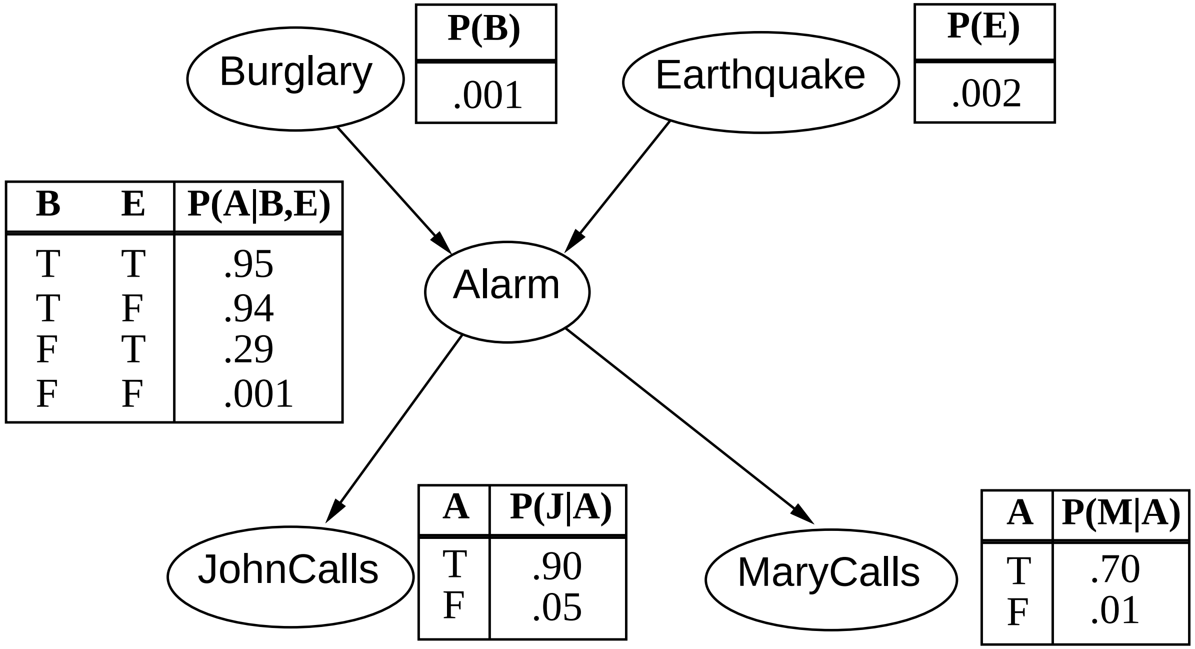
Classic Earthquake Example

I'm at work, neighbor John calls to say my alarm is ringing, but neighbor Mary doesn't call. Sometimes it's set off by minor earthquakes. Is there a burglar?

Variables: $Burglar$, $Earthquake$, $Alarm$, $JohnCalls$, $MaryCalls$
Network topology reflects "causal" knowledge:

- A burglar can set the alarm off
- An earthquake can set the alarm off
- The alarm can cause Mary to call
- The alarm can cause John to call

Bayesian Network for Earthquake Example



P(B)
.001

P(E)
.002

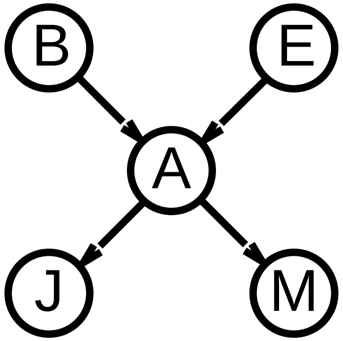
B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

Priors: $1(B) + 1(E) + 4(A) + 2\,(J) + 2(M) = 10$

Compactness



A CPT for Boolean X_i with k Boolean parents has 2^k rows for the combinations of parent values

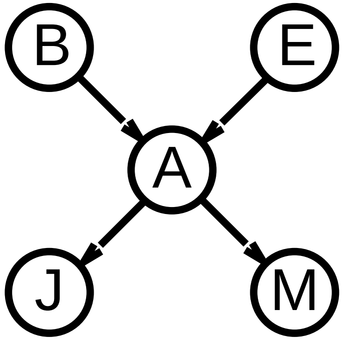
Each row requires one number p for $X_i = true$ (the number for $X_i = false$ is just $1 - p$)

If each variable has no more than k parents, the complete network requires $O(n \cdot 2^k)$ numbers

I.e., grows linearly with n, vs. $O(2^n)$ for the full joint distribution

For burglary net, $1+1+4+2+2=10$ numbers (vs. $2^5 - 1 = 31$)

Global semantics



Global semantics defines the full joint distribution as the product of the local conditional distributions:

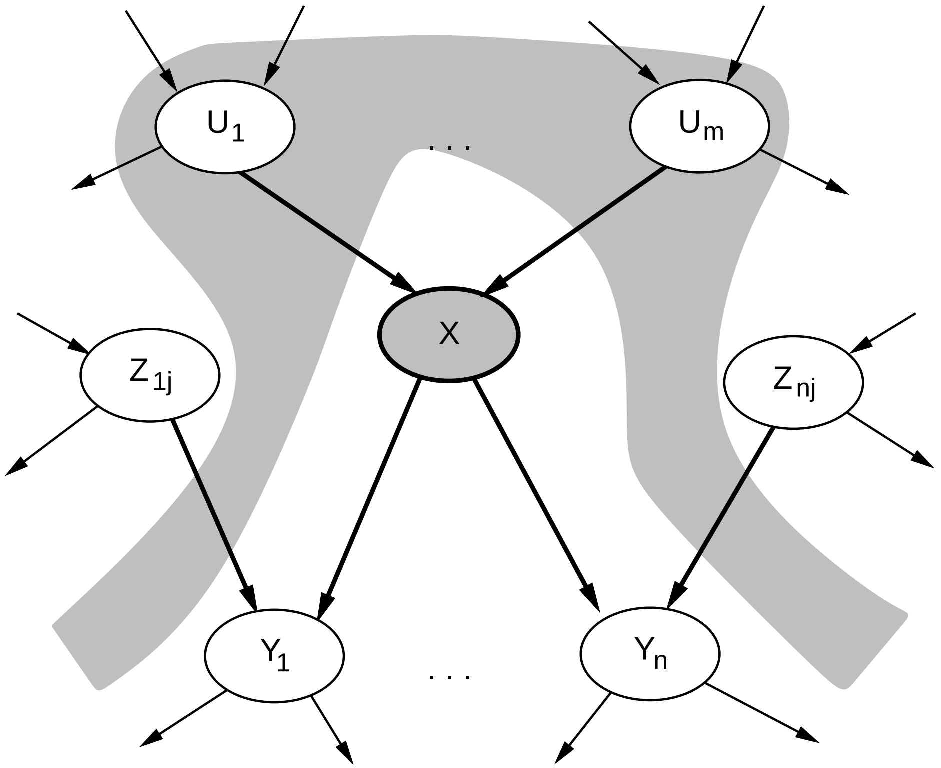
$$\mathbf{P}(X_1, \ldots, X_n) = \prod_{i=1}^{n} \mathbf{P}(X_i | Parents(X_i))$$

e.g., $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

$$= P(j|a)P(m|a)P(a|\neg b, \neg e)P(\neg b)P(\neg e)$$

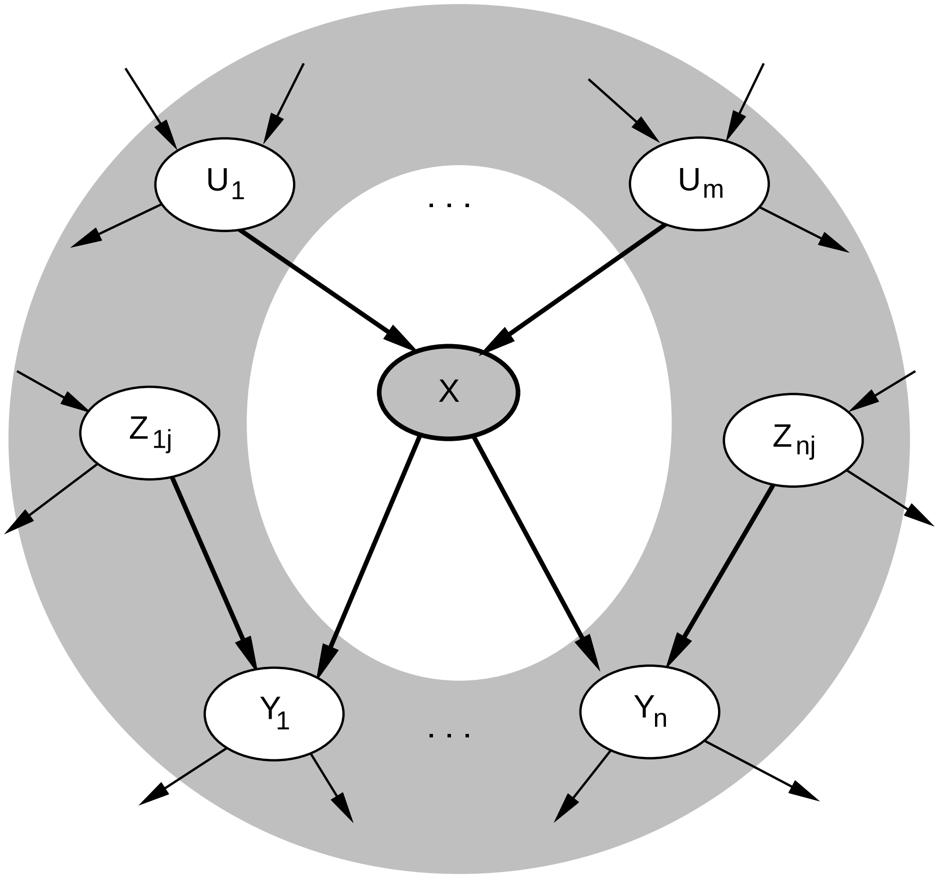
Local semantics

- Local semantics: each node is conditionally independent of its nondescendants, given its parents,
- So once you know parents' values, knowledge of earlier ancestors' values is of no extra help in determining the value of X.



Markov blanket

Each node is conditionally independent of all others given its Markov blanket: parents + children + children's parents



Markov blanket explained

- Parents - Make X cond indep of other ancestors.
- Children - Make X cond indep of other descendants.
- Children's Parents - Must be included, since, X is conditionally DEPENDENT upon Z_{nj}, given Y_n
  - $P(X \mid Y_n) \neq P(X \mid Y_n \wedge Z_{nj})$
  - E.g. If we know that the alarm went off, then knowing that there was a burglary has a STRONG influence on our belief that there was an earthquake (i.e. it drastically lowers P(Earthquake = T)).
  - Otherwise, in the absence of alarm information (it could be on or off), there is no relationship between Burglary and Earthquake.

Constructing Bayesian networks

Need a method such that a series of locally testable assertions of conditional independence guarantees the required global semantics

```
1. Choose an ordering of variables $X_1, \ldots, X_n$
2. For $i = 1$ to $n$
     add $X_i$ to the network
     select parents from $X_1, \ldots, X_{i-1}$ such that
          $\mathbf{P}(X_i|Parents(X_i)) = \mathbf{P}(X_i|X_1,\ \ldots,\ X_{i-1})$
```

- Hence, all parents of X_i come BEFORE it in the variable list.
- This makes probability calculations much easier.
- This also guarantees the global semantics:

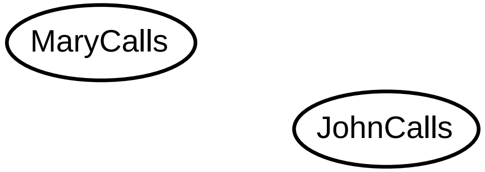
$$\begin{aligned}\mathbf{P}(X_1, \ldots, X_n) &= \prod_{i=1}^{n} \mathbf{P}(X_i|X_1,\ \ldots,\ X_{i-1}) \quad \text{(chain rule)} \\ &= \prod_{i=1}^{n} \mathbf{P}(X_i|Parents(X_i)) \quad \text{(by construction)}\end{aligned}$$

Reducing Priors

- Given: $X_1, \ldots, X_n$ binary variables, we have 2^n atomic events.
- If there is no independence among these variables, we need $2^n - 1$ priors.
- The Chain Rule alone does not reduce the number of priors.
- In fact, if each X_i has $X_1,\ \ldots,\ X_{i-1}$ as parents, then we need exactly: $2^{n-1} + 2^{n-2} + 2^{n-3} \ldots + 1 = 2^n - 1$ priors!
- However, in most situations, independence and conditional independence, will give many X_i with far less than i-1 parents.
- So the number of priors will reduce dramatically.

Net-making Decisions

Suppose we choose the ordering M,J,A,B,E.
We are free to choose any ordering among the variables, as long as all parents come before their children and we have conditional probability tables (filled in with prior probs) that connect each child to all of its parents.



- $P(J|M) = P(J)$?
  Is P(John calling) independent of P(Mary calling)?
- Clearly not, since, on any given day, if Mary called, then the probability that John called is much better than the background probability that he called.

Net-making Decisions (2)

- The alarm is the hidden cause that links J and M.
- Given knowledge about the alarm, J and M become independent: If the alarm goes off, then knowing that Mary called does not help determine whether John called. In terms of the atomic events, look at ONLY the days when the alarm goes off. On those days, you will see a weaker correlation between J and M than the correlation that you get when you look at ALL the days.
- But without alarm knowledge, the fact that Mary called is a very good predictor that John called.
- This is easier to understand by looking at a sample of atomic events.

Atomic Earthquake-Example Events

Alarm	Mary	John	Burglary	Earthquake
T	T	T	T	F
F	F	F	F	F
T	T	F	F	T
F	F	T	F	F
F	F	F	F	F
F	F	F	F	F
T	T	T	T	F
T	T	T	T	F
F	F	F	F	T
T	T	T	F	T

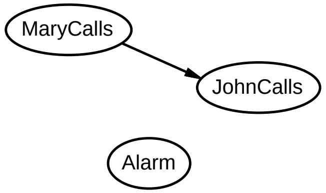
John, Mary, Alarm Probabilities

From the raw data:

- $P(JohnCalls = T) = .5$ and $P(MaryCalls = T) = .5$
- $P(JohnCalls = T \mid MaryCalls = T) = .8$
- Clearly, John calling is NOT independent of Mary calling.
- $P(JohnCalls \mid Alarm = T) = .8$
- $P(JohnCalls \mid Alarm = T \wedge MaryCalls = T) = .8$
- So adding the condition that Mary called does not affect the probability of John having called, given that the alarm went off.
- John calling and Mary calling are conditionally independent, given the alarm.
- You often need to think in terms of the atomic events in order to assess conditional independence, since thinking causally can sometimes confuse the issue.

Looking for Conditional Independences

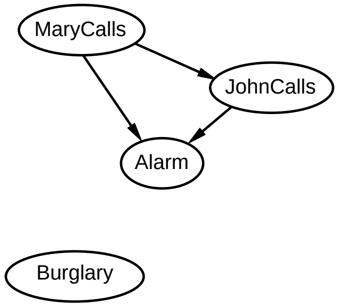
Only hook up nodes when the child is dependent upon the parent(s).
Method $=$ Try to establish conditional independence. If not, hook up the nodes.



$P(J|M) = P(J)$? No
$P(A|J, M) = P(A|J)$? $P(A|J, M) = P(A)$?
From the data set:

- $P(A = T \mid M = T \land J = T) =$ 1.0
- $P(A = T \mid J = T) = 0.8$ and $P(A = T) = 0.5$
- So A is not conditionally independent of J (Given M), and it is not independent of $J \land M$

Adding the Burglary Node

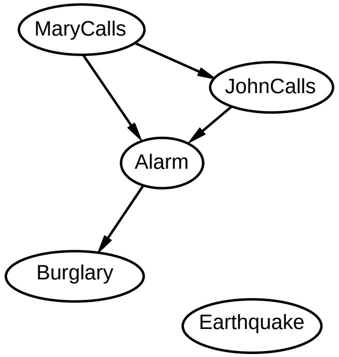


$P(J|M) = P(J)$? No
$P(A|J,M) = P(A|J)$? $P(A|J,M) = P(A)$? No
$P(B|A,J,M) = P(B|A)$? Is B cond indep of J and M, given A?
$P(B|A,J,M) = P(B)$?
From the data set:

- $P(B = T \mid A = T \wedge J = T \wedge M = T) = 0.75$
- $P(B = T \mid A = T) = 0.6$
- $P(B = T) = 0.3$ So B is clearly not independent of A, J and M.
- Still, it is quite plausible, in a larger data set, that $P(B|A,J,M) \approx P(B|A)$
- We would need to test with all possible values of A,J,M and B to verify cond indep.

Adding the Earthquake Node

Suppose we choose the ordering M, J, A, B, E



$P(J|M) = P(J)$? No

$P(A|J, M) = P(A|J)$? $P(A|J, M) = P(A)$? No

$P(B|A, J, M) = P(B|A)$? Yes

$P(B|A, J, M) = P(B)$? No

$P(E|B, A, J, M) = P(E|A)$?

$P(E|B, A, J, M) = P(E|A, B)$?

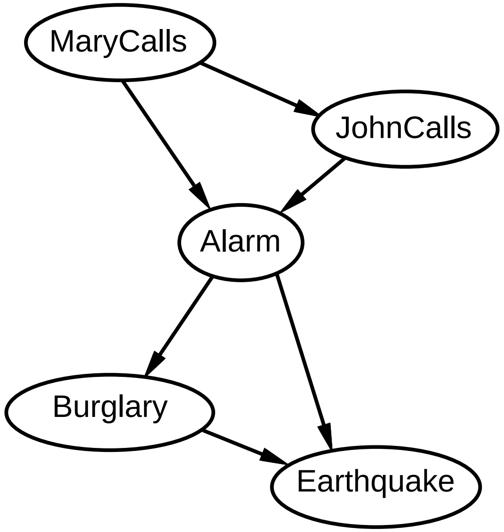
Adding the Earthquake Node (2)

From the data set:

- $P(E = T \mid B = F \wedge A = T \wedge J = T \wedge M = T) = 1.0$
- $P(E = T \mid A = T) = 0.4$
- $P(E = T \mid A = T \wedge B = F) = 1.0$
- So E is cond indep of J and M, given A and B.
- To verify conditional independence, we should verify with all values of all vars to be sure.
- But with such a small data set, chances are that we would not get a perfect indicator.

Complete Earthquake Bayesian Net

Suppose we choose the ordering M, J, A, B, E



$P(J|M) = P(J)$? No

$P(A|J, M) = P(A|J)$? $P(A|J, M) = P(A)$? No

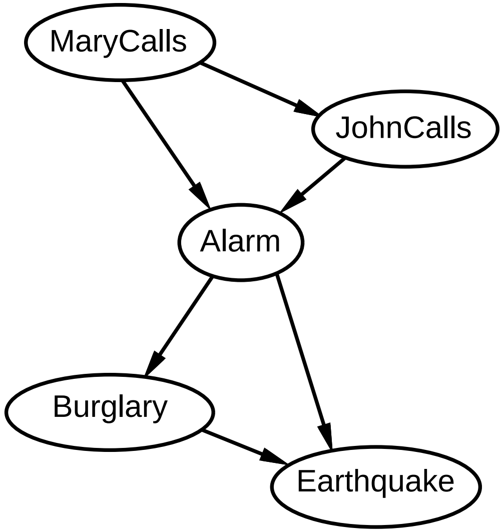
$P(B|A, J, M) = P(B|A)$? Yes

$P(B|A, J, M) = P(B)$? No

$P(E|B, A, J, M) = P(E|A)$? No

$P(E|B, A, J, M) = P(E|A, B)$? Yes

Aftermath of the Earthquake



Deciding conditional independence and conditional probs is hard in noncausal directions

Causal models and conditional independence seem hardwired for humans!

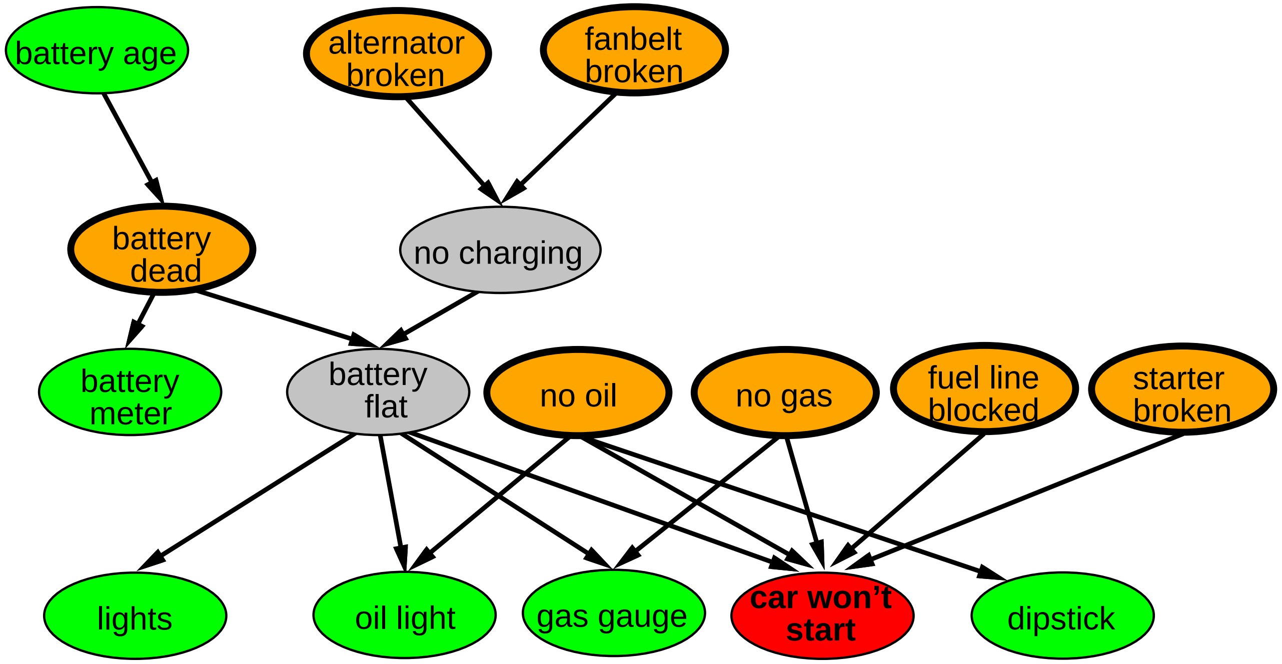
Network is less compact: 1(M) + 2(J) + 4(A) + 2(B) + 4(E) = 13 priors needed. The original (causal) network needed only 10 priors!

Example: Car diagnosis

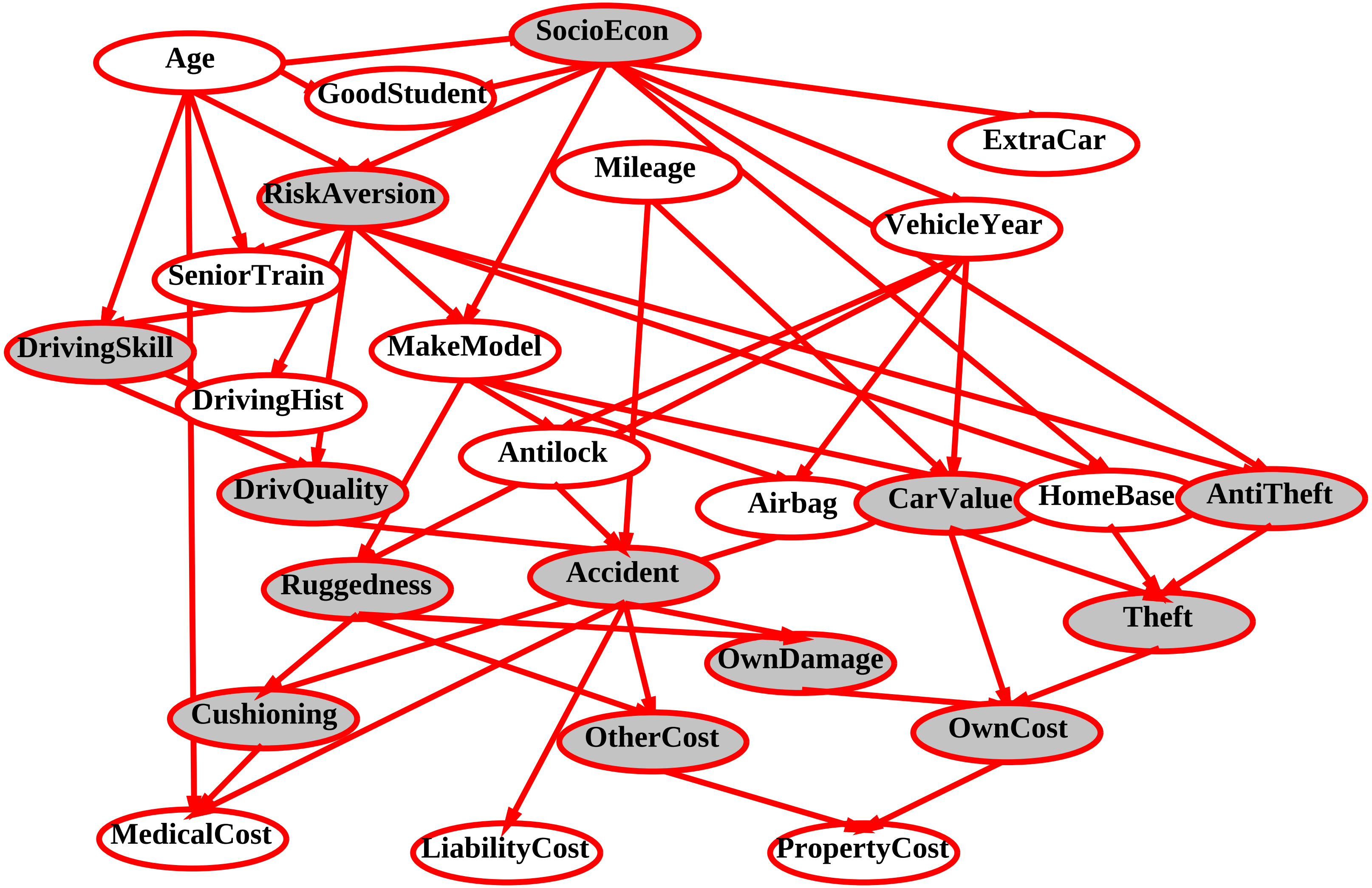
Initial evidence: car won't start

Testable variables (green), "broken, so fix it" variables (orange)

Hidden variables (gray) ensure sparse structure, reduce parameters



Example: Car insurance



Example: Nuclear Power Plant

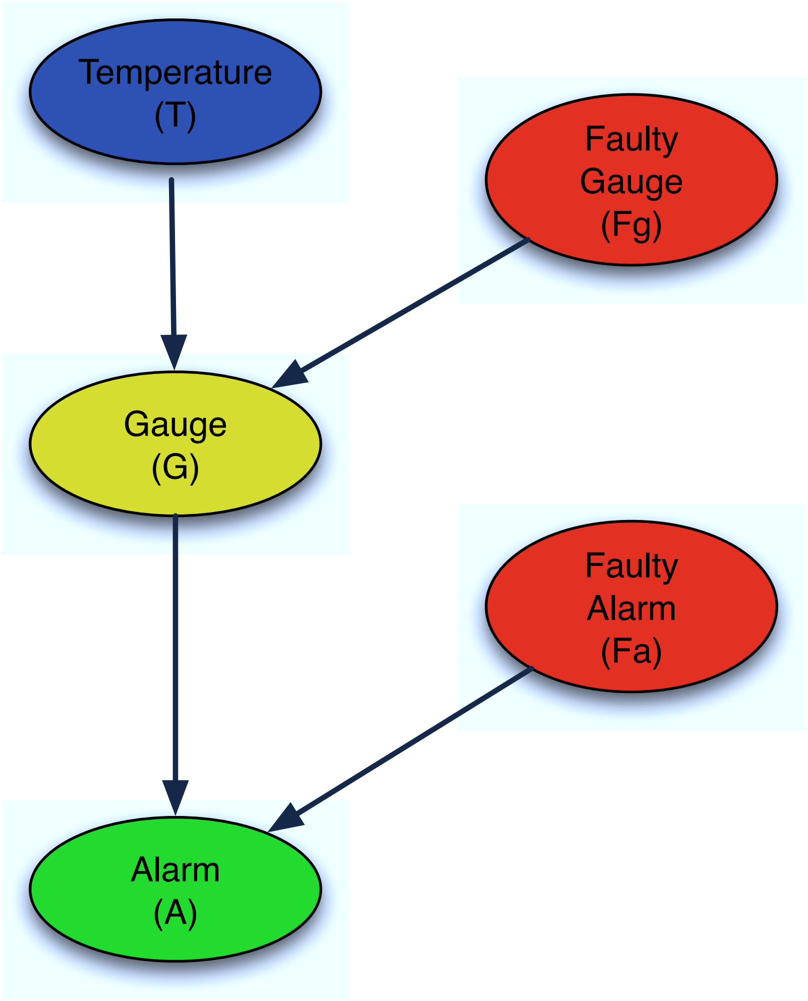
Variables:

- T - core temperature (normal, high)
- G - gauge that records T (normal, high)
- Fg - Faulty gauge (true, false)
- A - alarm that sounds when gauge reading is high (on, off)
- Fa - faulty alarm (true, false)

Draw a Bayesian Network that captures the correct causal dependencies. Draw the conditional probability tables for G and A given that:

- Probability that G gives correct temperature when it is working is x, and when faulty, y.
- Alarm works correctly when not faulted. But when faulted, it never rings.

Nuclear Power Plant Bayesian Network



Nuclear Power Plant CPTs

	T=normal		T=high	
	F_g	$\neg F_g$	F_g	$\neg F_g$
G=normal	y	x	1-y	1-x
G=high	1-y	1-x	y	x

	G=normal		G=high	
	F_a	$\neg F_a$	F_a	$\neg F_a$
Alarm = T	0	0	0	1
Alarm = F	1	1	1	0

Inference Using Bayesian Networks

- ♢ Exact inference by enumeration
- ♢ Approximate inference by stochastic simulation

Inference tasks

Simple queries: compute posterior marginal $\mathbf{P}(X_i|\mathbf{E}=\mathbf{e})$

e.g., $P(NoGas|Gauge=empty, Lights=on, Starts=false)$

Conjunctive queries: $\mathbf{P}(X_i, X_j|\mathbf{E}=\mathbf{e}) = \mathbf{P}(X_i|\mathbf{E}=\mathbf{e})\mathbf{P}(X_j|X_i, \mathbf{E}=\mathbf{e})$

Optimal decisions: decision networks include utility information;

probabilistic inference required for $P(outcome|action, evidence)$

Value of information: which evidence to seek next?

Sensitivity analysis: which probability values are most critical?

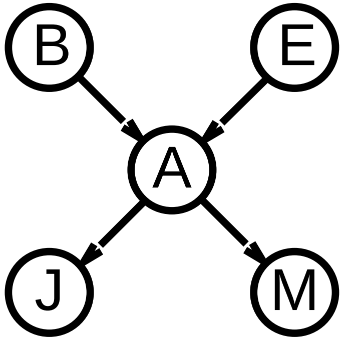
Explanation: why do I need a new starter motor?

Inference by enumeration

Slightly intelligent way to sum out variables from the joint without actually constructing its explicit representation

Simple query on the burglary network:

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



Rewrite full joint entries using product of CPT entries:

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

Recursive depth-first enumeration: $O(n)$ space, $O(d^n)$ time

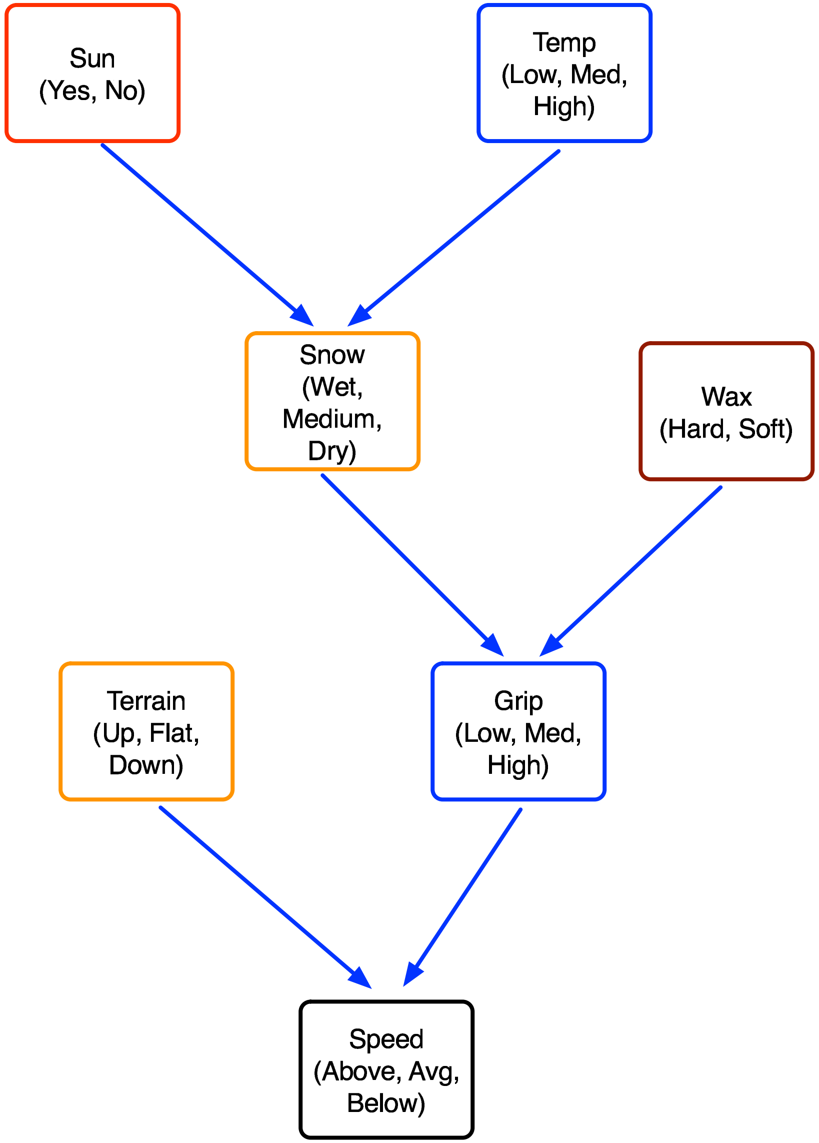
Enumeration algorithm

```
function ENUMERATION-ASK($X$, $\mathbf{e}$, $bn$) returns a distribution over $X$
    inputs: $X$, the query variable
            $\mathbf{e}$, observed values for variables $\mathbf{E}$
            $bn$, a Bayesian network with variables $\{X\} \cup \mathbf{E} \cup \mathbf{Y}$

    $\mathbf{Q}(X) \leftarrow$ a distribution over $X$, initially empty
    for each value $x_i$ of $X$ do
        extend $\mathbf{e}$ with value $x_i$ for $X$
        $\mathbf{Q}(x_i) \leftarrow$ ENUMERATE-ALL(VARS[$bn$], $\mathbf{e}$)
    return NORMALIZE($\mathbf{Q}(X)$)
```

```
function ENUMERATE-ALL($vars$, $\mathbf{e}$) returns a real number
    if EMPTY?($vars$) then return 1.0
    $Y \leftarrow$ FIRST($vars$)
    if $Y$ has value $y$ in $\mathbf{e}$
        then return $P(y \mid Pa(Y)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}$)
        else return $\Sigma_y \; P(y \mid Pa(Y)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}_y$)
            where $\mathbf{e}_y$ is $\mathbf{e}$ extended with $Y = y$
```

Enumeration Example: Skiing (1)



Enumeration Example: Skiing (2)

Ignore 4 factors (Wax, Terrain, Grip and Speed) just to keep this example simple.
Query: $P(Sun = Yes|Snow = Dry))$
Same as:
$P(Sun = Yes|Snow = Dry \wedge Temp = High)$
$+P(Sun = Yes|Snow = Dry \wedge Temp = Medium)$
$+P(Sun = Yes|Snow = Dry \wedge Temp = Low)$

But we do not have these probabilities!
What we do have are tables for each of these probabilities:
1) $P(Sun = Yes/No)$
2) $P(Temp = Low/Medium/High)$
3) $P(Snow = Wet/Med/Dry|Sun = Y/N \wedge Temp = Low/Med/High)$

Enumeration Example: Skiing (3)

Enumeration-Ask called with:
X = Sun
e = (Snow = Dry)
bn = the simplified Bayesian Net with only Sun, Temp and Snow.
VARS(bn) = (Sun, Temp, Snow) in THAT order: parents before kids.

Enumerate-All called 2 times from Enumerate-Ask, with e (the evidence) having 2 different values:
1) (Snow = Dry, Sun = Yes)
2) (Snow = Dry, Sun = No)
This yields 2 results:
1) $P(Snow = Dry \wedge Sun = Yes)$
2) $P(Snow = Dry \wedge Sun = No)$
Note that: $P(Snow = Dry \wedge Sun = Yes) + P(Snow = Dry \wedge Sun = No) = P(Snow = Dry)$
Since Sun only takes on these 2 values.

Enumeration Example: Skiing (4)

So when we normalize (i.e. when we divide each result by their sum), we are dividing by

$P(Snow = Dry)$

And thus, we are computing:

1) $\frac{P(Snow=Dry \wedge Sun=Yes)}{P(Snow=Dry)} = P(Sun = Yes|Snow = Dry)$

2) $\frac{P(Snow=Dry \wedge Sun=No)}{P(Snow=Dry)} = P(Sun = No|Snow = Dry)$

And 1 is exactly the query we wanted to answer!

First call to EnumerateAll with e = (Snow = Dry, Sun = Yes):

Sun = first(vars) and Sun $\in$ e, so the recursive calculation is:

$P(Sun = Y) \times EnumerateAll\{(temp, snow), e + (Sun = Yes)\}$

Second call to EnumerateAll with e = (Snow = Dry, Sun = Yes):

Temp = first(vars) and Temp $\notin$ e, so the recursive calculation is:

$\sum_{X\in\{H,M,L\}} (P(Temp = X) \times EnumerateAll\{(snow), e + (Temp = X)\})$

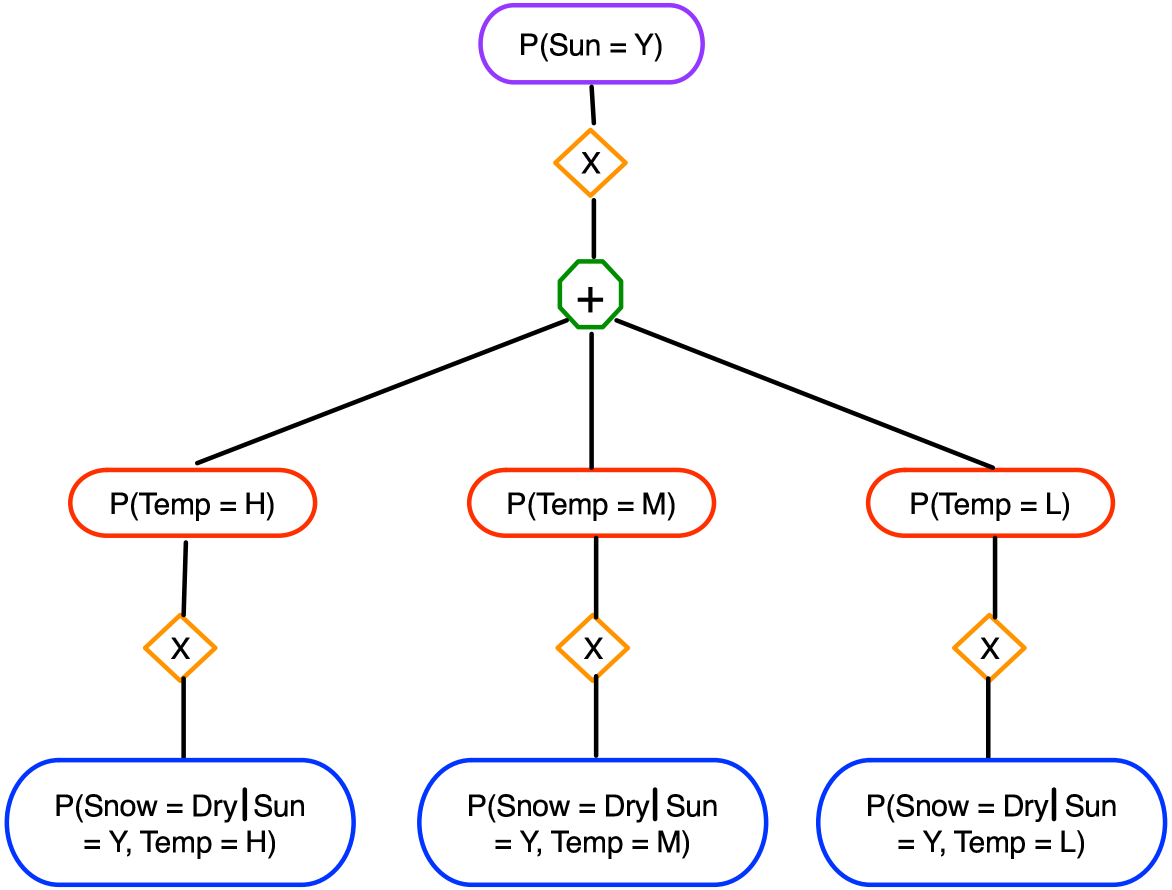
Enumeration Example: Skiing (5)

3rd, 4th and 5th calls to EnumerateAll with
e = (Snow = Dry, Sun = Yes, Temp = X):
Snow = first(vars) and Snow $\in$ e, so the recursive calculation is:
$P(Snow = Dry) \times EnumerateAll\{\{\,\}, e + (Snow = Dry)\}$

6th, 7th and 8th calls to EnumerateAll with
e = (Snow = Dry, Sun = Yes, Temp = X)
and vars = $\{\,\}$ are just base cases,
so all return 1.0.

Enumeration Example: Skiing (6)

Viewing the recursive calls to Enumerate-All as a tree:

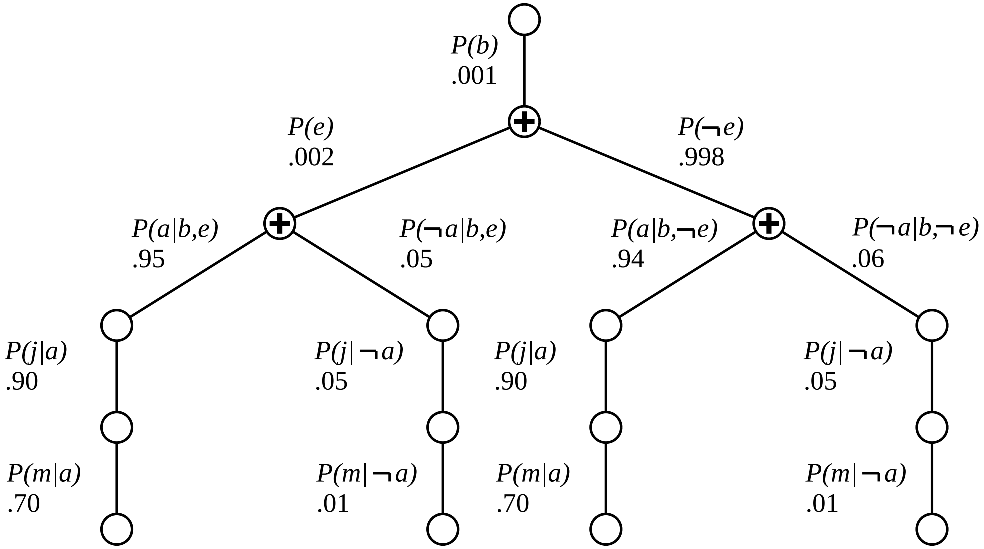

P(Sun = Y)
X
+
P(Temp = H)
P(Temp = M)
P(Temp = L)
X
X
X
P(Snow = Dry | Sun = Y, Temp = H)
P(Snow = Dry | Sun = Y, Temp = M)
P(Snow = Dry | Sun = Y, Temp = L)

*Note: all probs in tree are directly available from the tables of the Bayesian Net.

Evaluation tree

Enumeration is inefficient: repeated computation

e.g., computes $P(j|a)P(m|a)$ for each value of e

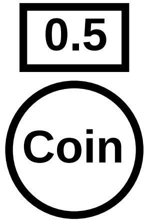


Solution: Use Variable-Elimination Algorithms (See book).

Inference by stochastic simulation

Basic idea:

1) Draw N samples from a sampling distribution S



2) Compute an approximate posterior probability $\hat{P}$

3) 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
- Markov chain Monte Carlo (MCMC): sample from a stochastic process whose stationary distribution is the true posterior

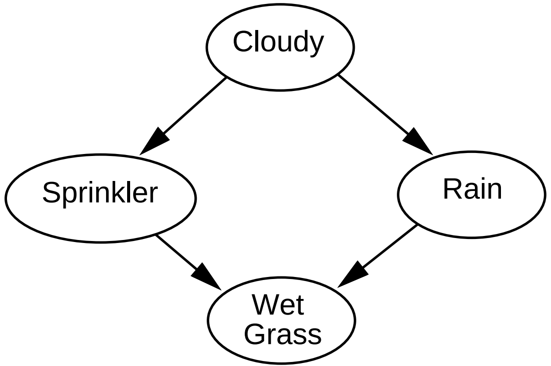
Sampling from an empty network

```
function PRIOR-SAMPLE($bn$) returns an event sampled from $bn$
    inputs: $bn$, a belief network specifying joint distribution $\mathbf{P}(X_1, \ldots, X_n)$

    $\mathbf{x} \leftarrow$ an event with $n$ elements
    for $i = 1$ to $n$ do
        $x_i \leftarrow$ a random sample from $\mathbf{P}(X_i \mid Parents(X_i))$
    return $\mathbf{x}$
```

Example

P(C)
.50



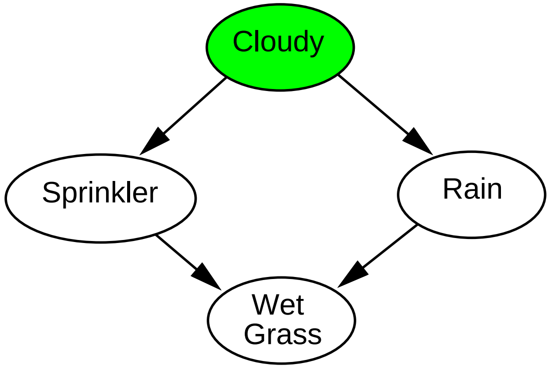
C	P(S|C)
T	.10
F	.50

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

Example

P(C)
.50

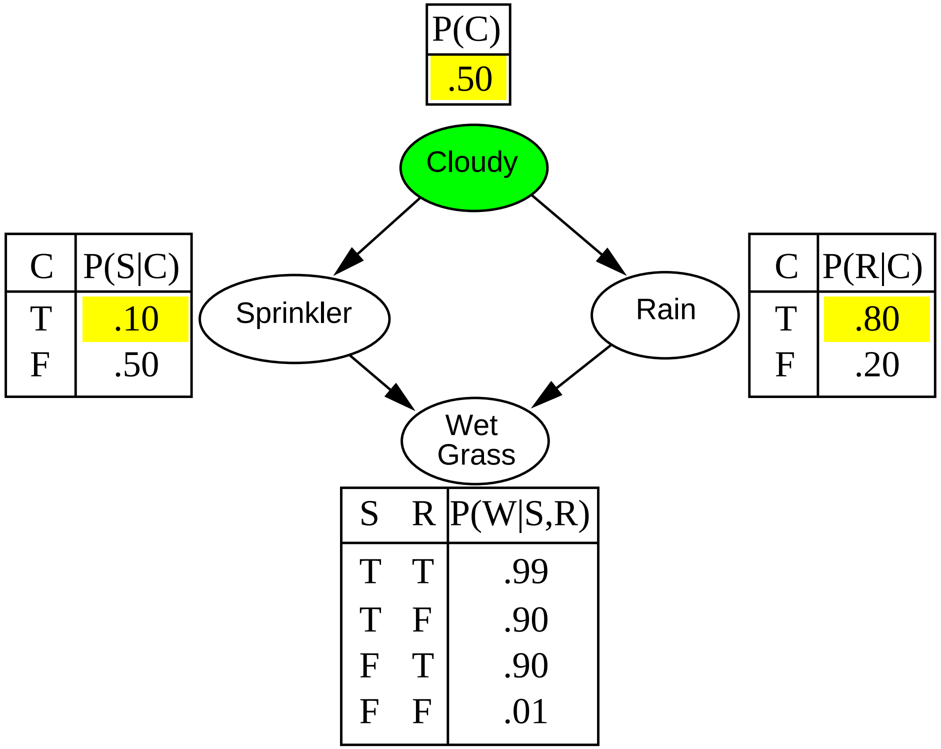


C	P(S|C)
T	.10
F	.50

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

Example



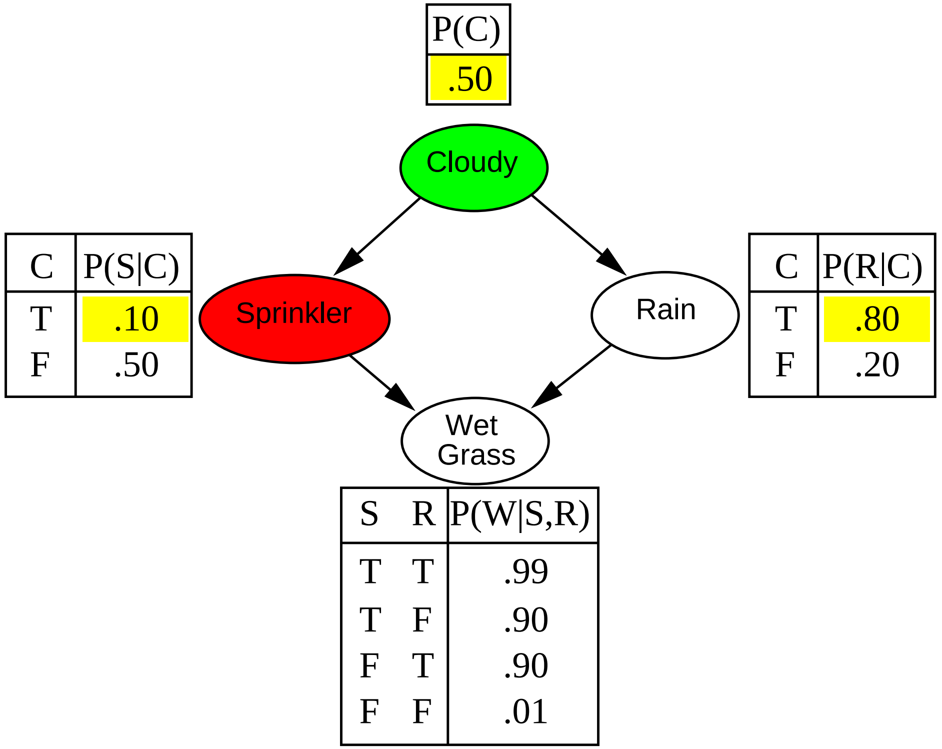
P(C)
.50

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

Example



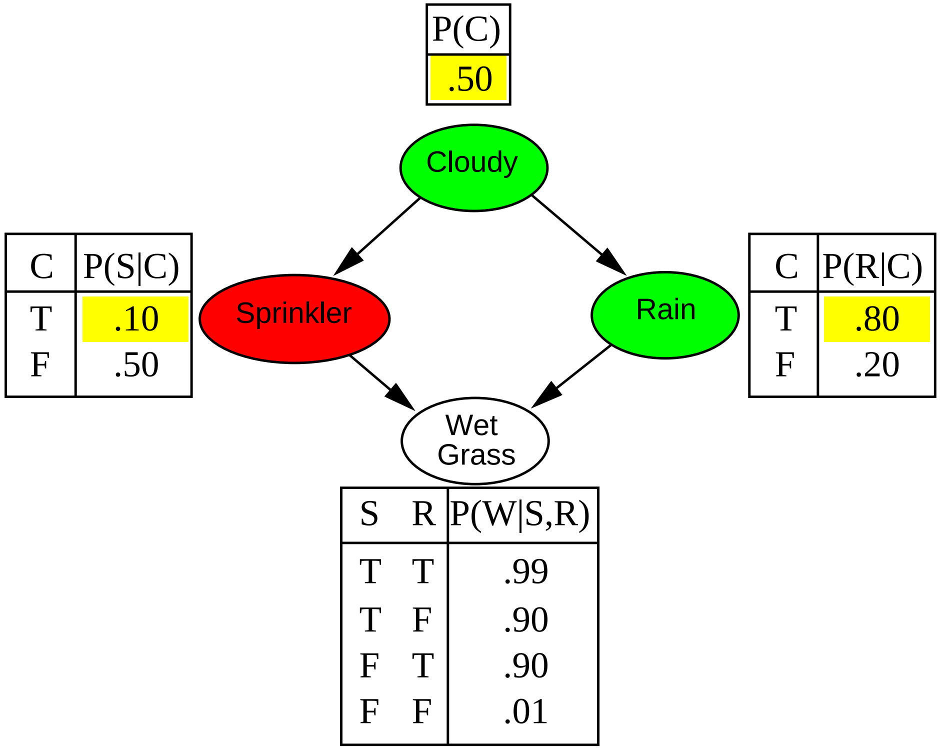
P(C)
.50

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

Example



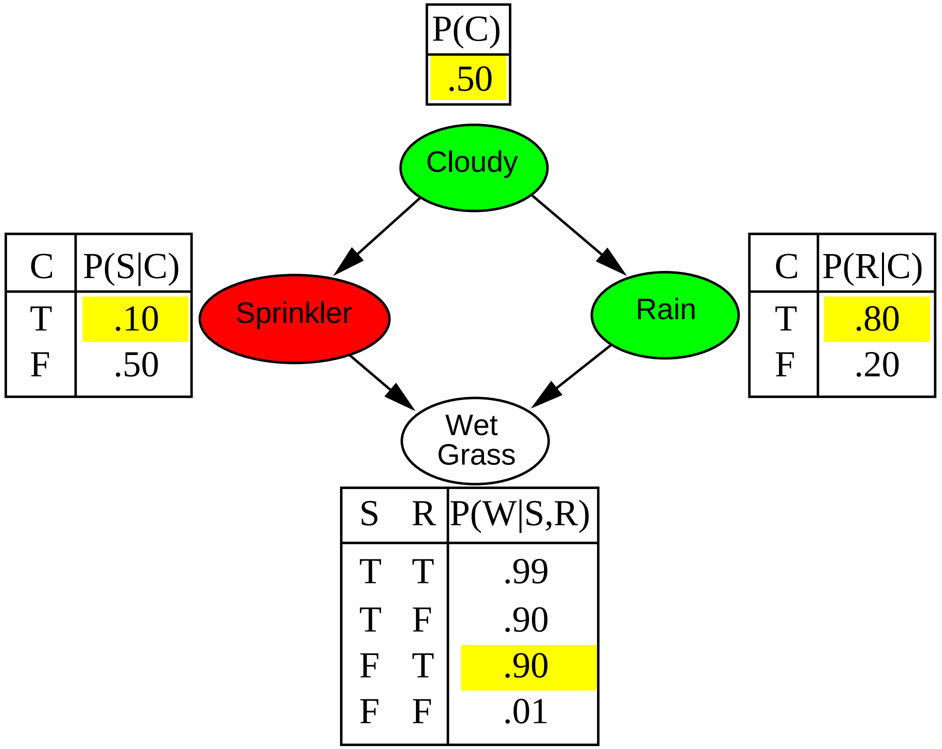
P(C)
.50

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

Example



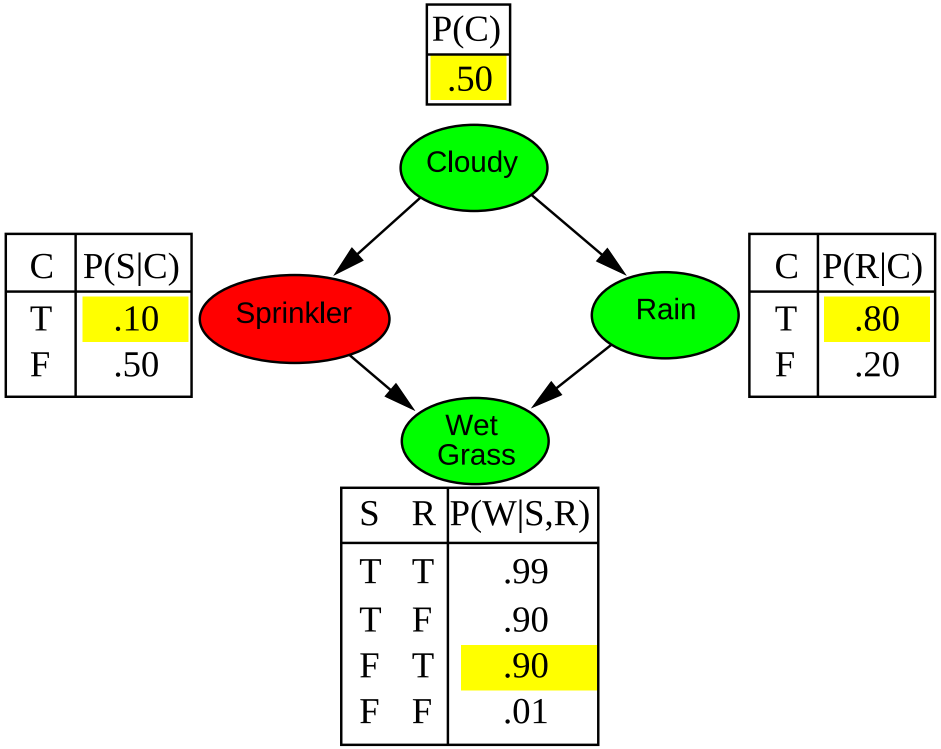
P(C)
.50

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

Example



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

Sampled Event = (Cloudy = T, Sprinker = F, Rain = T, Wet-Grass = T)
Reset variables and sample again, and again, and...

Sampling from an empty network contd.

Probability that PriorSample generates a particular event

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

i.e., the true prior probability

E.g., $S_{PS}(t,f,t,t) = 0.5 \times 0.9 \times 0.8 \times 0.9 = 0.324 = P(t,f,t,t)$

Let $N_{PS}(x_1 \ldots x_n)$ be the number of samples generated for event $x_1, \ldots, x_n$

Then we have

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

That is, estimates derived from PriorSample are consistent

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

Rejection sampling

$\hat{\mathbf{P}}(X|\mathbf{e})$ estimated from samples agreeing with $\mathbf{e}$

```
function REJECTION-SAMPLING($X$, $\mathbf{e}$, $bn$, $N$) returns an estimate of $P(X|\mathbf{e})$
    local variables: $\mathbf{N}$, a vector of counts over $X$, initially zero

    for $j$ = 1 to $N$ do
        $\mathbf{x} \leftarrow$ PRIOR-SAMPLE($bn$)
        if $\mathbf{x}$ is consistent with $\mathbf{e}$ then
            $\mathbf{N}[x] \leftarrow \mathbf{N}[x]+1$ where $x$ is the value of $X$ in $\mathbf{x}$
    return NORMALIZE($\mathbf{N}[X]$)
```

E.g., estimate $\mathbf{P}(Rain|Sprinkler = true)$ using 100 samples

27 samples have $Sprinkler = true$

Of these, 8 have $Rain = true$ and 19 have $Rain = false$.

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

Similar to a basic real-world empirical estimation procedure

Analysis of rejection sampling

$\hat{\mathbf{P}}(X|\mathbf{e}) = \alpha \mathbf{N}_{PS}(X, \mathbf{e})$ (algorithm defn.)

$= \mathbf{N}_{PS}(X, \mathbf{e})/N_{PS}(\mathbf{e})$ (normalized by $N_{PS}(\mathbf{e})$)

$\approx \mathbf{P}(X, \mathbf{e})/P(\mathbf{e})$ (property of PriorSample)

$= \mathbf{P}(X|\mathbf{e})$ (defn. of conditional probability)

Hence rejection sampling returns consistent posterior estimates

Problem: hopelessly expensive if $P(\mathbf{e})$ is small

$P(\mathbf{e})$ drops off exponentially with number of evidence variables!

Likelihood weighting

Idea: fix evidence variables, sample only nonevidence variables,
and weight each sample by the likelihood it accords the evidence

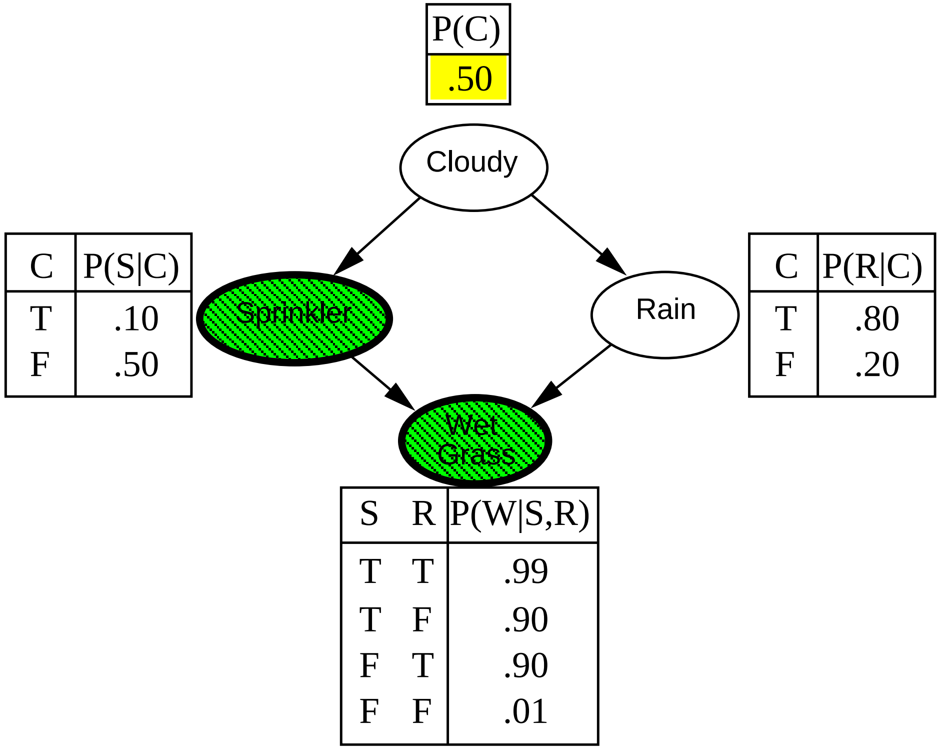
```
function LIKELIHOOD-WEIGHTING($X$, $\mathbf{e}$, $bn$, $N$) returns an estimate of $P(X|\mathbf{e})$
    local variables: $\mathbf{W}$, a vector of weighted counts over $X$, initially zero

    for $j = 1$ to $N$ do
        $\mathbf{x}$, $w \leftarrow$ WEIGHTED-SAMPLE($bn$)
        $\mathbf{W}[x] \leftarrow \mathbf{W}[x] + w$ where $x$ is the value of $X$ in $\mathbf{x}$
    return NORMALIZE($\mathbf{W}[X]$)
```

```
function WEIGHTED-SAMPLE($bn$, $\mathbf{e}$) returns an event and a weight

    $\mathbf{x} \leftarrow$ an event with $n$ elements; $w \leftarrow 1$
    for $i = 1$ to $n$ do
        if $X_i$ has a value $x_i$ in $\mathbf{e}$
            then $w \leftarrow w \times\ P(X_i = x_i \mid Parents(X_i))$
            else $x_i \leftarrow$ a random sample from $\mathbf{P}(X_i \mid Parents(X_i))$
    return $\mathbf{x}$, $w$
```

Likelihood weighting example



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

$w = 1.0$

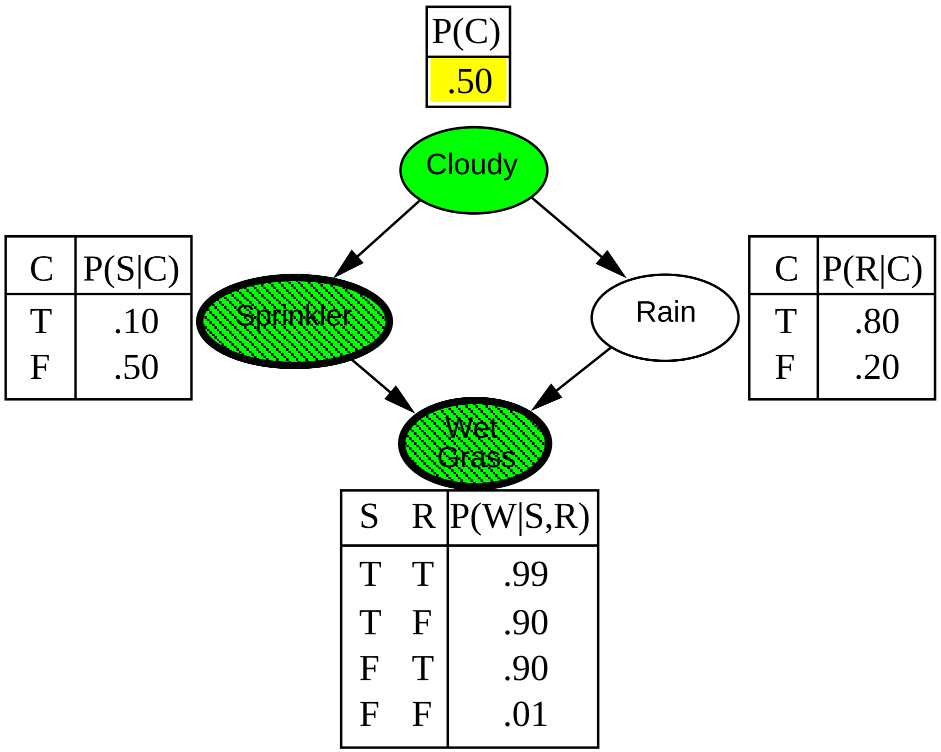
Note: Only update the weight when an EVIDENCE variable is reached.

That weight replaces an actual sample.

Non-evidence vars are actually sampled, but that does not affect the weight.

So each variable contributes by either weighting or sampling, not both.

Likelihood weighting example



P(C)
.50

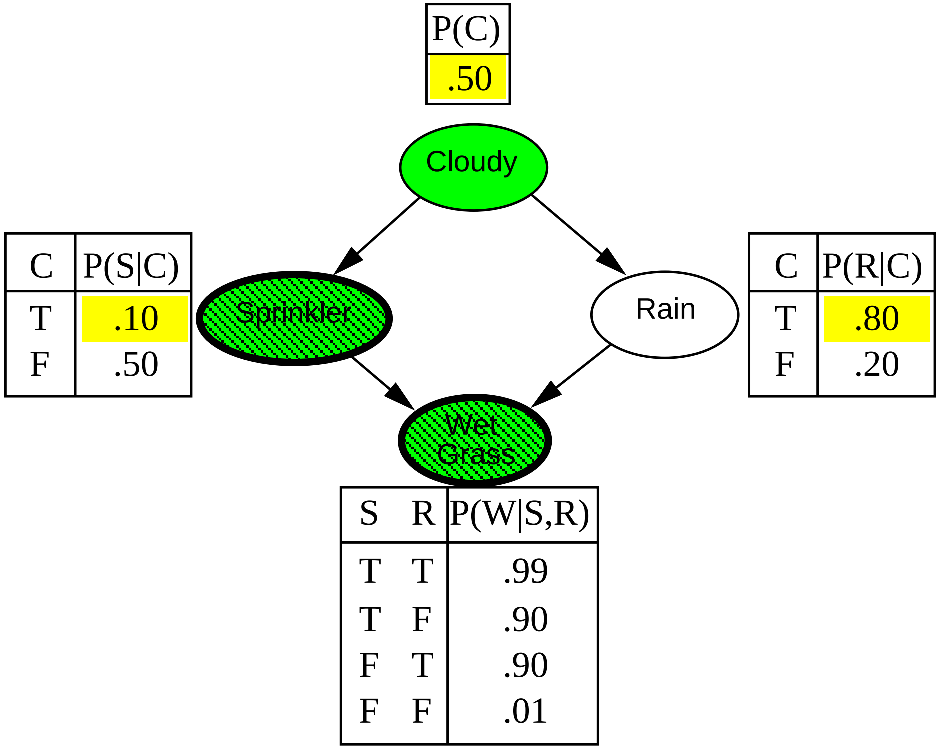
C	P(S|C)
T	.10
F	.50

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

$w = 1.0$

Likelihood weighting example



P(C)
.50

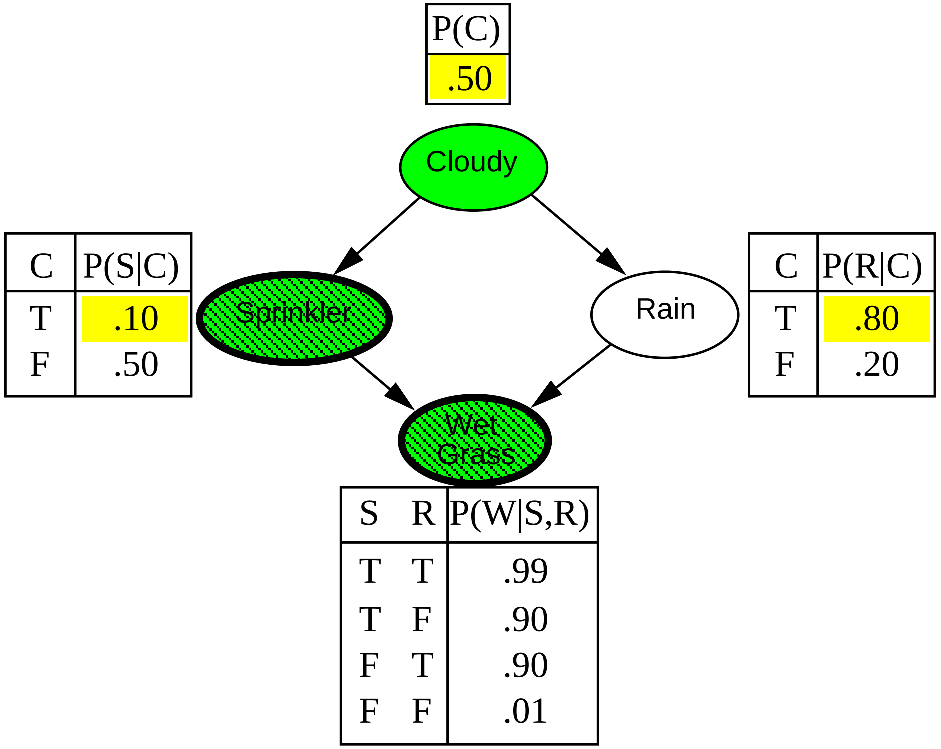
C	P(S|C)
T	.10
F	.50

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

$w = 1.0$

Likelihood weighting example



P(C)
.50

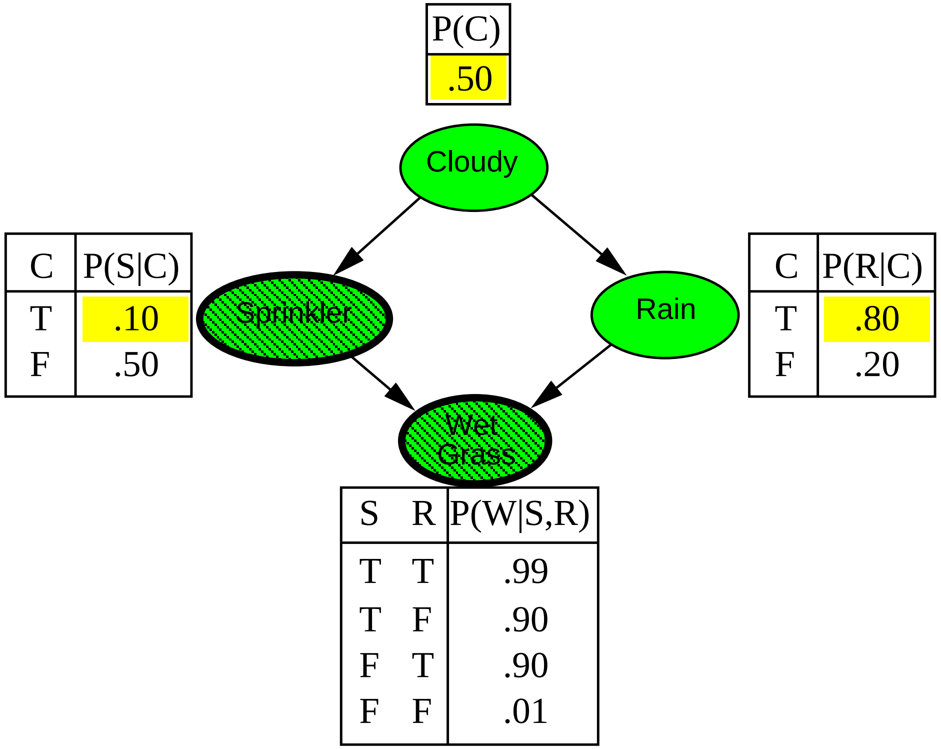
C	P(S|C)
T	.10
F	.50

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

$w = 1.0 \times 0.1$

Likelihood weighting example



P(C)
.50

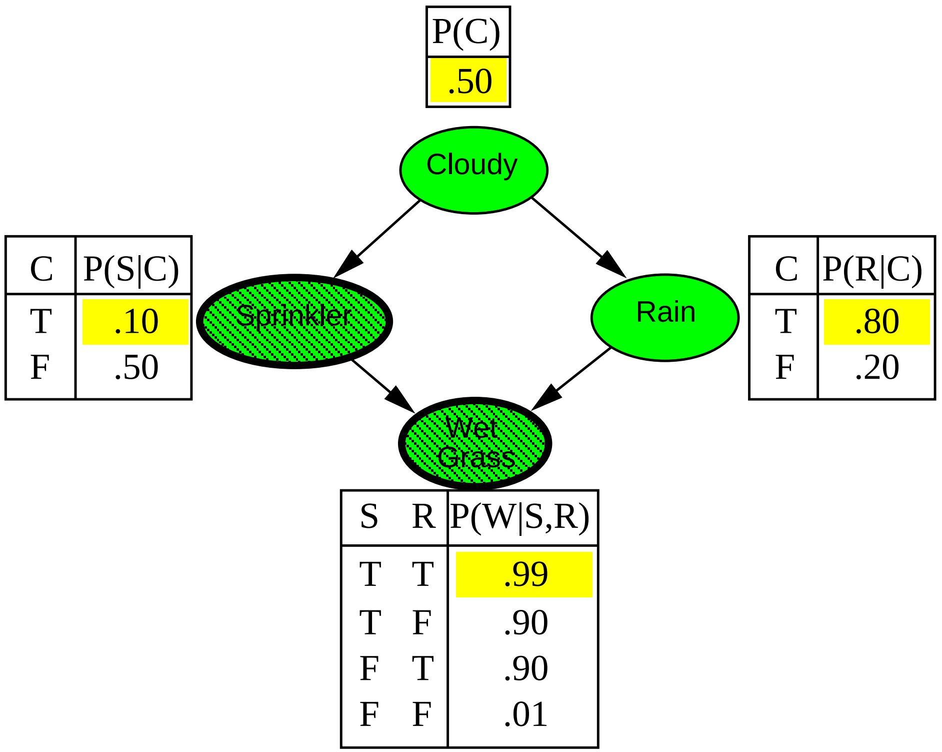
C	P(S|C)
T	.10
F	.50

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

$w = 1.0 \times 0.1$

Likelihood weighting example



P(C)
.50

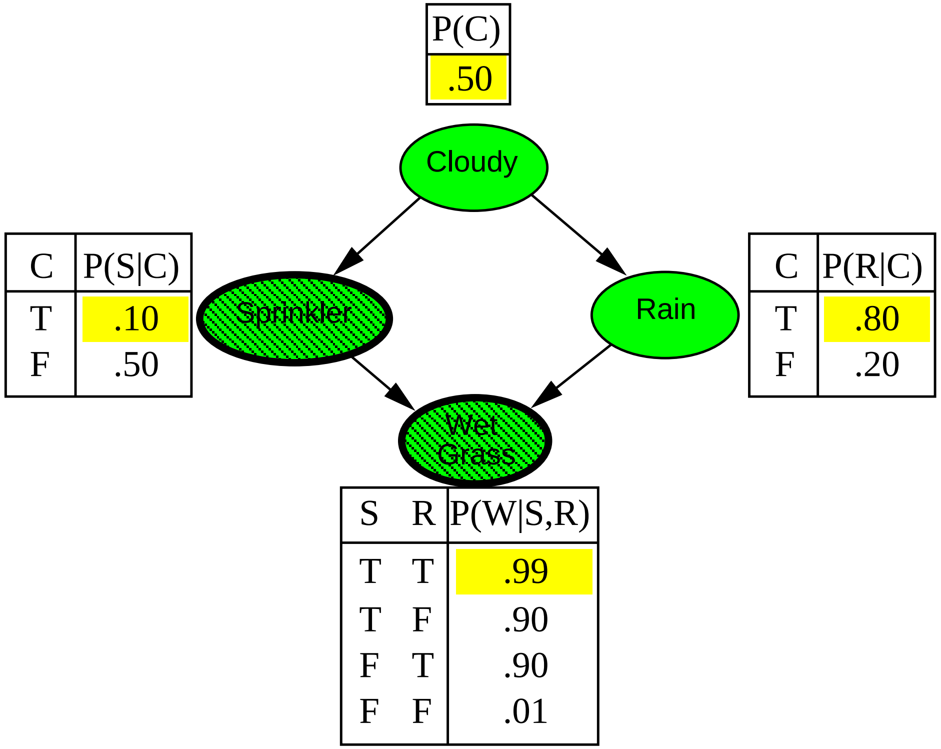
C	P(S|C)
T	.10
F	.50

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

$$w = 1.0 \times 0.1$$

Likelihood weighting example



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

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

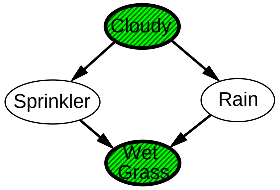
Likelihood weighting analysis

Sampling probability for WeightedSample is

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

Note: pays attention to evidence in **ancestors** only

$\Rightarrow$ somewhere "in between" prior and posterior distribution



Weight for a given sample $\mathbf{z}, \mathbf{e}$ is

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

Weighted sampling probability is

$$S_{WS}(\mathbf{z}, \mathbf{e}) w(\mathbf{z}, \mathbf{e})$$
$$= \prod_{i=1}^{l} P(z_i | Parents(Z_i)) \ \prod_{i=1}^{m} P(e_i | Parents(E_i))$$
$$= P(\mathbf{z}, \mathbf{e}) \text{ (by standard global semantics of network)}$$

Hence likelihood weighting returns consistent estimates but performance still degrades with many evidence variables because a few samples have nearly all the total weight.