Bayes Networks

Robert Platt
Northeastern University

Some images, slides, or ideas are used from:
1. AIMA
2. Berkeley CS188
3. Chris Amato

What is a Bayes Net?

What is a Bayes Net?

Suppose we're given this distribution:

cavity	P(T,C)	P(T,!C)	P(!T,C)	P(!T,!C)
true	0.16	0.018	0.018	0.002
false	0.048	0.19	0.11	0.448

Variables:
Cavity
Toothache (T)
Catch (C)

What is a Bayes Net?

Suppose

Can we summarize aspects of this probability distribution with a graph?

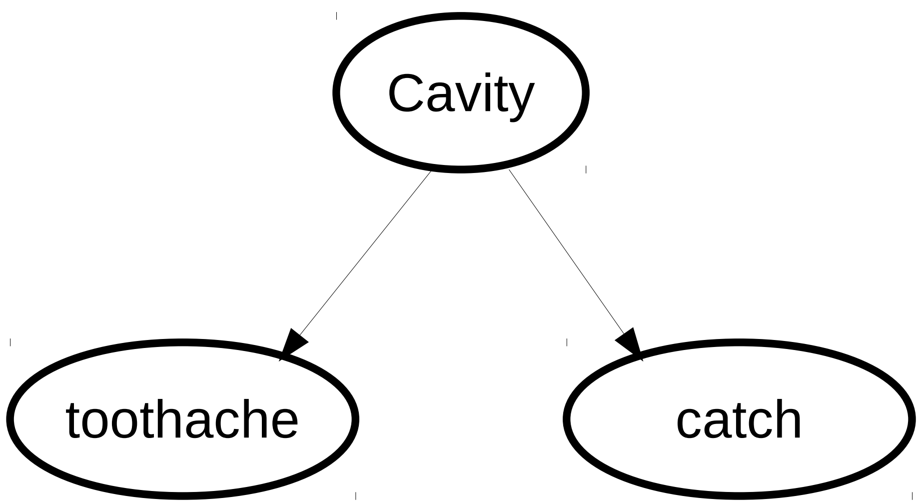
cavi				T,!C)
tru				002
false	0.048	0.19	0.11	0.448

<u>Variables:</u>
Cavity
Toothache (T)
Catch (C)

What is a Bayes Net?

cavity	P(T,C)	P(T,!C)	P(!T,C)	P(!T,!C)
true	0.16	0.018	0.018	0.002
false	0.048	0.19	0.11	0.448

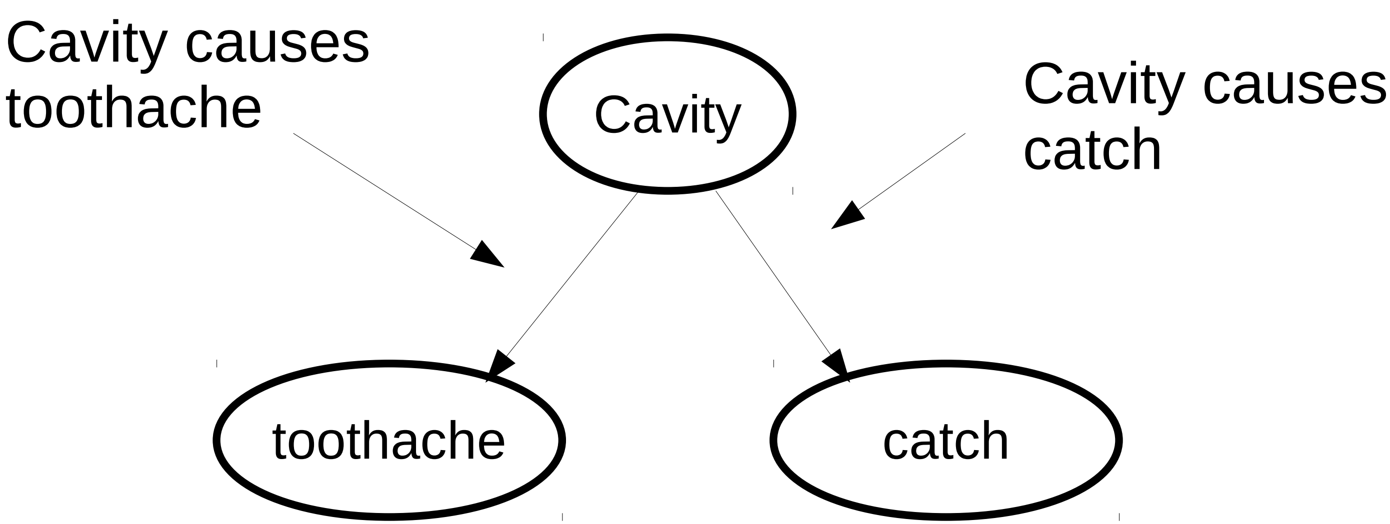
This diagram captures important information
that is hard to extract from table by looking at it:



What is a Bayes Net?

cavity	P(T,C)	P(T,!C)	P(!T,C)	P(!T,!C)
true	0.16	0.018	0.018	0.002
false	0.048	0.19	0.11	0.448

This diagram captures important information that is hard to extract from table by looking at it:

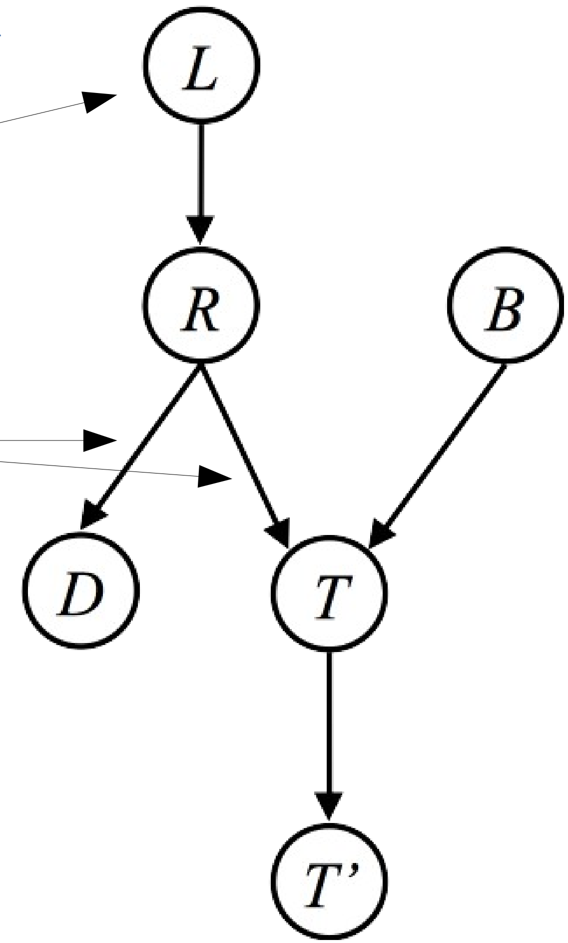


What is a Bayes Net?

Something that looks like this:

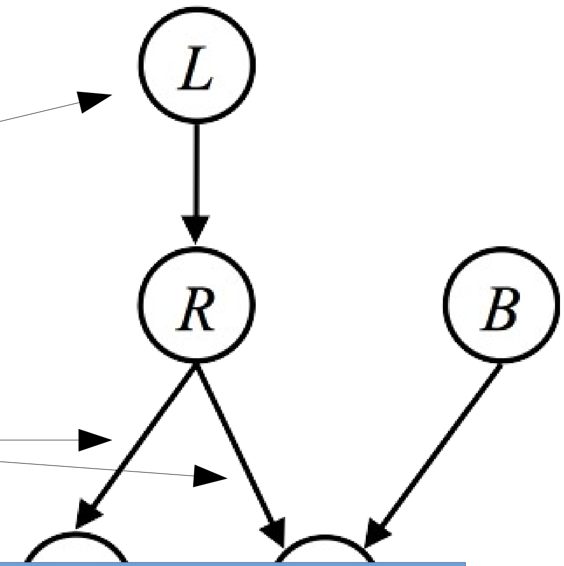
<u>Bubbles</u>: random variables

<u>Arrows</u>: dependency relationships between variables



What is a Bayes Net?

Something that looks like this:



Bubbles: random variables

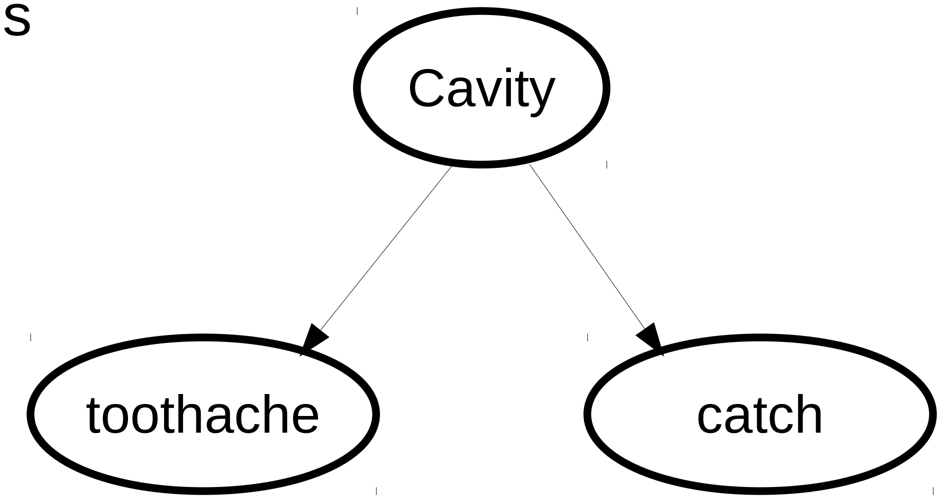
Arrows: dependency relationships between variables

A Bayes net is a compact way of representing a probability distribution

Bayes net example

Diagram encodes the fact that toothache is conditionally independent of catch given cavity

– therefore, all we need are the following distributions



cavity	P(T|cav)
true	0.9
false	0.3

Prob of toothache given cavity

cavity	P(C|cav)
true	0.9
false	0.2

Prob of catch given cavity

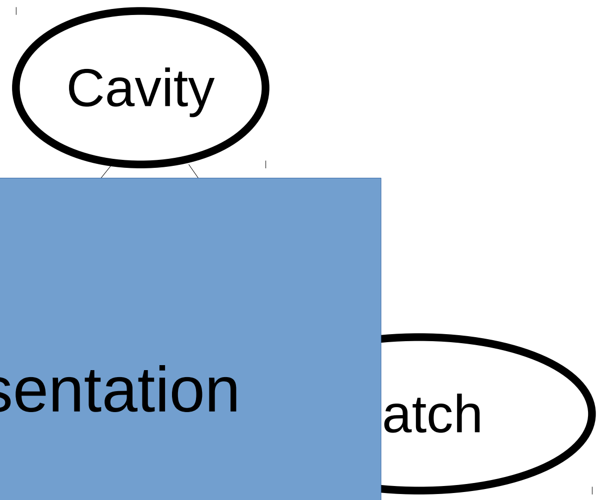
P(cavity) = 0.2

Prior probability of cavity

Bayes net example

Diagram encodes the fact that toothache is conditionally independent of catch given cavity

– therefore, all
distributions



This is called a “factored” representation

cavity	P(T|cav)
true	0.9
false	0.3

Prob of toothache given cavity

cavity	P(C|cav)
true	0.9
false	0.2

Prob of catch given cavity

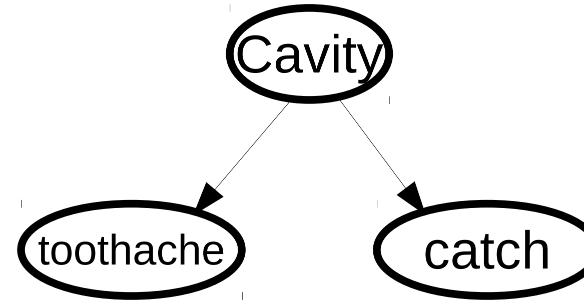
P(cavity) = 0.2

Prior probability of cavity

Bayes net example

cavity	P(T|cav)
true	0.9
false	0.3

cavity	P(C|cav)
true	0.9
false	0.2



P(cavity) = 0.2

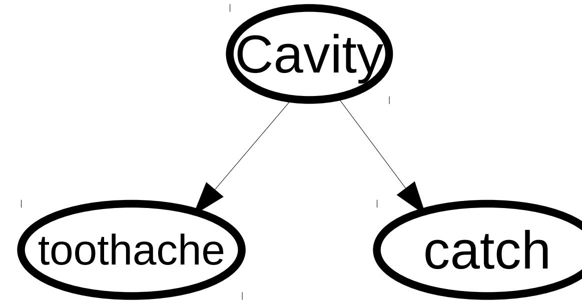
How do we recover joint distribution from factored representation?

cavity	P(T,C)	P(T,!C)	P(!T,C)	P(!T,!C)
true	0.16	0.018	0.018	0.002
false	0.048	0.19	0.11	0.448

Bayes net example

cavity	P(T|cav)
true	0.9
false	0.3

cavity	P(C|cav)
true	0.9
false	0.2



P(cavity) = 0.2

P(T,C,cavity) = P(T,C|cav)P(cav) ← What is this step?

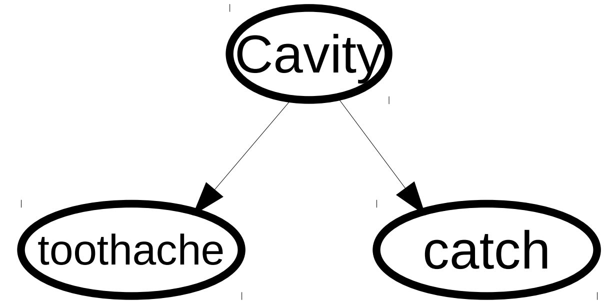
= P(T|cav)P(C|cav)P(cav) ← What is this step?

cavity	P(T,C)	P(T,!C)	P(!T,C)	P(!T,!C)
true	0.16	0.018	0.018	0.002
false	0.048	0.19	0.11	0.448

Bayes net example

cavity	P(T|cav)
true	0.9
false	0.3

cavity	P(C|cav)
true	0.9
false	0.2



P(cavity) = 0.2

P(T,C,cavity) = P(T,C|cav)P(cav)

= P(T|cav)P(C|cav)P(cav)

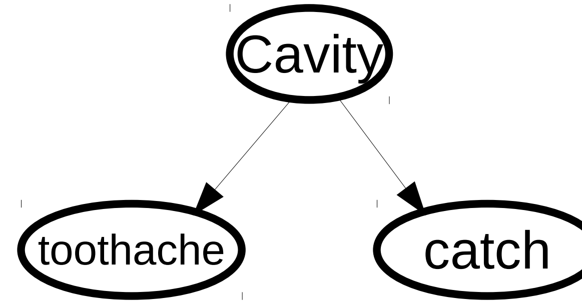
cavity	P(T,C)	P(T,!C)	P(!T,C)	P(!T,!C)
true	0.16	0.018	0.018	0.002
false	0.048	0.19	0.11	0.448

How calculate these?

Bayes net example

cavity	P(T|cav)
true	0.9
false	0.3

cavity	P(C|cav)
true	0.9
false	0.2



vity) = 0.2

P(T,C

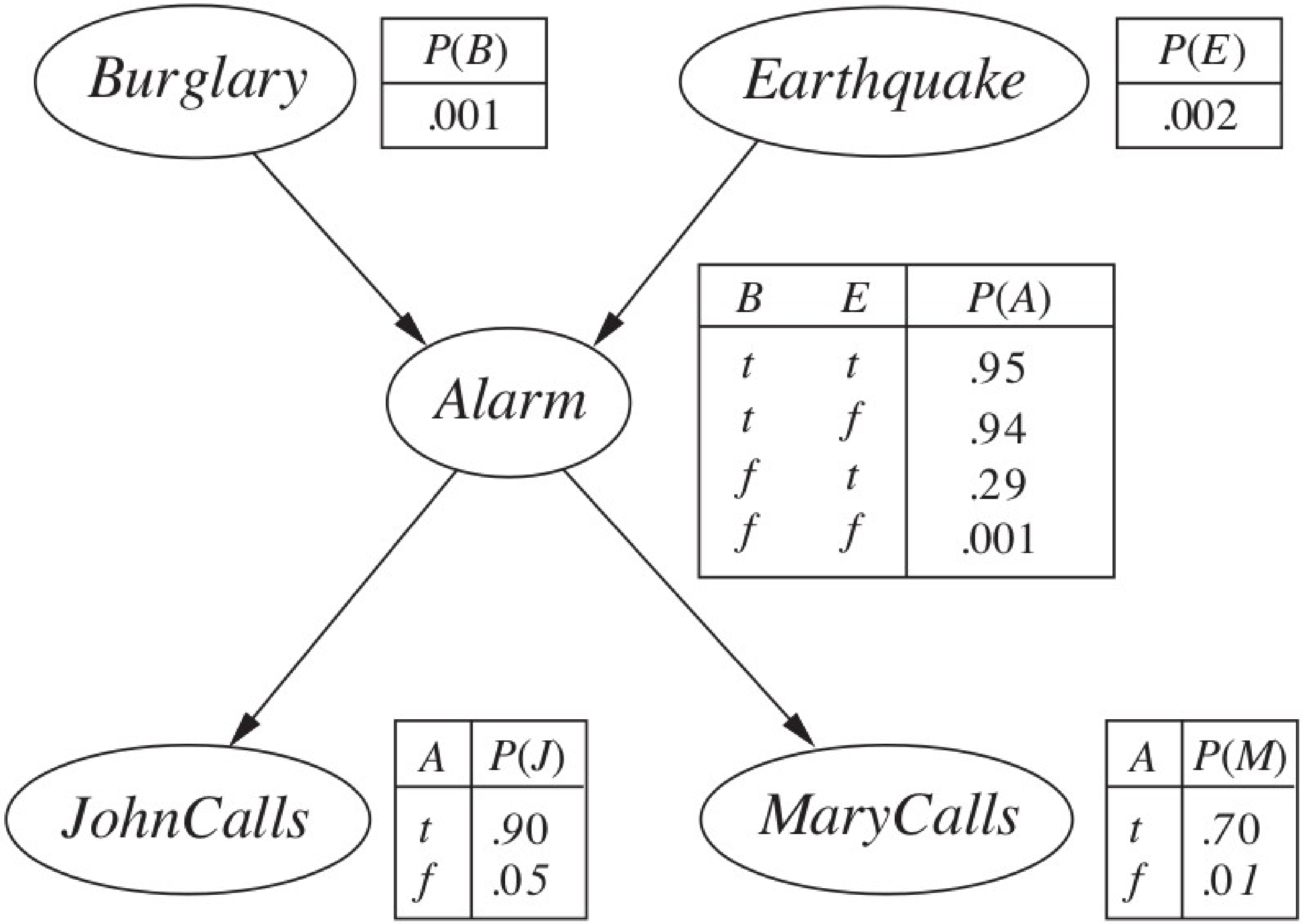
In general:

$$P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i | \mathit{parents}(X_i))$$

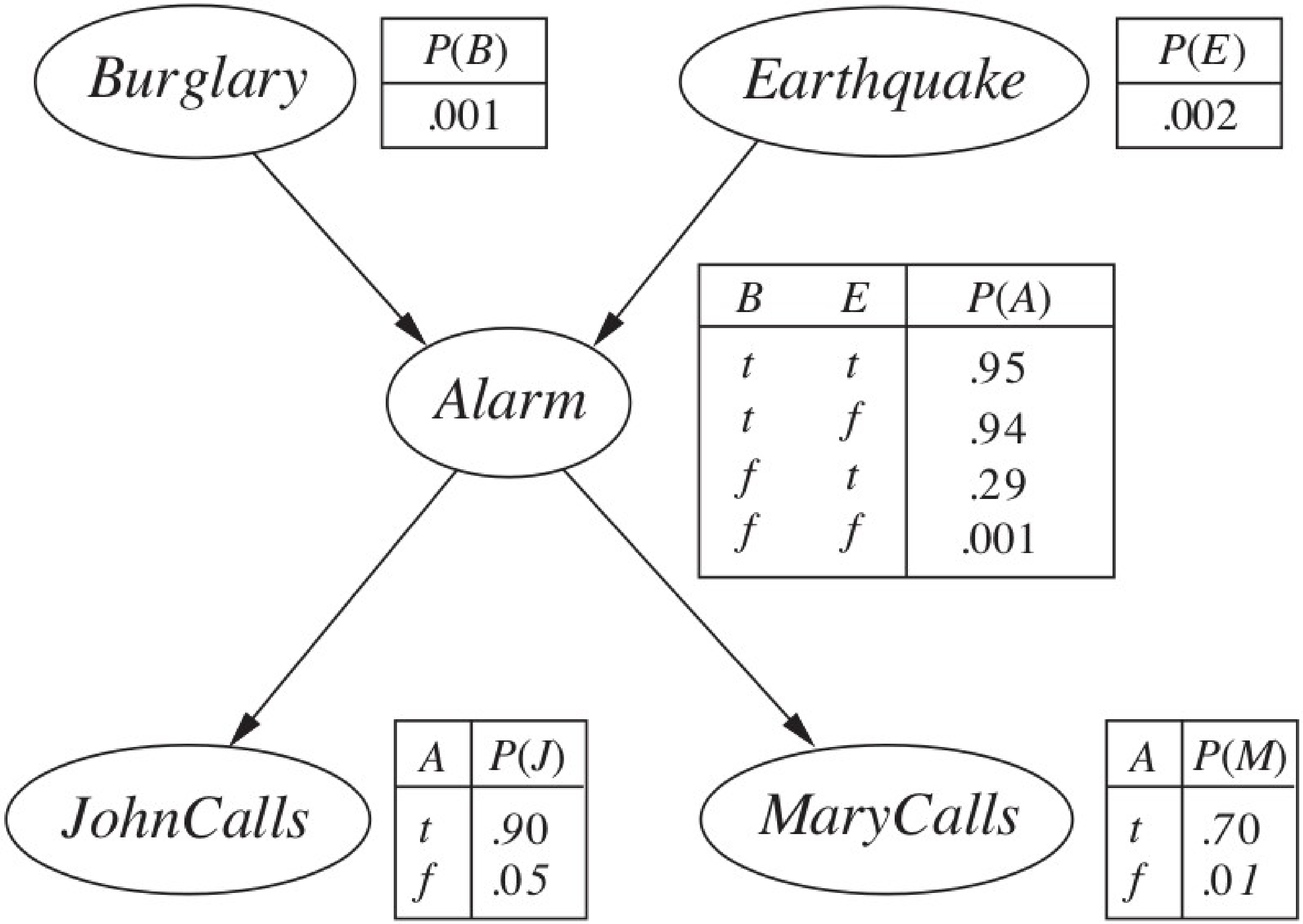
cavity	P(T,C)	P(T,!C)	P(!T,C)	P(!T,!C)
true	0.16	0.018	0.018	0.002
false	0.048	0.19	0.11	0.448

How calculate these?

Another example



Another example



$P(B)$
.001

$P(E)$
.002

B	E	$P(A)$
t	t	.95
t	f	.94
f	t	.29
f	f	.001

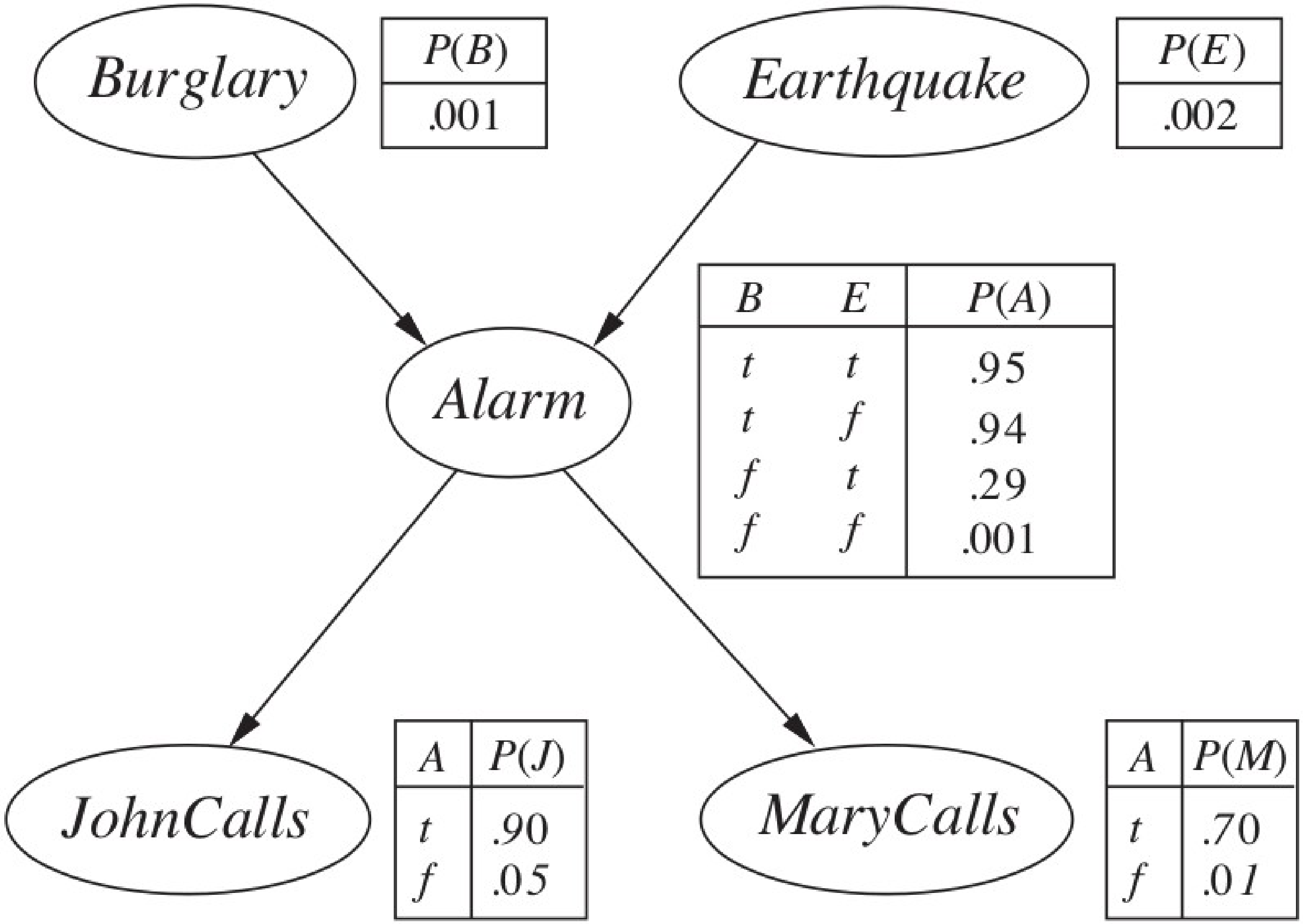
A	$P(J)$
t	.90
f	.05

A	$P(M)$
t	.70
f	.01

$$P(j, m, a, \neg b, \neg e) = \text{?}$$

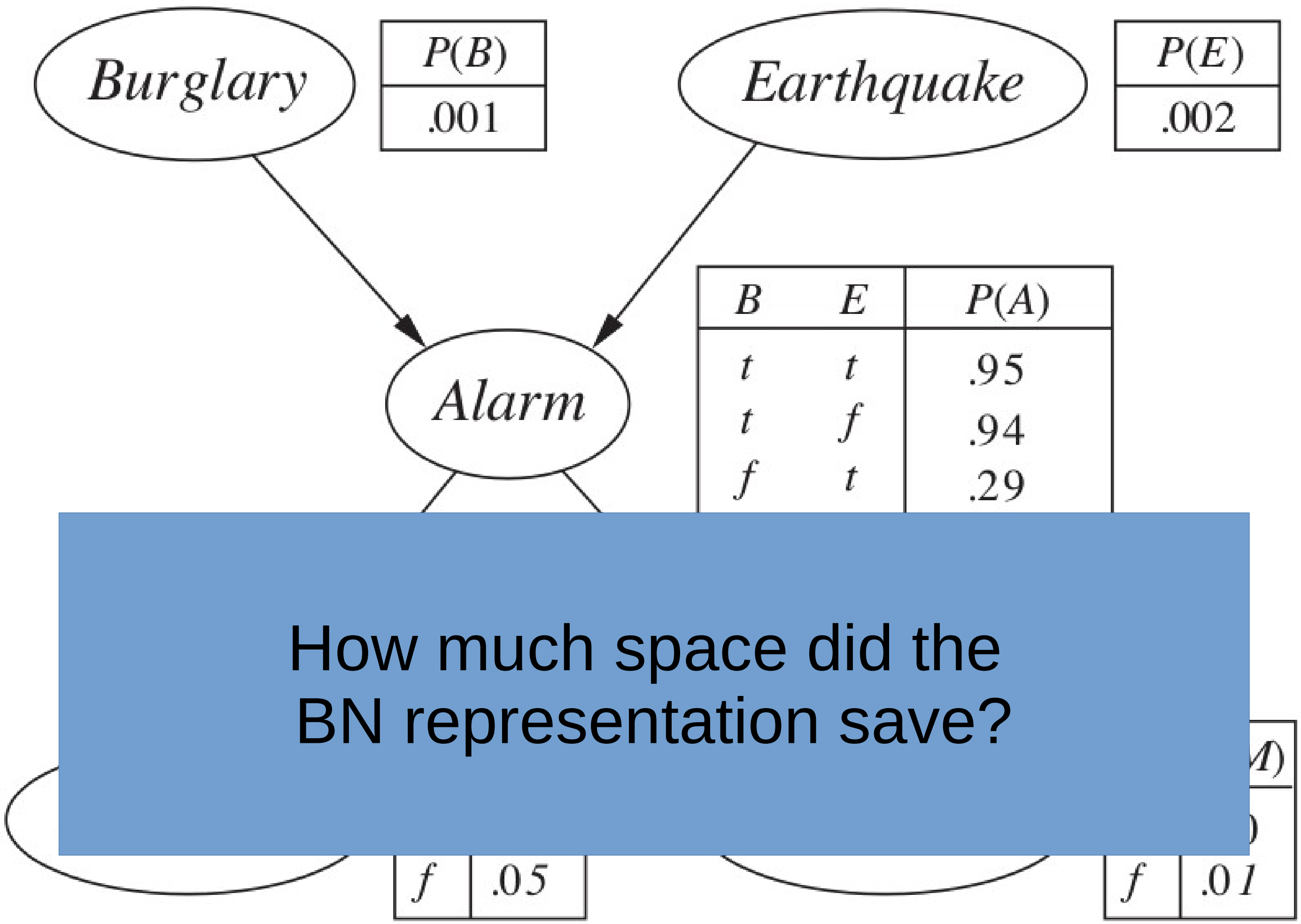
$$=$$

Another example



$$
\begin{aligned}
P(j, m, a, \neg b, \neg e) &= P(j \mid a) P(m \mid a) P(a \mid \neg b \wedge \neg e) P(\neg b) P(\neg e) \\
&= 0.90 \times 0.70 \times 0.001 \times 0.999 \times 0.998 = 0.000628
\end{aligned}
$$

Another example



How much space did the BN representation save?

$$P(j, m, a, \neg b, \neg e) = P(j \mid a) P(m \mid a) P(a \mid \neg b \wedge \neg e) P(\neg b) P(\neg e)$$
$$= 0.90 \times 0.70 \times 0.001 \times 0.999 \times 0.998 = 0.000628$$

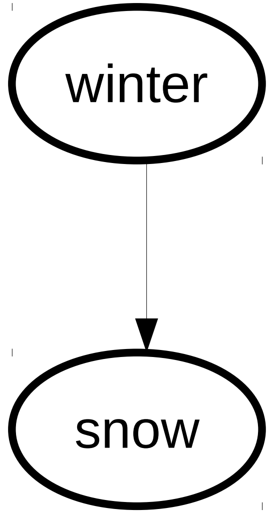
A simple example

Parameters of Bayes network

winter	P(S|W)
true	0.3
false	0.01

P(winter)=0.5

Structure of Bayes network



Joint distribution implied by bayes network

	winter	!winter
snow	0.15	0.005
!snow	0.35	0.495

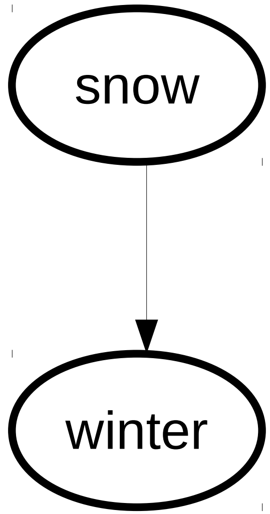
A simple example

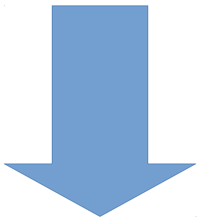
Parameters of Bayes network

snow	P(W|S)
true	0.968
false	0.414

P(snow)=0.155

Structure of Bayes network





Joint distribution implied by bayes network

	winter	!winter
snow	0.15	0.005
!snow	0.35	0.495

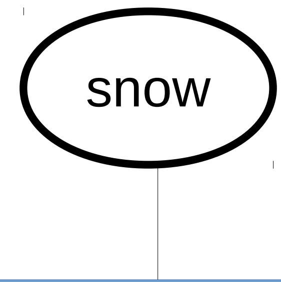
A simple example

Parameters of Bayes network

snow	P(W|S)
true	0.968
false	0.414

P(snow)=0.155

Structure of Bayes network



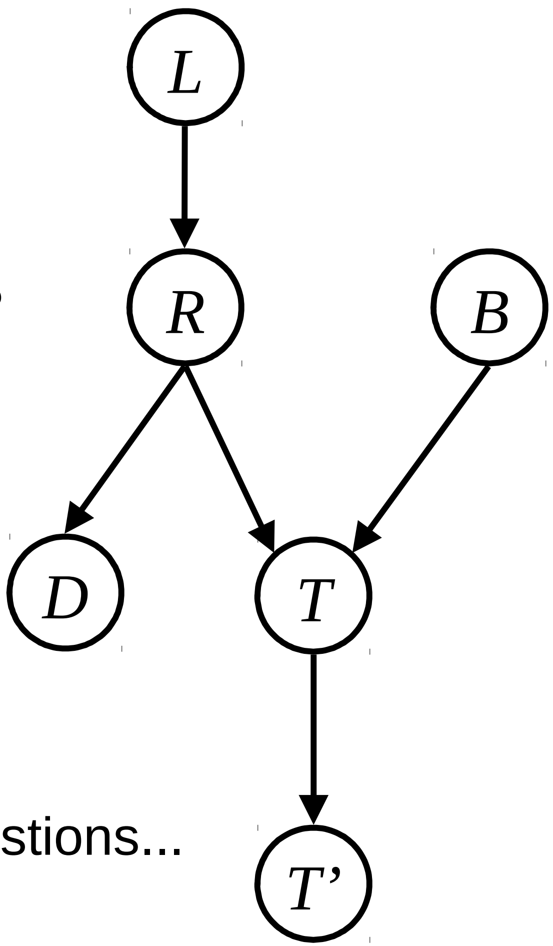
What does this say about causality
and bayes net semantics?
– what does bayes net topology encode?

Jo

	winter	!winter
snow	0.15	0.005
!snow	0.35	0.495

D-separation

What does bayes network structure imply about conditional independence among variables?



Are D and T independent?

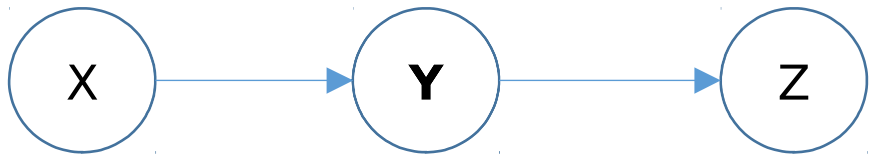
Are D and T conditionally independent given R?

Are D and T conditionally independent given L?

D-separation is a method of answering these questions...

D-separation

<u>Causal chain:</u>



Z is conditionally independent of X given Y
If Y is unknown, then Z is correlated with X

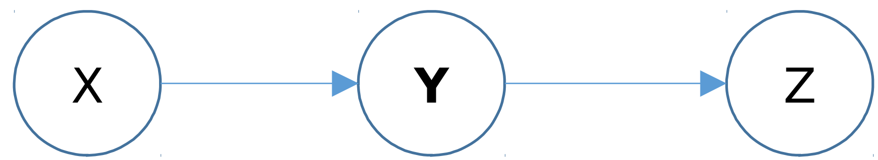
<u>For example:</u>
X = I was hungry
Y = I put pizza in the oven
Z = house caught fire

<u>Fire is conditionally independent of Hungry given Pizza...</u>

– Hungry and Fire are dependent if Pizza is unknown
– Hungry and Fire are independent if Pizza is known

D-separation

Causal chain:



Exercise: Prove it!

use caught fire

Fire is conditionally independent of Hungry given Pizza...

– Hungry and Fire are dependent if Pizza is unknown

– Hungry and Fire are independent if Pizza is known

D-separation

C

Exercise: Prove it!

$$
\begin{aligned}
P(z|x,y) &= \frac{P(x,y,z)}{P(x,y)} \\
&= \frac{P(z|y)P(y|x)P(x)}{P(y|x)P(x)} \\
&= P(z|y)
\end{aligned}
$$

ouse caught fire

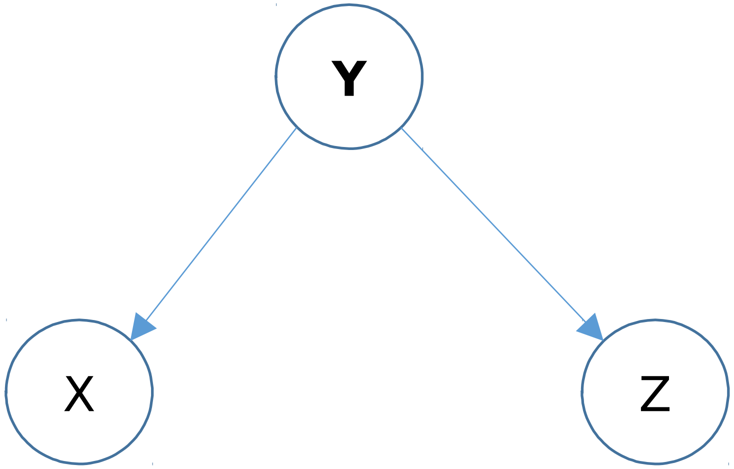
<u>Fire is conditionally independent of Hungry given Pizza...</u>

– Hungry and Fire are dependent if Pizza is unknown

– Hungry and Fire are independent if Pizza is known

D-separation

Common cause:



Z is conditionally independent of X given Y.
If Y is unknown, then Z is correlated with X

For example:
X = john calls
Y = alarm
Z = mary calls

D-separation

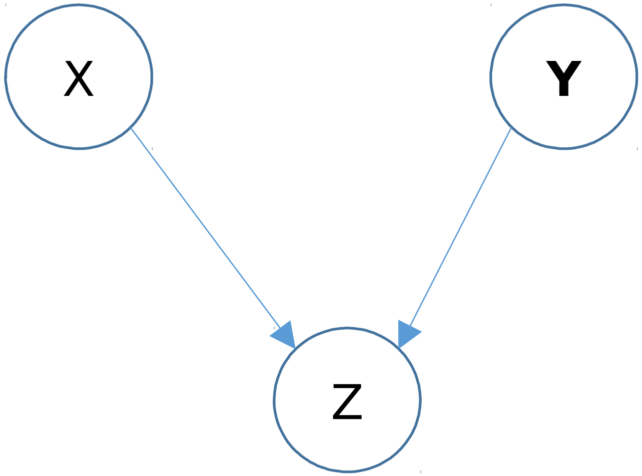
C

Exercise: Prove it!

$$
\begin{aligned}
P(z|x,y) &= \frac{P(x,y,z)}{P(x,y)} \\
&= \frac{P(z|y)P(x|y)P(y)}{P(x|y)P(y)} \\
&= P(z|y)
\end{aligned}
$$

D-separation

Common effect:



If Z is unknown, then X, Y are independent
If Z is known, then X, Y are correlated

For example:
X = burglary
Y = earthquake
Z = alarm

D-separation

Given an arbitrary Bayes Net, you can find out whether two variables are independent just by looking at the graph.

D-separation

Given an arbitrary Bayes Net, you can find out whether two variables are independent just by looking at the graph.

How?

D-separation

Given an arbitrary Bayes Net, you can find out whether two variables are independent just by looking at the graph.

Are X, Y independent given A, B, C?

1. enumerate all paths between X and Y

2. figure out whether any of these paths are active

3. if no active path, then X and Y are independent

D-separation

What's an active path?

Are X, Y independent given A, B,

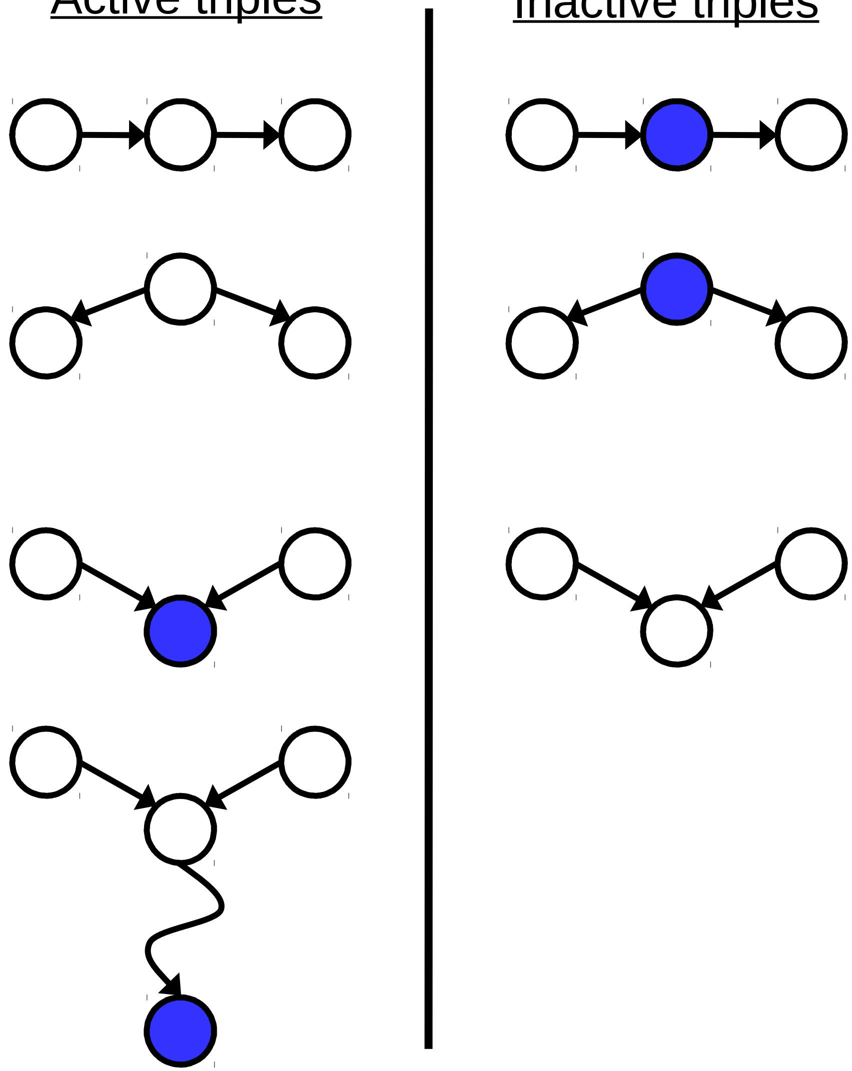
1. enumerate all paths between X and

2. figure out whether any of these paths are active

3. if no active path, then X and Y are independent

Active path

Active triples | Inactive triples



Any path that has an inactive triple on it is inactive

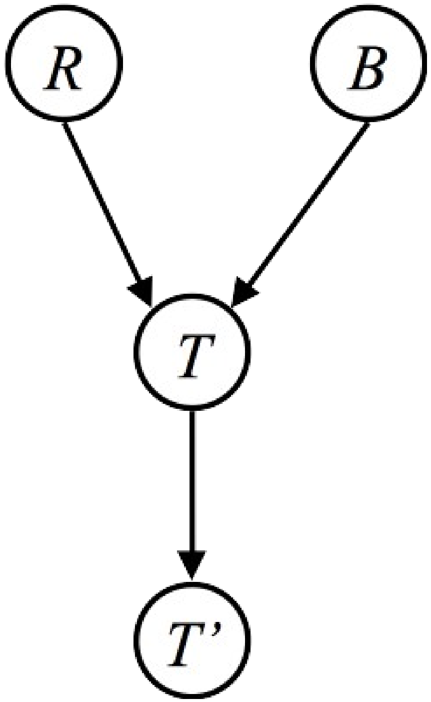
If a path has only active triples, then it is active

Example

$$R \perp\!\!\!\perp B$$

$$R \perp\!\!\!\perp B | T$$

$$R \perp\!\!\!\perp B | T'$$



Example

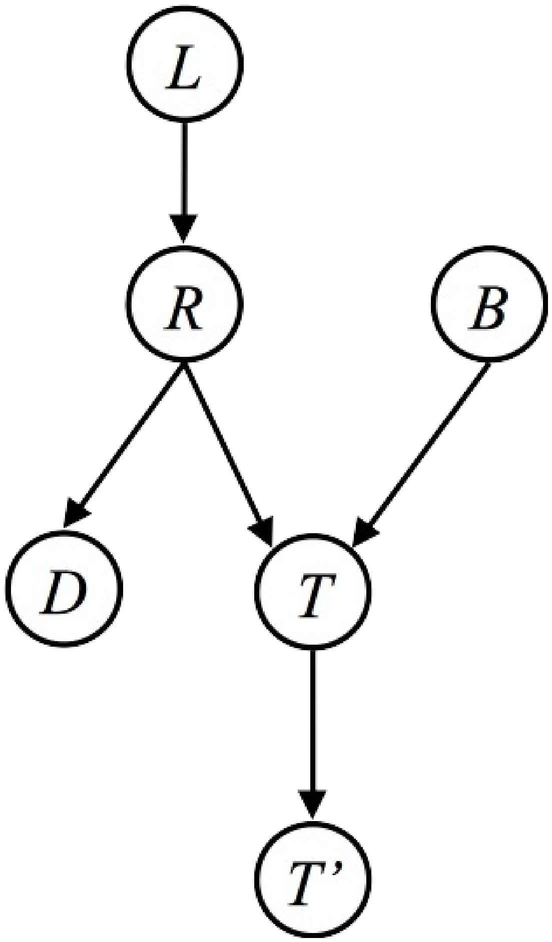
$$L \perp\!\!\!\perp T' | T$$

$$L \perp\!\!\!\perp B$$

$$L \perp\!\!\!\perp B | T$$

$$L \perp\!\!\!\perp B | T'$$

$$L \perp\!\!\!\perp B | T, R$$

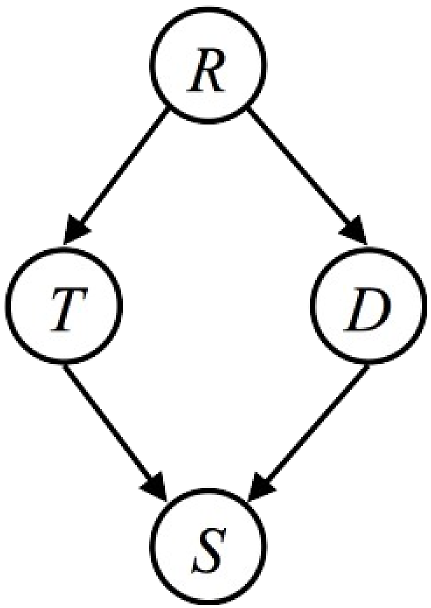


Example

$$T \perp\!\!\!\perp D$$

$$T \perp\!\!\!\perp D | R$$

$$T \perp\!\!\!\perp D | R, S$$



D-separation

<u>What Bayes Nets do:</u>

– constrain probability distributions that can be represented

– reduce the number of parameters

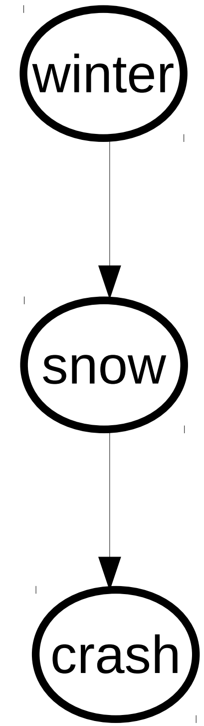
Constrained by conditional independencies induced by structure

– can figure out what these are by using d-separation

Is there a Bayes Net can represent any distribution?

Exact Inference

P(winter)=0.5



Given this
Bayes Network

winter	P(S|W)
true	0.3
false	0.01

snow	P(C|S)
true	0.1
false	0.01

Calculate P(C)

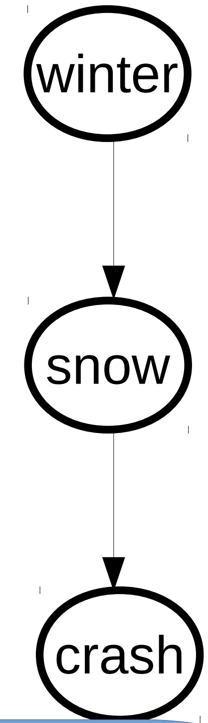
Calculate P(C|W)

Exact Inference

P(winter)=0.5

winter	P(S|W)
true	0.3
false	0.01

snow	P(C|S)
true	0.1
false	0.01



Given this
Bayes Network

Calculate P(C)

Calculate P(C|W)

Exact Inference:
– Can't read off answer from the CPTs.
– Must *infer* the answers.

Infer P(C) given P(C|S), P(S|W), P(W)

Infer P(C|W) given P(C|S), P(S|W), P(W)

Exact Inference

Given this
Bayes Network

<u>Exact Inference</u>:
– Can't read off answer from the CPTs.
– Must *infer* the answers.

Infer P(C) given P(C|S), P(S|W), P(W)

Infer P(C|W) given P(C|S), P(S|W), P(W)

Calculate P(C)

$$P(C) = \sum_w \sum_s P(C|s)P(s|w)p(w)$$

Calculate P(C|W)

$$P(C|W) = \frac{\sum_s P(C|s)P(s|W)p(W)}{P(W)}$$

Inference by enumeration

How exactly calculate this? $P(C) = \sum_w \sum_s P(C|s)P(s|w)p(w)$

<u>Inference by enumeration:</u>

1. calculate joint distribution
2. marginalize out variables we don't care about.

Inference by enumeration

How exactly calculate this? $P(C) = \sum_{w} \sum_{s} P(C|s)P(s|w)p(w)$

Inference by enumeration:

1. calculate joint distribution
2. marginalize out variables we don't care about.

P(winter)=0.5

winter	P(S|W)
true	0.3
false	0.1

snow	P(C|S)
true	0.1
false	0.01

Joint distribution

winter	snow	P(c,s,w)
true	true	0.015
false	true	0.005
true	false	0.0035
false	false	0.0045

Inference by enumeration

How exactly calculate this? $P(C) = \sum_w \sum_s P(C|s)P(s|w)p(w)$

Inference by enumeration:

1. calculate joint distribution
2. marginalize out variables we don't care about.

Joint distribution

winter	snow	P(c,s,w)
true	true	0.015
false	true	0.005
true	false	0.0035
false	false	0.0045

P(C) = 0.015+0.005+0.0035+0.0045

= 0.028

Inference by enumeration

How e (w)

Inferen

1. calc
2. marg

Pros/cons?

Pro: it works

Con: you must calculate the full joint distribution first
– what's wrong w/ that???

winter		
true	true	0.015
false	true	0.005
true	false	0.0035
false	false	0.0045

P(C) = 0.015+0.005+0.0035+0.0045

= 0.028

Enumeration vs variable elimination

Enumeration

$$P(C) = \sum_w \sum_s P(C|s)P(s|w)p(w)$$

Join on w

Join on s

Eliminate s

Eliminate w

Variable elimination

$$P(C) = \sum_s P(C|s) \sum_w P(s|w)p(w)$$

Join on w

Eliminate w

Join on s

Eliminate s

Variable elimination marginalizes early
– why does this help?

Variable elimination

$$P(C) = \sum_s P(C|s) \sum_w P(s|w)p(w)$$

P(winter)=0.5

winter	P(s|W)
true	0.3
false	0.1

Join on W

winter	P(s,W)
true	0.15
false	0.05

Sum out W

P(snow)=0.2

P(snow)=0.2

snow	P(c|S)
true	0.1
false	0.01

Join on S

snow	P(c,S)
true	0.02
false	0.008

Sum out S

P(crash)=0.08

⋮

Variable elimination

$$P(C) = \sum P(C|s) \sum P(s|w)p(w)$$

P(wi

winter
true
false

)=0.2

How does this change if we are given evidence?
– i.e. suppose we are know that it is winter time?

P(snow)=0.2

snow	P(c|S)
true	0.1
false	0.01

Join on S

snow	P(c,S)
true	0.02
false	0.008

Sum out S

P(crash)=0.08

Variable elimination w/ evidence

$$P(C|w) = \eta \sum_{s} P(C|s)P(s|w)p(w)$$

P(winter)=0.5

winter	P(s|w)
true	0.3
~~false~~	~~0.1~~

Select +w

$$P(s,w) = P(s|w)p(w)$$

snow	P(s,w)
true	0.15
false	0.35

snow	P(c|S)
true	0.1
false	0.01

Join on S

$$P(c,s,w) = P(c|s)P(s,w)$$

snow	P(c,S,w)
true	0.015
false	0.0035

snow	P(!c,S,w)
true	0.135
false	0.3465

Sum out S

P(c,w)=0.0185

Sum out S

P(!c,w)=0.4815

Normalize

P(c|w)=0.037

P(!c|w)=0.963

Variable elimination: general procedure

```
Variable elimination:

Given: evidence variables, e_1, …, e_m; variable to infer, Q
Given: all CPTs (i.e. factors) in the graph
Calculate: P(Q|e_1, dots, e_m)

1. select factors for the given evidence
2. select ordering of “hidden” variables: vars = {v_1, …, n_n}
3. for i = 1 to n
4.   join on v_i
5.   marginalize out v_i
6. join on query variable
7. normalize on query: P(Q|e_1, dots, e_m)
```

Variable elimination: general procedure

winter	P(s|W)
true	0.3
false	0.1

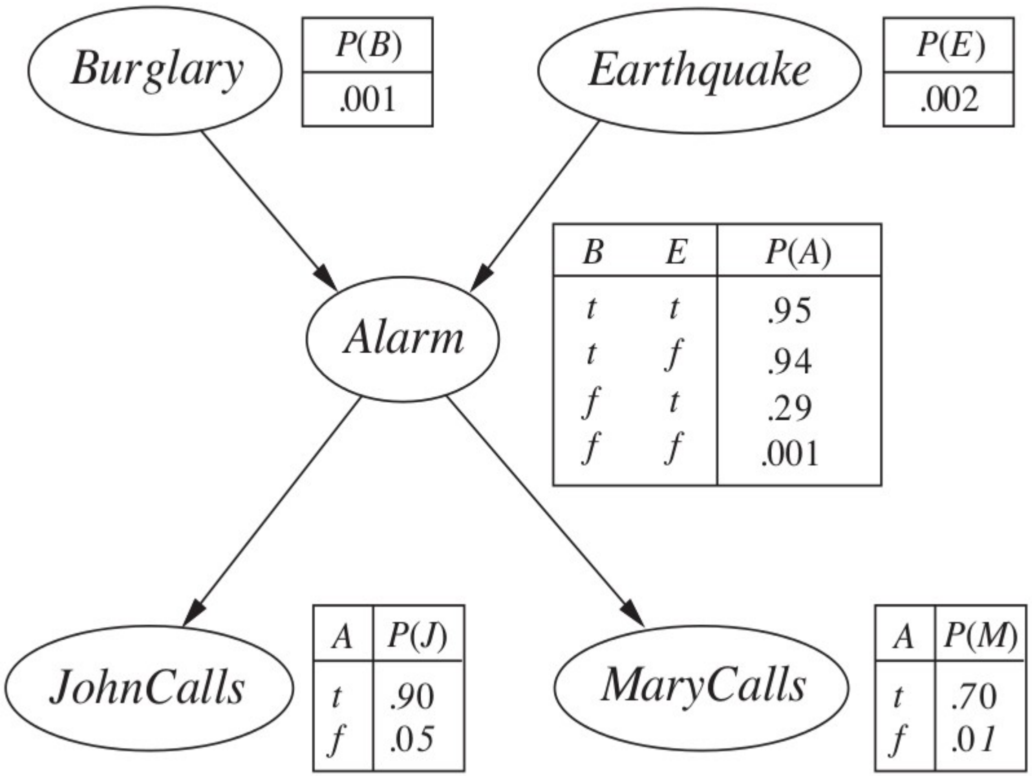
```
Variable elimination:

Given: evidence variables, e_1, …, e_m; variable to infer, Q
Given: all CPTs (i.e. factors) in the graph
Calculate: P(Q|e_1, dots, e_m)

1. select factors for the given evidence
2. select ordering of “hidden” variables: vars = {v_1, …, n_n}
3. for i = 1 to n
4.   join on v_i
5.   marginalize out v_i
6. join on query variable
7. normalize on query: P(Q|e_1, dots, e_m)
```

i.e. not query or evidence

– What are the evidence variables in the winter/snow/crash example?
– What are hidden variables? Query variables?

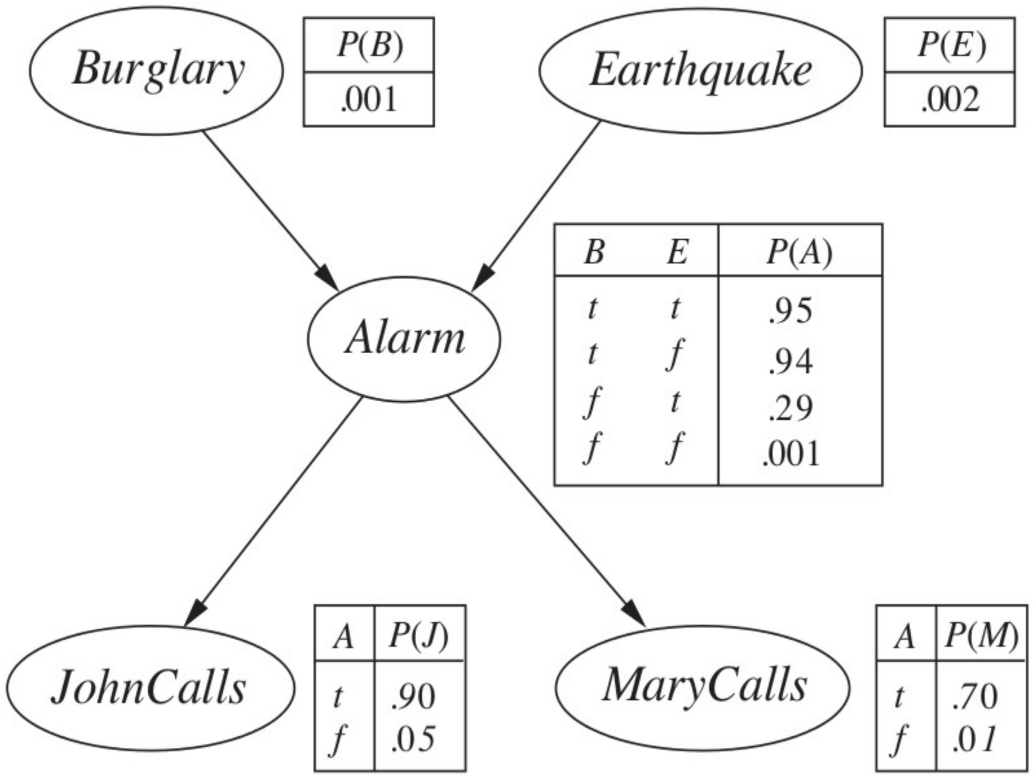
Variable elimination: general procedure example

P(b|m,j) = ?



Variable elimination: general procedure example

P(b|m,j) = ?

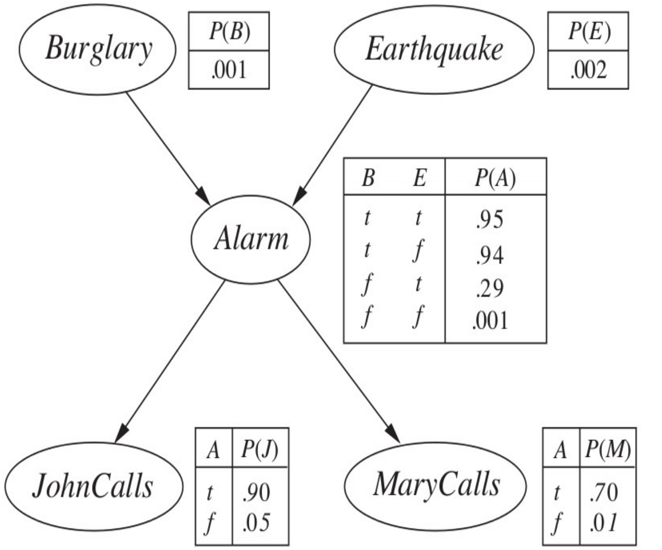


1. select evidence variables
   – P(m|A) P(j|A)
2. select variable ordering: A,E
3. join on A
   – P(m,j,A|B,E) = P(m|A) P(j|A) P(A|B,E)
4. marginalize out A
   – P(m,j|B,E) = \sum_A P(m,j,A|B,E)
5. join on E
   – P(m,j,E|B) = P(m,j|B,E) P(E)
6. marginalize out E
   – P(m,j|B) = \sum_E P(m,j,E|B)
7. join on B
   – P(m,j,B) = P(m,j|B)P(B)
8. normalize on B
   – P(B|m,j)

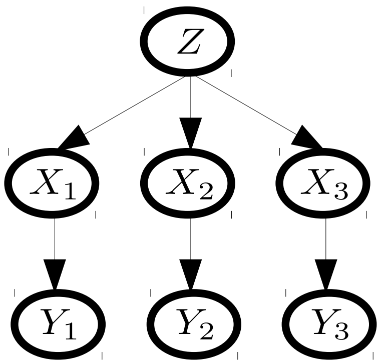
Variable elimination: general procedure example

Same example with equations: P(b|m,j) = ?

$$
\begin{aligned}
P(B|j,m) &\propto P(B,j,m) \\
&= \sum_{e,a} P(B,j,m,e,a) \\
&= \sum_{e,a} P(B)P(e)P(a|B,e)P(j|a)P(m|a) \\
&= \sum_{e} P(B)P(e) \sum_{a} P(a|B,e)P(j|a)P(m|a) \\
&= \sum_{e} P(B)P(e) f_1(B,e,j,m) \\
&= P(B) \sum_{e} P(e) f_1(B,e,j,m) \\
&= P(B) f_2(B,j,m)
\end{aligned}
$$



Another example



Calculate P(X_3|y_1,y_2,y_3)
Use this variable ordering: X_1, X_2, Z

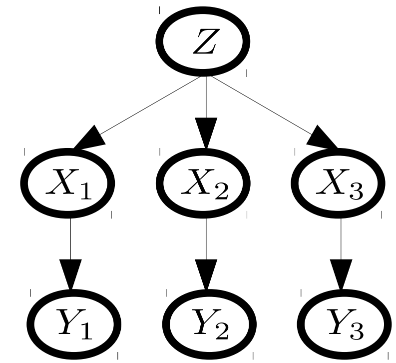
$$P(X_3|y_1,y_2,y_3) = \underbrace{\sum_Z P(Z) \underbrace{\sum_{X_1} P(X_1|Z)P(y_1|X_1)}_{P(y_1|Z)} \underbrace{\sum_{X_2} P(X_2|Z)P(y_2|X_2)}_{P(y_2|Z)} P(X_3|Z)P(y_3|X_3)}_{P(y_1,y_2,y_3,X_3)}$$

$$P(y_1, y_2, X_3)$$

normalize

$$P(X_3|y_1,y_2,y_3)$$

Another example



Calculate P(X_3|y_1,y_2,y_3)
Use this variable ordering: X_1, X_2, Z

$$P(X_3|y_1,y_2,y_3) = \sum_Z P(Z) \sum_{X_1} P(X_1|Z)P(y_1|X_1) \sum_{X_2} P(X_2|Z)P(y_2|X_2)P(X_3|Z)P(y_3|X_3)$$

$$P(y_1|Z)$$

$$P(y_2|Z)$$

$$P(y_1,y_2,X_3)$$

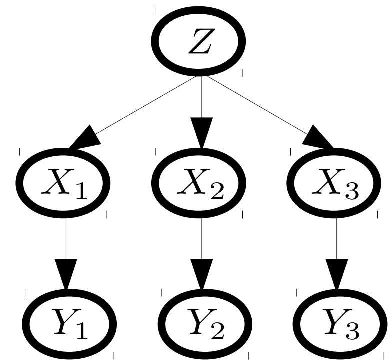
$$P(y_1,y_2,y_3,X_3)$$

normalize

$$P(X_3|y_1,y_2,y_3)$$

What would this look like if we used a different ordering: Z, X_1, X_2?
– why is ordering important?

Another example



Calculate P(X_3|y_1,y_2,y_3)
Use this variable ordering: X_1, X_2, Z

Ordering has a major impact on size of largest factor

– size 2^n vs size 2
– an ordering w/ small factors might not exist for a given network
– in worst case, inference is np-hard in the number of variables
  – an efficient solution to inference would produce efficent sol'ns to 3SAT

normalize

$$P(X_3|y_1, y_2, y_3)$$

What would this look like if we used a different ordering: Z, X_1, X_2?
– why is ordering important?

Polytrees

Polytree:

- bayes net w/ no undirected cycles
- inference is simpler than the general case (why)?
  - what is maximum factor size?
  - what is the complexity of inference?

Can you do cutset conditioning?

Approximate Inference

Can't do exact inference in all situations (because of complexity)

Alternatives?

Approximate Inference

Can't do exact inference in all situations (because of complexity)

Alternatives?

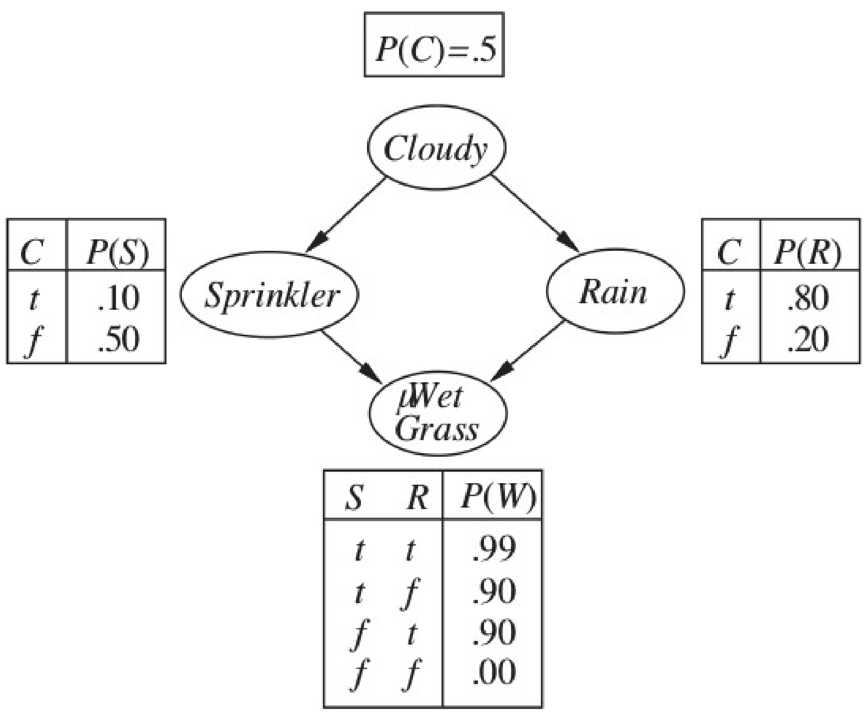
Yes: approximate inference

Basic idea: sample from the distribution and then evaluate distribution of interest

Direct Sampling/Rejection Sampling

Calculate P(Q|e_1,...,e_n)

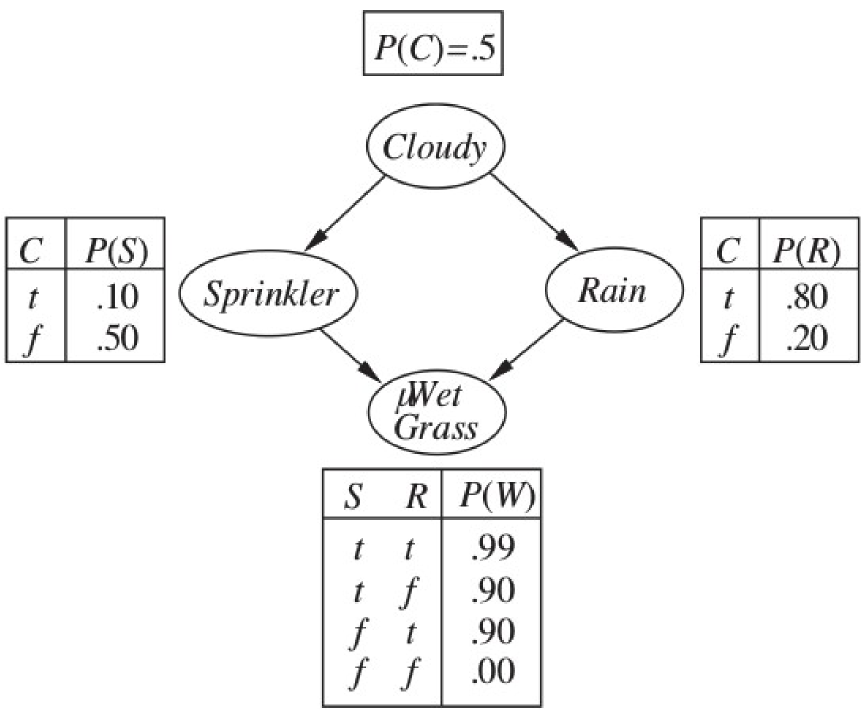
1. sort variables in topological order (partial order)
2. starting with root, draw one sample for each variable, X_i, from P(X_i|parents(X_i))
3. repeat step 2 n times and save the results
4. induce distribution of interest from samples



Direct Sampling/Rejection Sampling

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. starting with root, draw one sample for each variable, X_i, from P(X_i|parents(X_i))
3. repeat step 2 n times and save the results
4. induce distribution of interest from samples

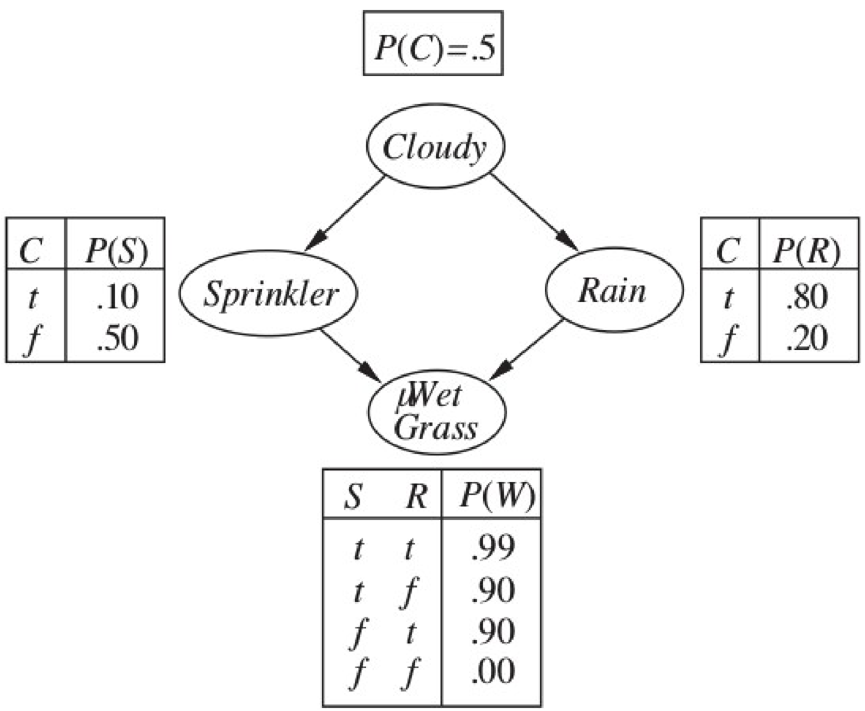


Topological sort: C,S,R,W

Direct Sampling/Rejection Sampling

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. starting with root, draw one sample for each variable, X_i, from P(X_i|parents(X_i))
3. repeat step 2 n times and save the results
4. induce distribution of interest from samples



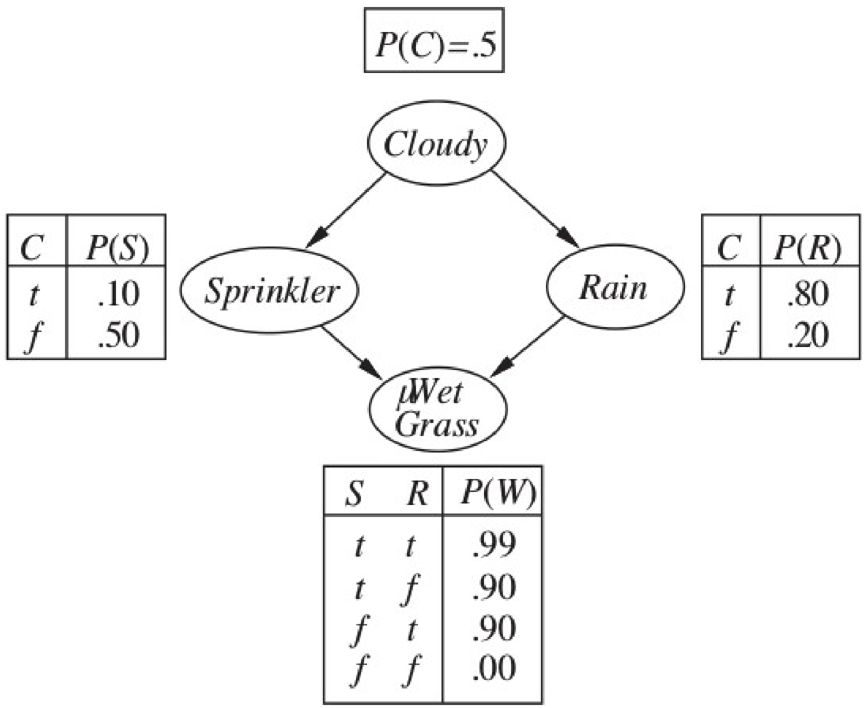
Topological sort: C,S,R,W

```
C, S, R, W
```

Direct Sampling/Rejection Sampling

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. starting with root, draw one sample for each variable, X_i, from P(X_i|parents(X_i))
3. repeat step 2 n times and save the results
4. induce distribution of interest from samples



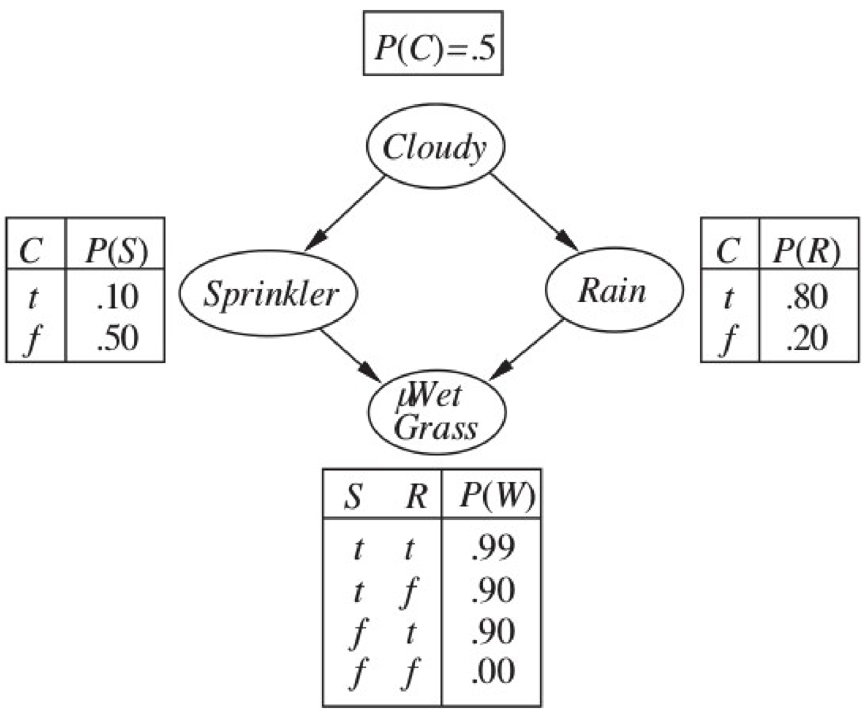
Topological sort: C,S,R,W

```
C, S, R, W
1
```

Direct Sampling/Rejection Sampling

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. starting with root, draw one sample for each variable, X_i, from P(X_i|parents(X_i))
3. repeat step 2 n times and save the results
4. induce distribution of interest from samples



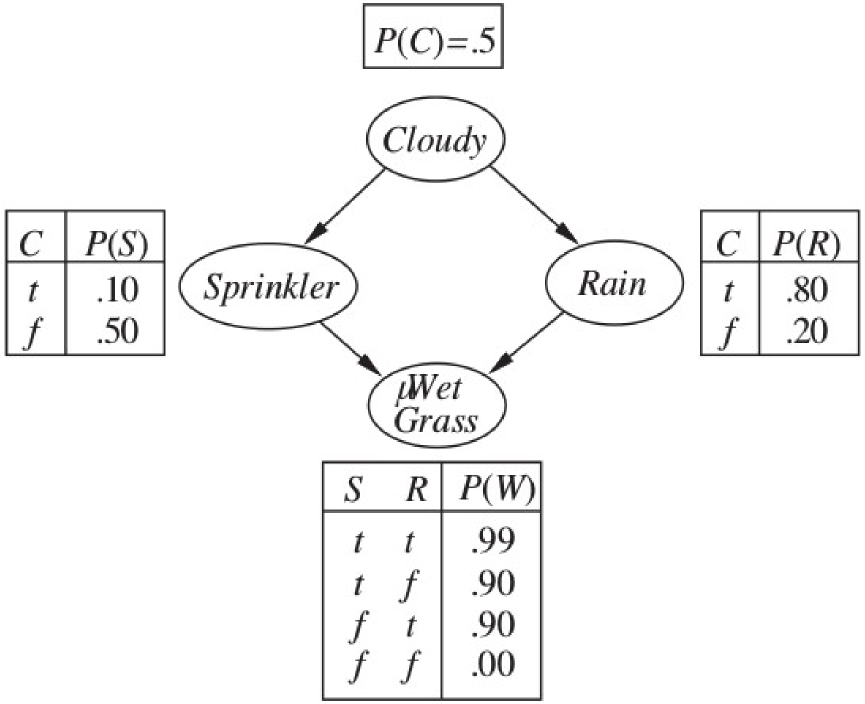
Topological sort: C,S,R,W

```
C, S, R, W
1, 1
```

Direct Sampling/Rejection Sampling

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. starting with root, draw one sample for each variable, X_i, from P(X_i|parents(X_i))
3. repeat step 2 n times and save the results
4. induce distribution of interest from samples



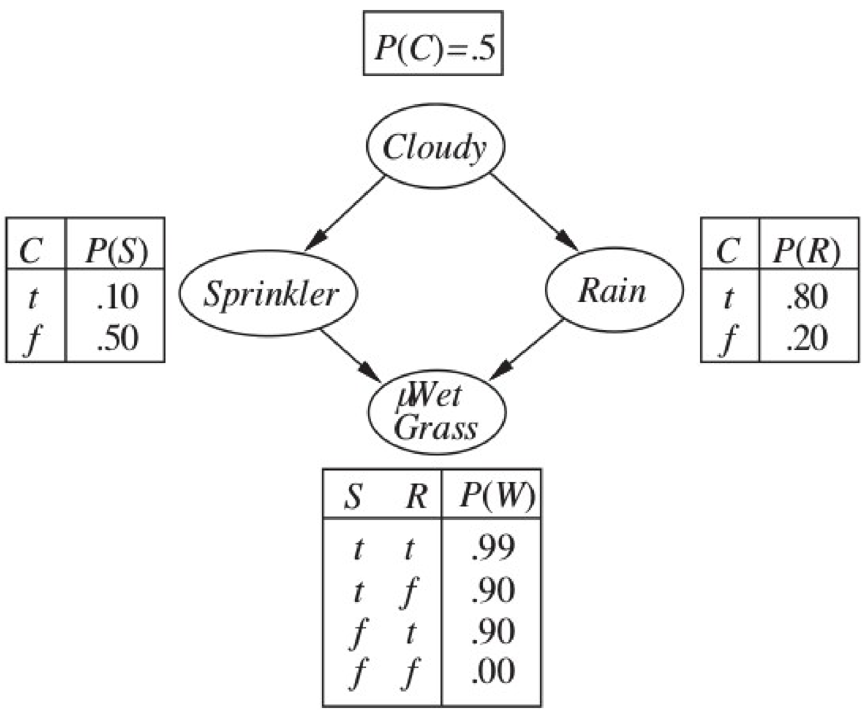
Topological sort: C,S,R,W

```
C, S, R, W
1, 1, 0
```

Direct Sampling/Rejection Sampling

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. starting with root, draw one sample for each variable, X_i, from P(X_i|parents(X_i))
3. repeat step 2 n times and save the results
4. induce distribution of interest from samples



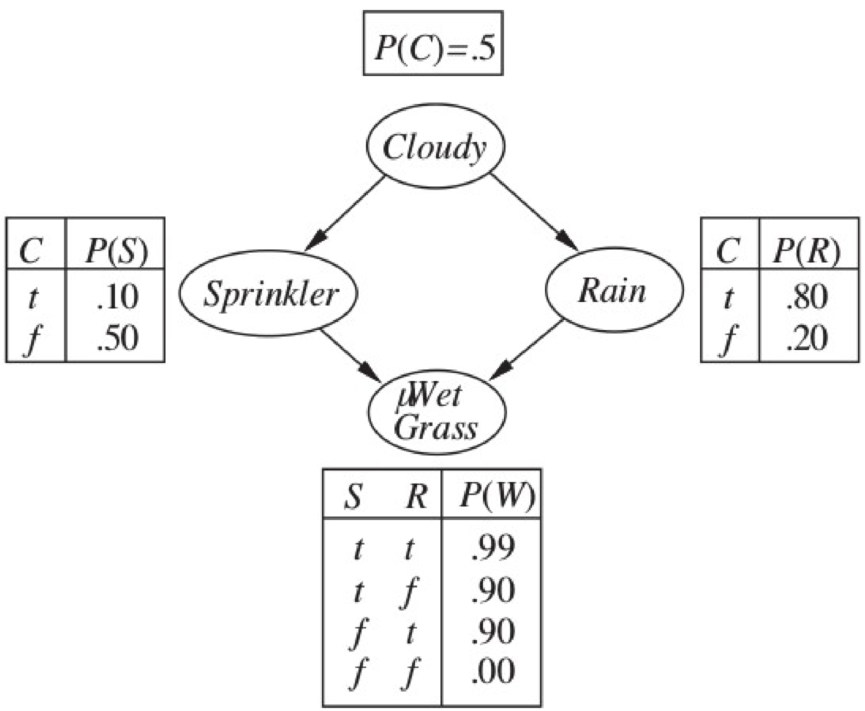
Topological sort: C,S,R,W

```
C, S, R, W
1, 1, 0, 1
```

Direct Sampling/Rejection Sampling

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. starting with root, draw one sample for each variable, X_i, from P(X_i|parents(X_i))
3. repeat step 2 n times and save the results
4. induce distribution of interest from samples



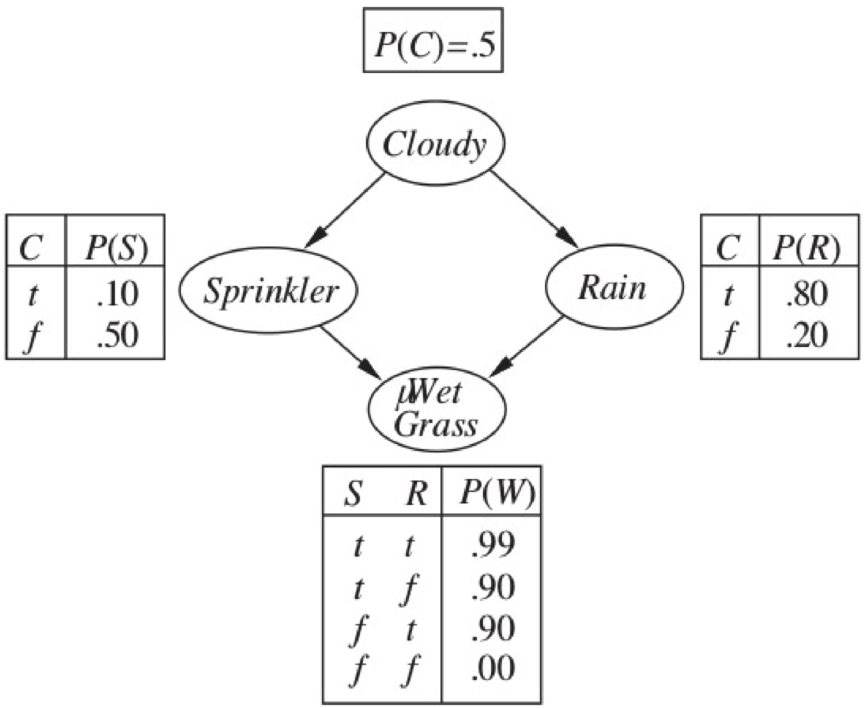
Topological sort: C,S,R,W

```
C, S, R, W
1, 1, 0, 1
1, 0, 1, 1
0, 1, 0, 1
1, 0, 1, 1
0, 0, 1, 1
...
```

Direct Sampling/Rejection Sampling

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. starting with root, draw one sample for each variable, X_i, from P(X_i|parents(X_i))
3. repeat step 2 n times and save the results
4. induce distribution of interest from samples



Topological sort: C,S,R,W

```
C, S, R, W
1, 1, 0, 1
1, 0, 1, 1
0, 1, 0, 1
1, 0, 1, 1
0, 0, 1, 1
...
```

P(W|C) = 3/3

P(R|S) = 0/2

P(W) = 5/5

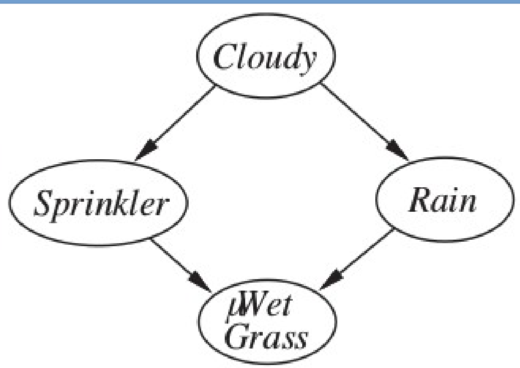
Direct Sampling/Rejection Sampling

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2 … _i))
3
4

What are the strengths/weakness of this approach?

Topological sort: C,S,R,W



C	P(S)
t	.10
f	.50

C	P(R)
t	.80
f	.20

S	R	P(W)
t	t	.99
t	f	.90
f	t	.90
f	f	.00

```
C, S, R, W
1, 1, 0, 1
1, 0, 1, 1
0, 1, 0, 1
1, 0, 1, 1
0, 0, 1, 1
...
```

P(W|C) = 3/3

P(R|S) = 0/2

P(W) = 5/5

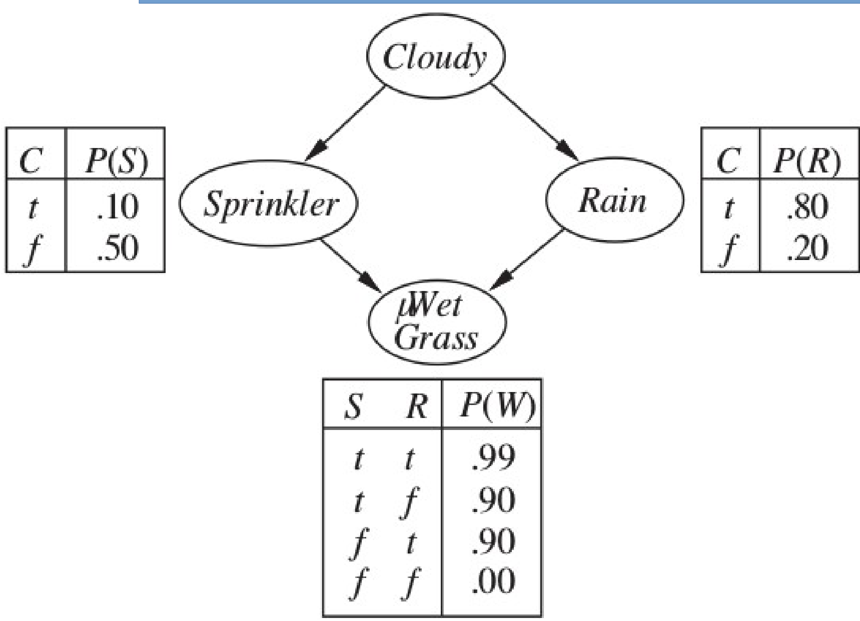
Direct Sampling/Rejection Sampling

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2 … _i))
3
4

What are the strengths/weakness of this approach?
– inference is easy
– estimates are consistent (what does that mean?)
– hard to get good estimates if evidence occurs rarely

Topological sort: C,S,R,W




```
C, S, R, W
1, 1, 0, 1
1, 0, 1, 1
0, 1, 0, 1
1, 0, 1, 1
0, 0, 1, 1
...
```

P(W|C) = 3/3

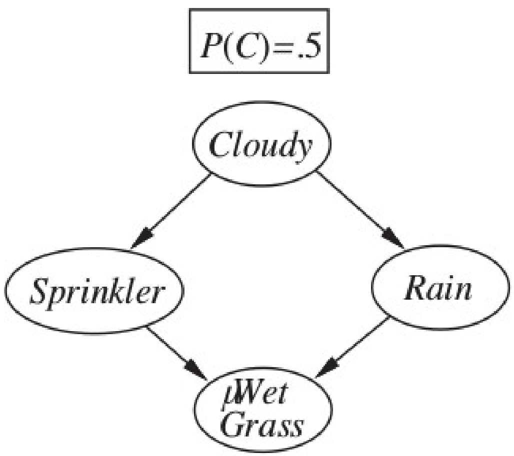
P(R|S) = 0/2

P(W) = 5/5

Likelihood weighting

What if the evidence is unlikely?
- use likelihood weighting!

<u>Idea</u>:
- only generate samples consistent w/ evidence
- but weight that samples according to likelihood of evidence in that scenario



C	P(S)
t	.10
f	.50

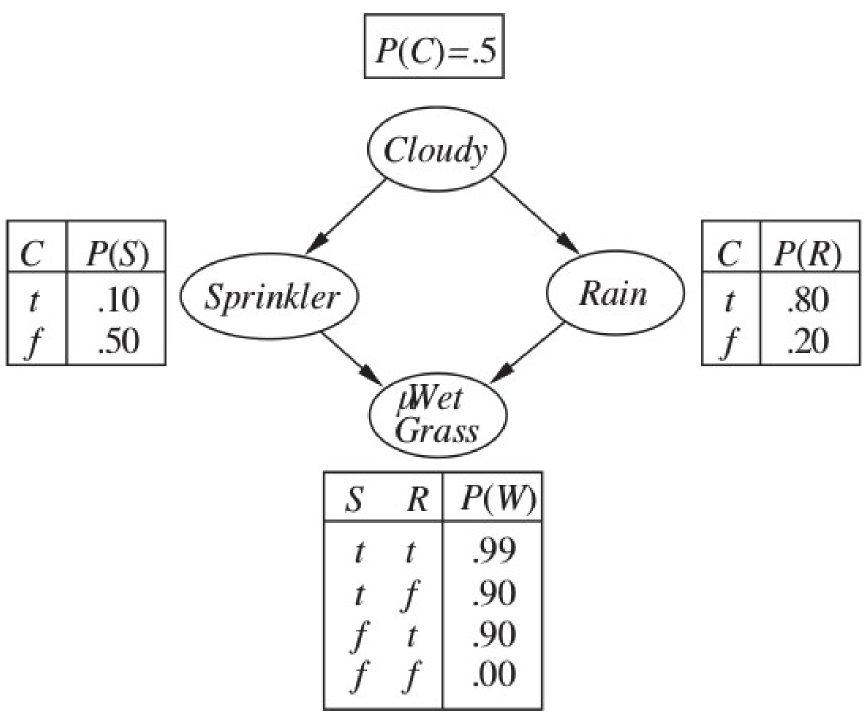
C	P(R)
t	.80
f	.20

S	R	P(W)
t	t	.99
t	f	.90
f	t	.90
f	f	.00

Likelihood weighting

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. init W = 1
3. set all evidence variables to their query values
4. starting with root, draw one sample for each non-evidence variable:
   X_i, from P(X_i|parents(X_i))
5. as you encounter the evidence variables, W=W*P(e|samples)
6. repeat steps 2--5 n times and save the results
7. induce distribution of interest from weighted samples



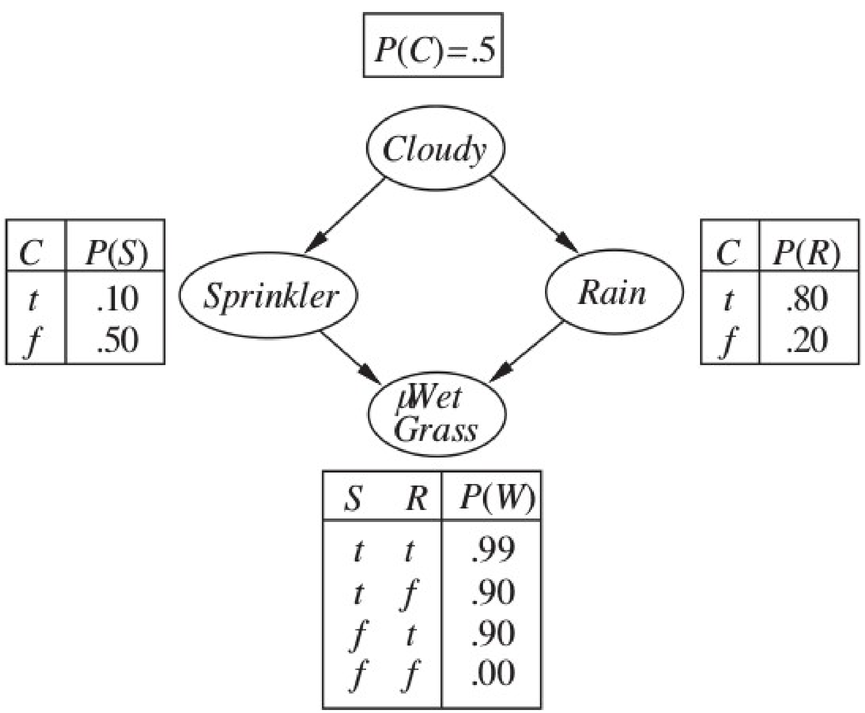
Calculate: P(S,R|c,w)

```
C, S, R, W, weight
                1
```

Likelihood weighting

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. init W = 1
3. set all evidence variables to their query values
4. starting with root, draw one sample for each non-evidence variable:
    X_i, from P(X_i|parents(X_i))
5. as you encounter the evidence variables, W=W*P(e|samples)
6. repeat steps 2--5 n times and save the results
7. induce distribution of interest from weighted samples



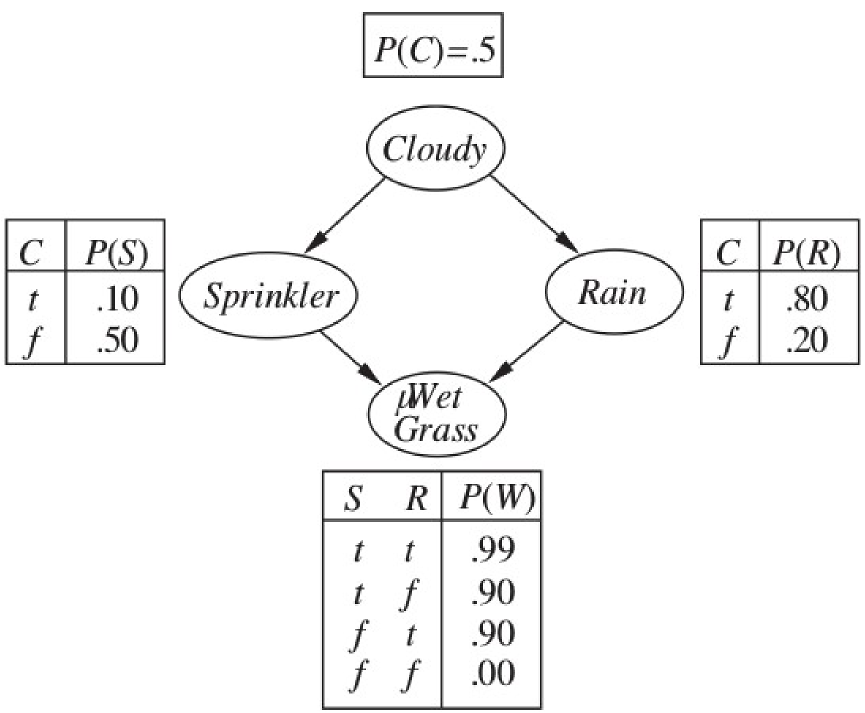
Calculate: P(S,R|c,w)

```
C, S, R, W, weight
1,           0.5
```

Likelihood weighting

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. init W = 1
3. set all evidence variables to their query values
4. starting with root, draw one sample for each non-evidence variable:
   X_i, from P(X_i|parents(X_i))
5. as you encounter the evidence variables, W=W*P(e|samples)
6. repeat steps 2--5 n times and save the results
7. induce distribution of interest from weighted samples



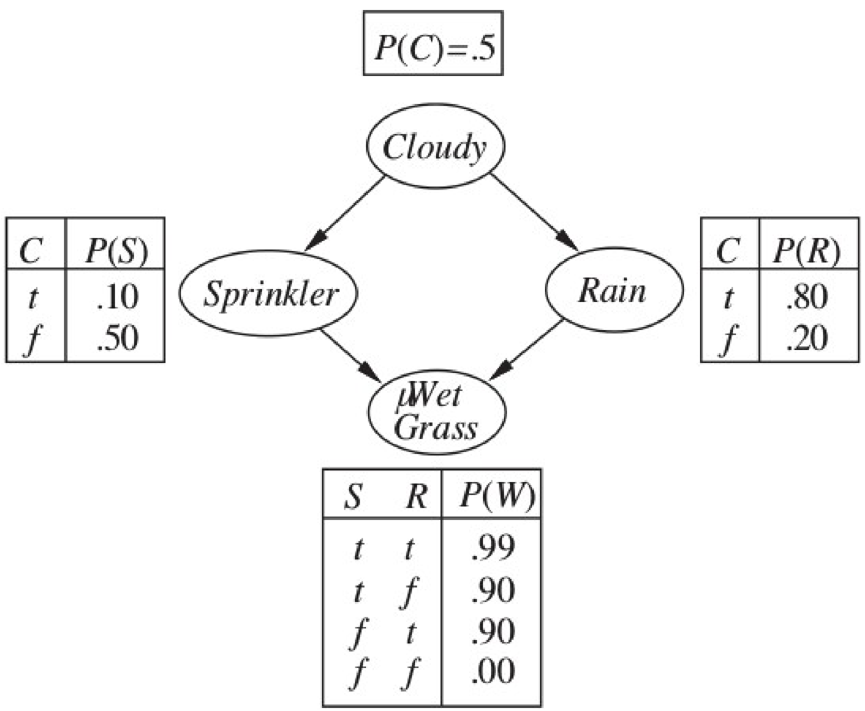
Calculate: P(S,R|c,w)

```
C, S, R, W, weight
1, 0,       0.5
```

Likelihood weighting

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. init W = 1
3. set all evidence variables to their query values
4. starting with root, draw one sample for each non-evidence variable:
   X_i, from P(X_i|parents(X_i))
5. as you encounter the evidence variables, W=W*P(e|samples)
6. repeat steps 2--5 n times and save the results
7. induce distribution of interest from weighted samples



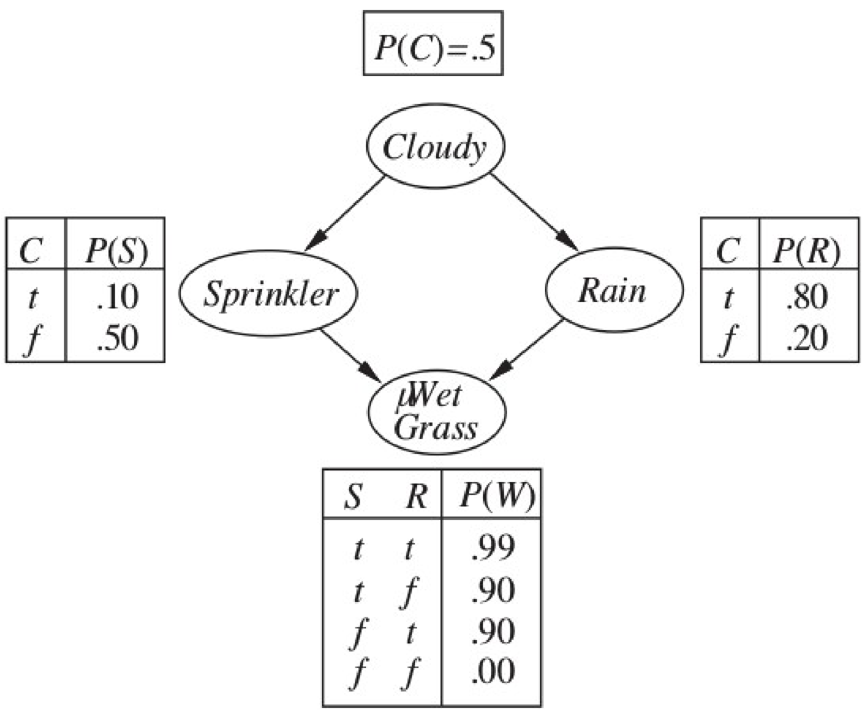
Calculate: P(S,R|c,w)

```
C, S, R, W, weight
1, 0, 1,    0.5
```

Likelihood weighting

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. init W = 1
3. set all evidence variables to their query values
4. starting with root, draw one sample for each non-evidence variable:
   X_i, from P(X_i|parents(X_i))
5. as you encounter the evidence variables, W=W*P(e|samples)
6. repeat steps 2--5 n times and save the results
7. induce distribution of interest from weighted samples



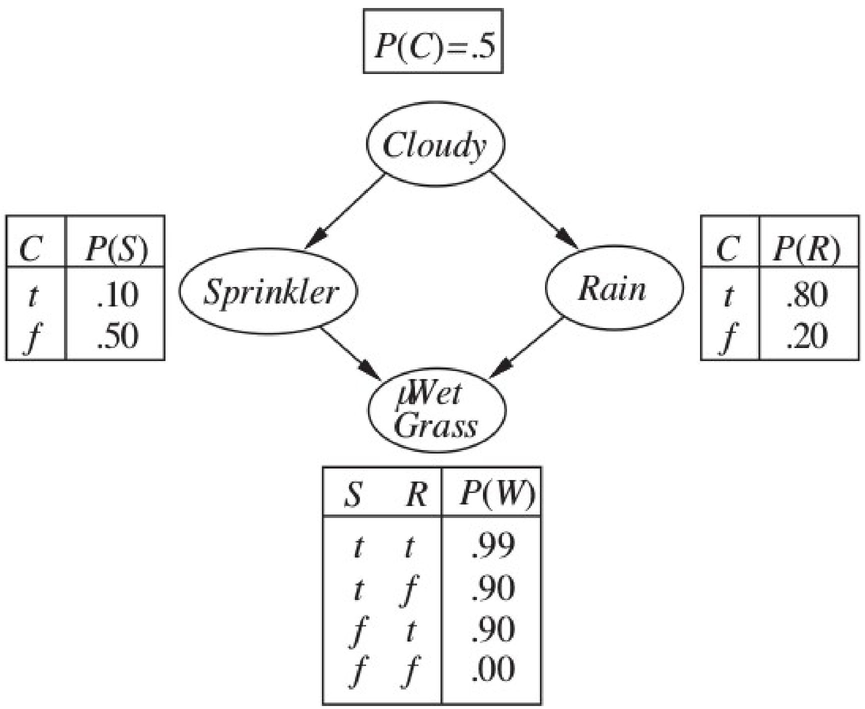
Calculate: P(S,R|c,w)

```
C, S, R, W, weight
1, 0, 1, 1, 0.45
```

Likelihood weighting

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. init W = 1
3. set all evidence variables to their query values
4. starting with root, draw one sample for each non-evidence variable:
   X_i, from P(X_i|parents(X_i))
5. as you encounter the evidence variables, W=W*P(e|samples)
6. repeat steps 2--5 n times and save the results
7. induce distribution of interest from weighted samples



Calculate: P(S,R|c,w)

```
C, S, R, W, weight
1, 0, 1, 1, 0.45
1, 1, 0, 1, 0.45
1, 1, 1, 1, 0.495
1, 0, 0, 1, 0
1, 0, 1, 1, 0.45
...
```

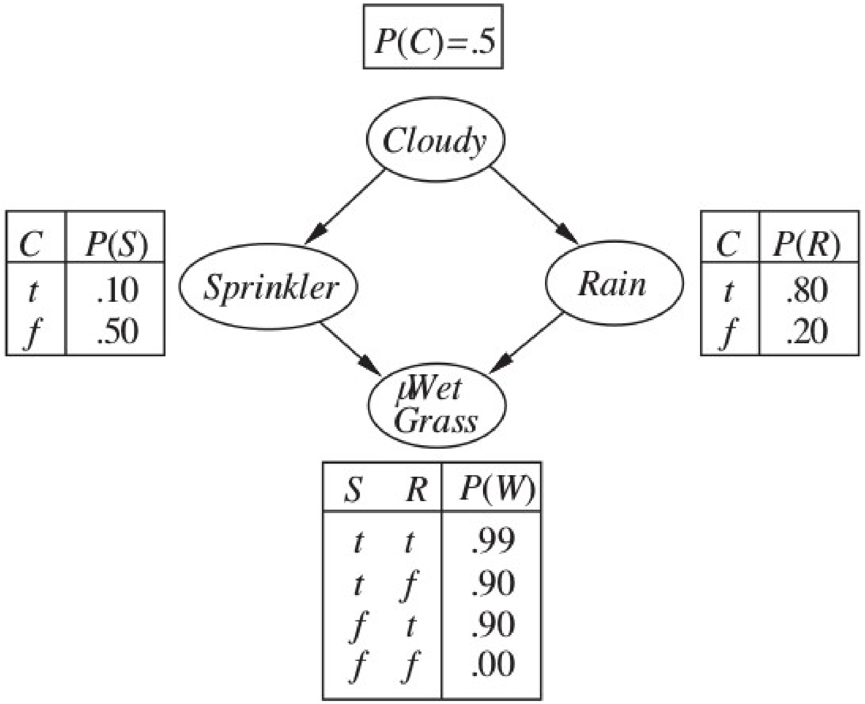
P(s|c,w) = 0.476/sum W

P(r|c,w) = 0.46/sum W

Likelihood weighting

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. init W = 1
3. set all evidence variables to their query values
4. starting with root, draw one sample for each non-evidence variable:
   X_i, from P(X_i|parents(X_i))
5. as you encounter the evidence variables, W=W*P(e|samples)
6. repeat steps 2--5 n times and save the results
7. induce distribution of interest from weighted samples



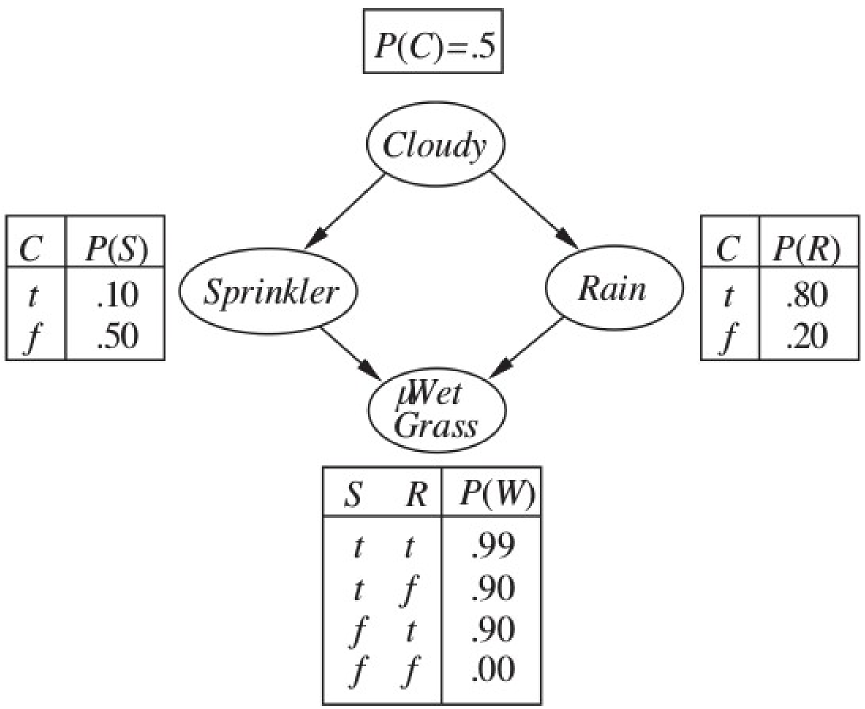
Calculate: P(S,R|c,w)

```
C, S, R, W, weight
1, 0, 1, 1, 0.45
1, 1, 0, 1, 0.45
1, 1, 1, 1, 0.495
1, 0, 0, 1, 0
1, 0, 1, 1, 0.45
...
```

Likelihood weighting

Calculate P(Q|e_1,...,e_n)

1. sort variables in topological order (partial order)
2. init W = 1
3. set all evidence variables to their query values
4. starting with root, draw one sample for each non-evidence variable:
   X_i, from P(X_i|parents(X_i))
5. as you encounter the evidence variables, W=W*P(e|samples)
6. repeat steps 2--5 n times and save the results
7. induce distribution of interest from weighted samples



Calculate: P(S,R|c,w)

```
C, S, R, W, weight
1, 0, 1, 1, 0.45
1, 1, 0, 1, 0.45
1, 1, 1, 1, 0.495
1, 0, 0, 1, 0
1, 0, 1, 1, 0.45
...
```

P(s|c,w) = 0.476/sum W

P(r|c,w) = 0.46/sum W

Bayes net example

cavity	P(T,C)	P(T,!C)	P(!T,C)	P(!T,!C)
true	0.16	0.018	0.018	0.002
false	0.048	0.19	0.11	0.448

Is there a way to represent this distribution more compactly?

Bayes net example

cavity	P(T,C)	P(T,!C)	P(!T,C)	P(!T,!C)
true	0.16	0.018	0.018	0.002
false	0.048	0.19	0.11	0.448

Is there a way to represent this distribution more compactly?

– does this diagram help?