COMS W4701: Artificial Intelligence

Lecture 8a: Bayesian Networks

Tony Dear, Ph.D.
Department of Computer Science
School of Engineering and Applied Sciences

Today

- Bayesian networks
- D-separation
- Inference by enumeration
- Factors and variable elimination

Probabilistic Graphical Models

- Probabilistic models can encode knowledge and associated uncertainty, including exceptions and special cases without full enumeration
- System aspects are captured by joint distributions over random variables

- A graphical model uses graphs to compactly encode a complex distribution
- It also represents *factorizations* that can be used to simplify the model

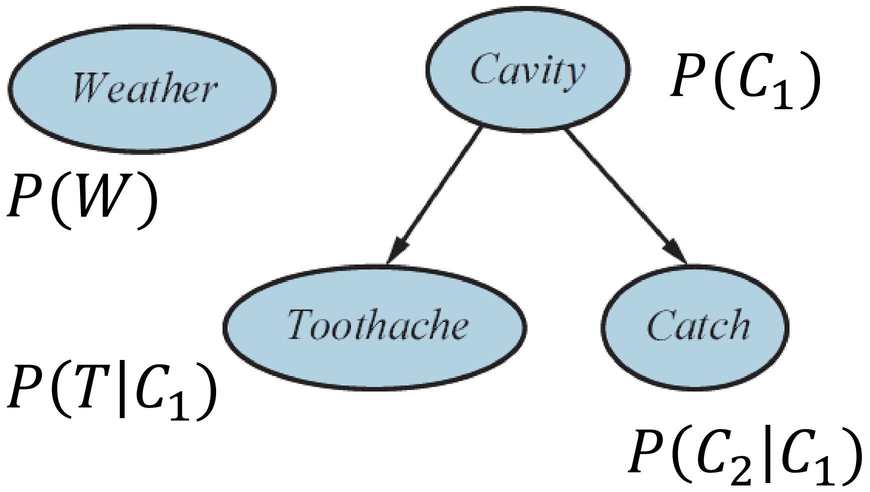
- Such models are more easily interpretable and transparent for users
- Are more amenable to *inference* and *learning* for model construction

Bayesian Networks

- A **Bayesian network** is a directed acyclic graph (DAG) representing a joint distribution
- Captures both a factorization as well as a set of conditional independences

- Each node corresponds to a random variable

- Each edge indicates influence or correlation
- May also be causation, but not always



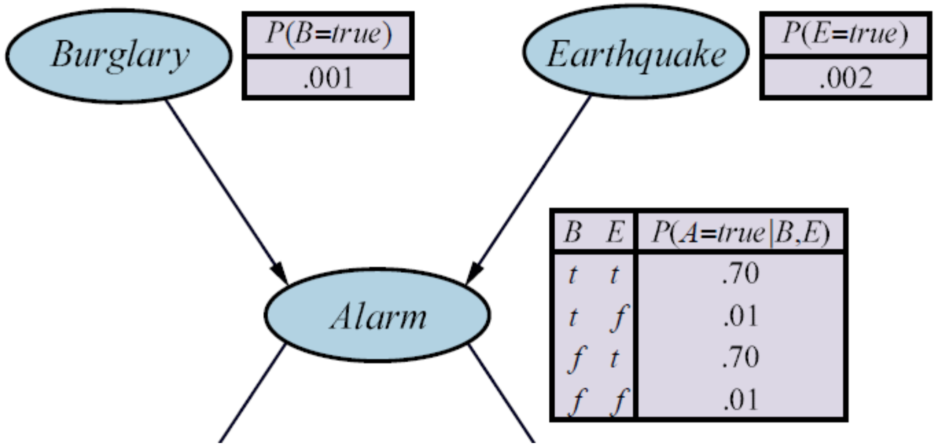
- **Parameters** of the Bayes net: A *local* conditional probability table (CPT) for each node
- The CPT for node X_i contains the values $P(X_i|parents(X_i))$

Conditional Probability Tables

- A CPT contains *all* possible conditional distributions $P(X_i|parents(X_i))$

- If each RV domain is size d and X_i has k parents, then there are d^k combinations of parent values, d^k different conditional distributions
- If X_i is also size k, then CPT has d^{k+1} parameters in total

- Optimization for CPTs of binary RVs:
- Can simply store half of the parameters



Joint Distribution

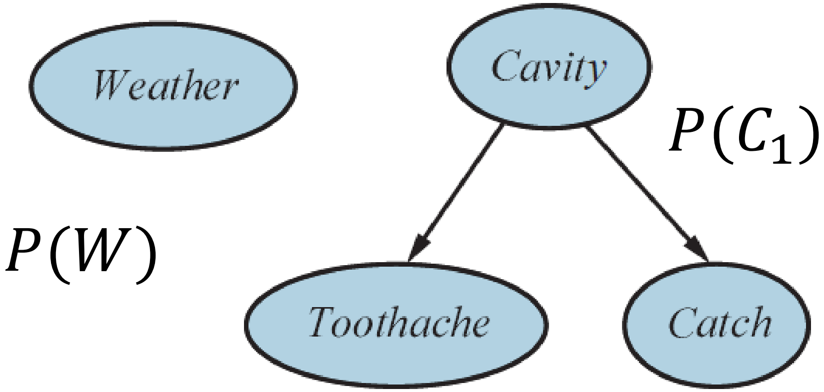
- *Assumption*: X_i is conditionally independent of its non-descendants given its parents

- Given a **topological ordering** of nodes $X_1, \dots, X_n$ s.t. all ancestors of a node occur before it, Bayes net joint probabilities are defined as follows:

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

- Example calculations:

$$\begin{aligned} P(w, c_1, t, c_2) &= P(w)P(c_1)P(t|c_1)P(c_2|c_1) \\ &= P(c_1)P(c_2|c_1)P(t|c_1)P(w) \end{aligned}$$



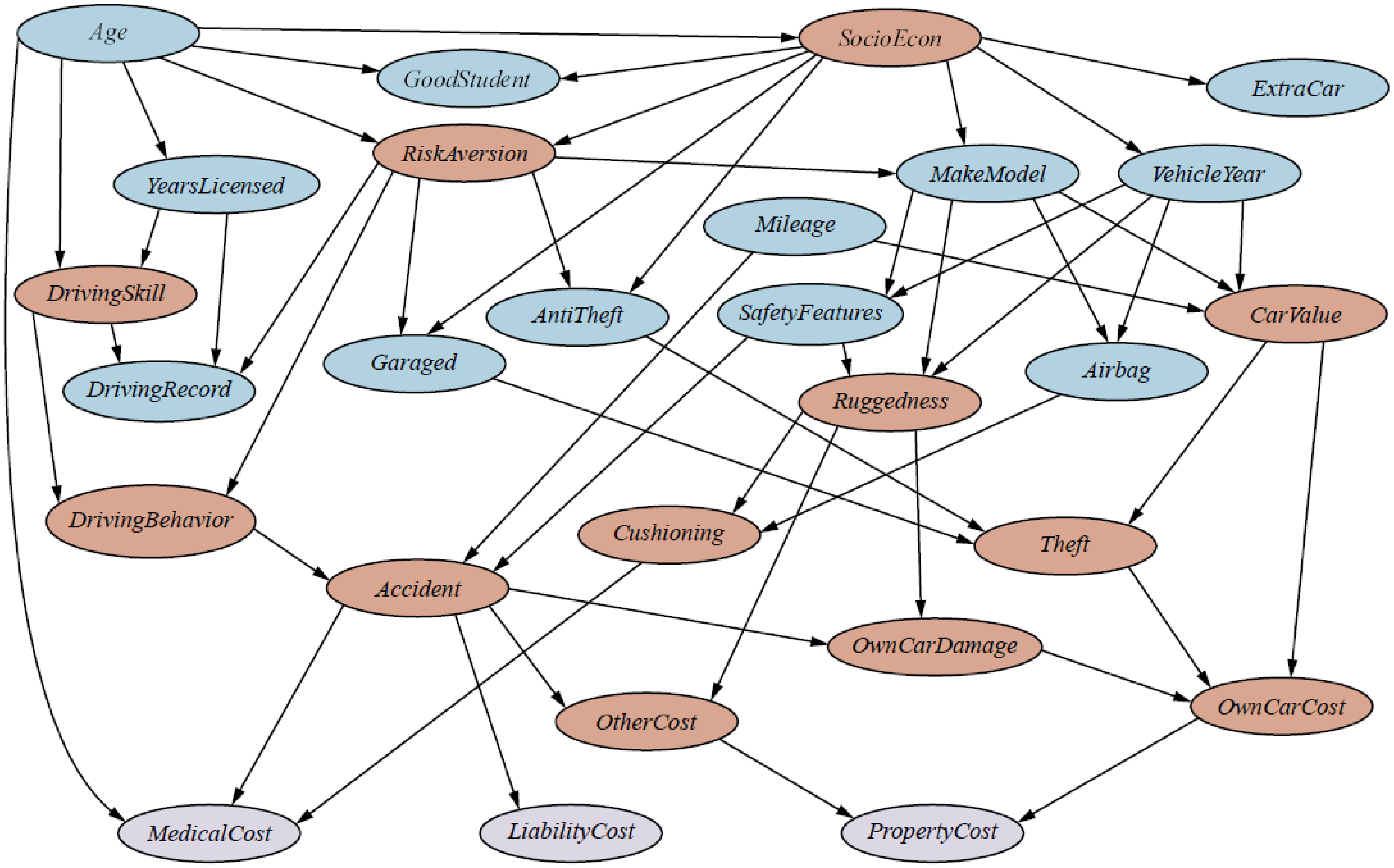
$P(C_1)$

$P(W)$

$P(T|C_1)$

$P(C_2|C_1)$

Example: Car Insurance

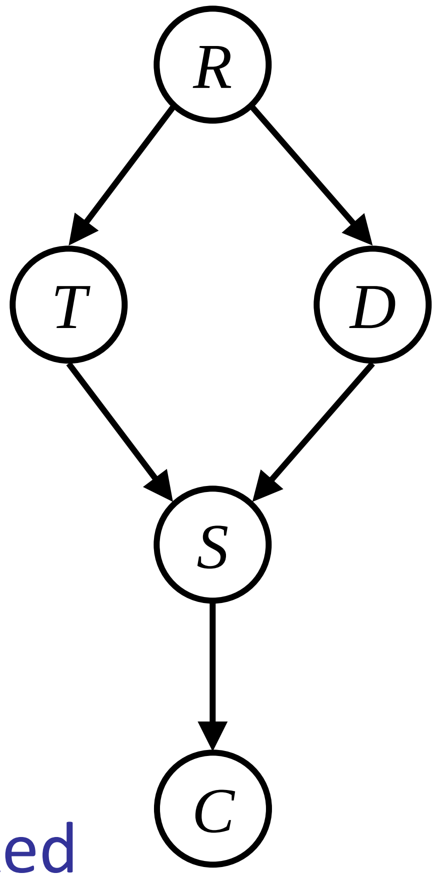


Inferring Conditional Independence

- Recall: A node X is conditionally independent of all non-descendants given observed values of all its parents
- Think of observed nodes as *blocking information flow*

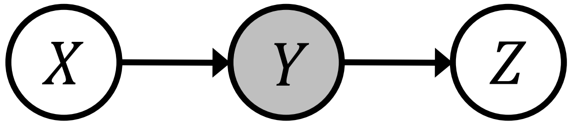
- We can *extend* this independence guarantee to other pairs of nodes if observed nodes also block all paths between them
- Examine local structures of 3 nodes (2 edges) at a time

- X_i and X_j are independent if all paths between them are blocked



Chains and Forks

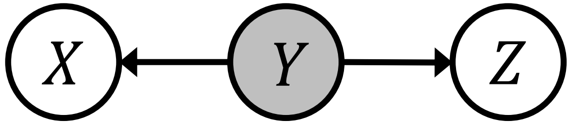
- Generally, nodes X and Z in **chain** and **fork** structures are not independent



$P(X)$ $P(Y|X)$ $P(Z|Y)$

- If Y is observed, then path between X and Z is blocked and they *become* conditionally independent

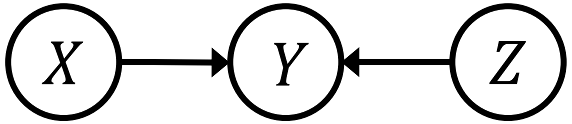
- If removing Y breaks the network into two components, all nodes in X's component become conditionally independent of all nodes in Z's



$P(X|Y)$ $P(Y)$ $P(Z|Y)$

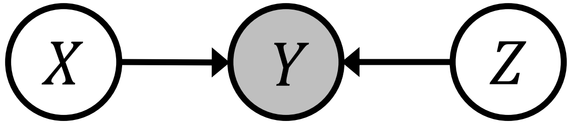
Colliders

- If X and Z share only **colliders** (descendants), the pair is guaranteed to be independent if no colliders are observed



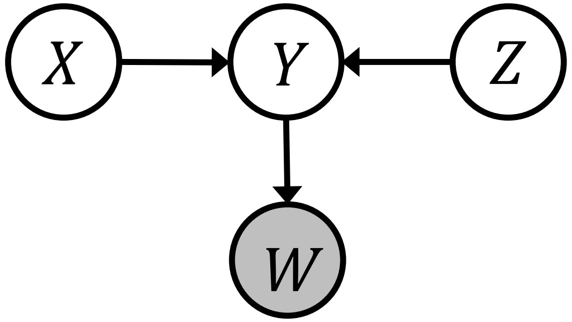
$P(X) \quad P(Y|X,Z) \quad P(Z)$

- But X and Z are *not* guaranteed conditionally independent given observation of a collider!



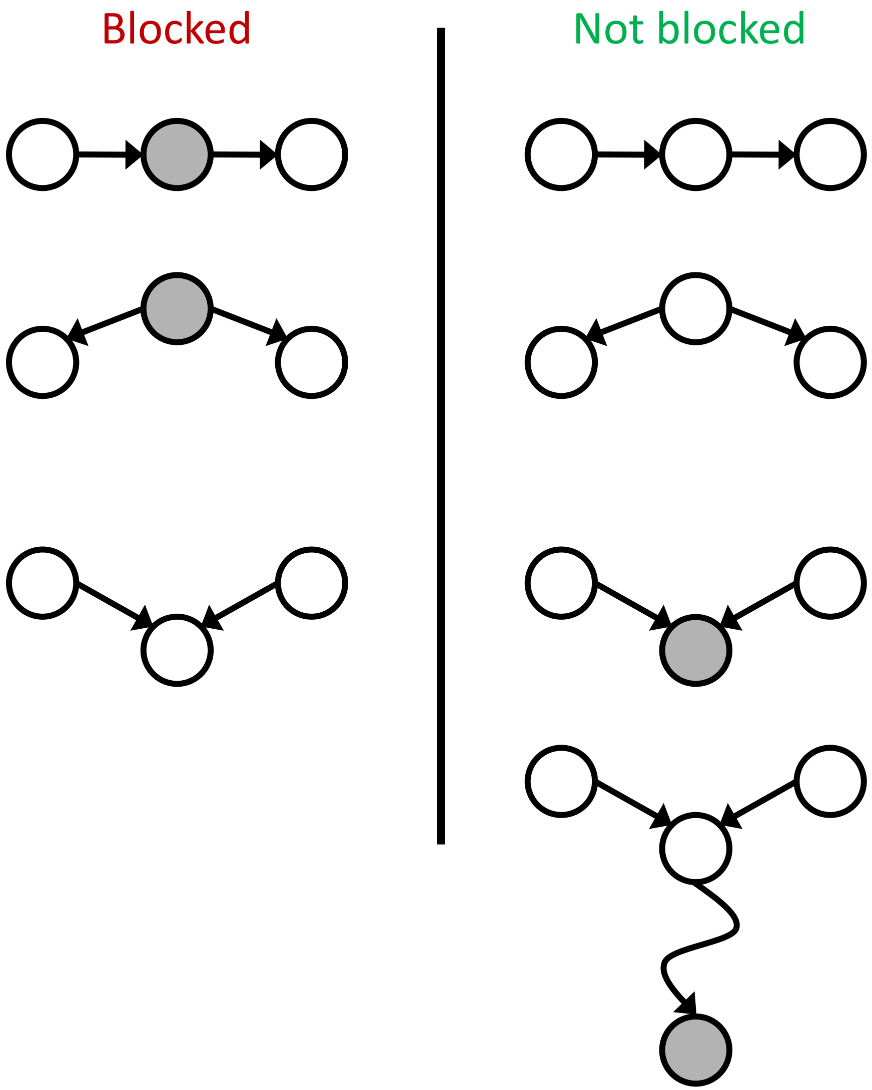
- $P(x,z|y) = \frac{P(x,y,z)}{P(y)} = \frac{P(x)P(z)P(y|x,z)}{P(y)}$
- $P(x|y)P(z|y) = \frac{P(y|x)P(x)}{P(y)}\frac{P(y|z)P(z)}{P(y)}$

Generally not equal!



D-Separation

- To check whether X_i and X_j are conditionally independent given a set of observed nodes Z:
- Check every possible path between X_i and X_j in the "undirected" version of the Bayes net
- **Independent** and "d-separated" if *every* path is blocked; otherwise, **not guaranteed independent** if *at least one* path is not blocked



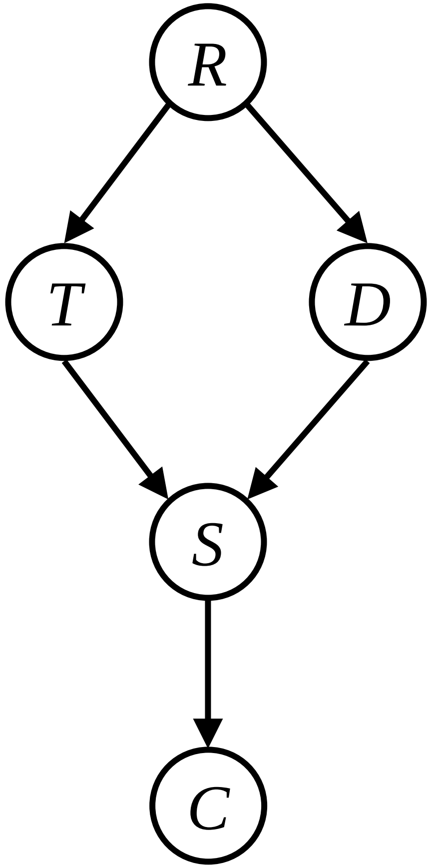
Example: D-Separation

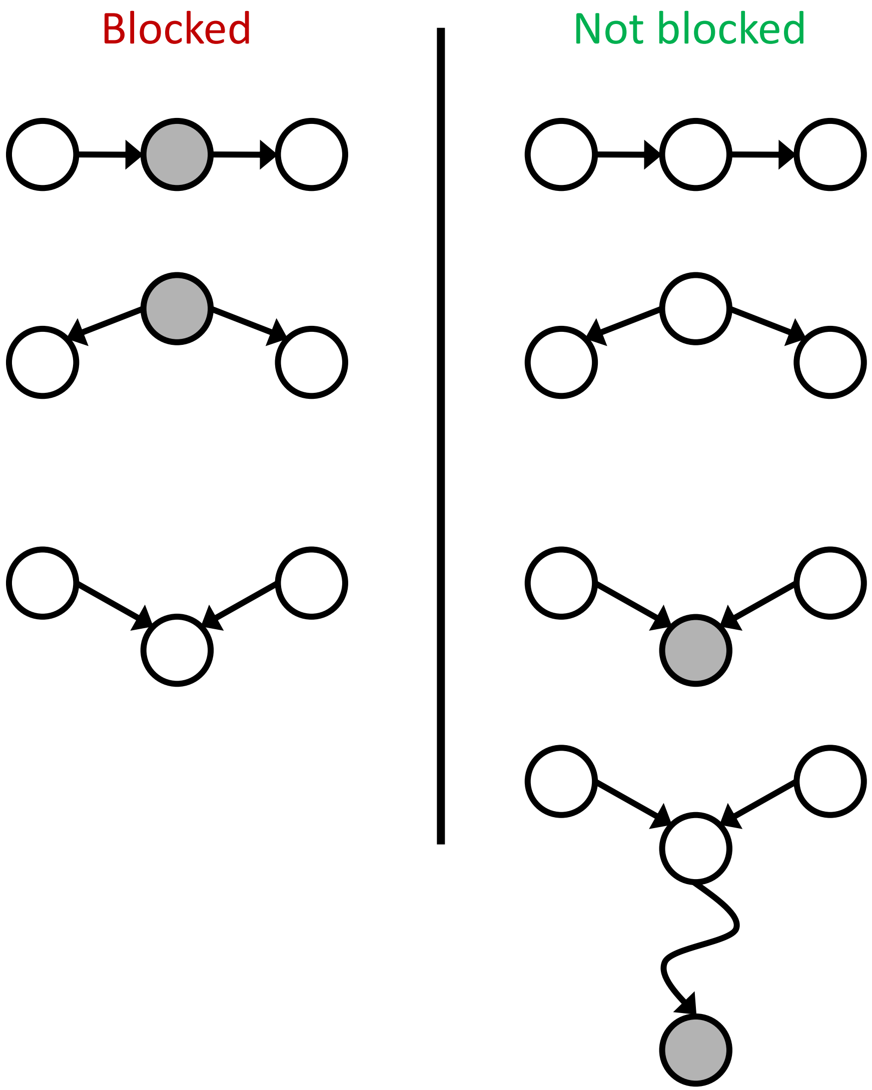
Which nodes are independent…

- Given S?
- Given R?

- Given T or D?
- Given T and D?

- Given R and S?
- Given R and C?





Inference in Bayes Nets

- General task: Find the *posterior* distribution of a set of **query** variables X given a set of observed **evidence** e
- There may also be **hidden** variables Y interacting with X and E

- *Enumeration* strategy: Construct joint distributions via "simplified" chain rule and remove hidden variables via marginalization

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

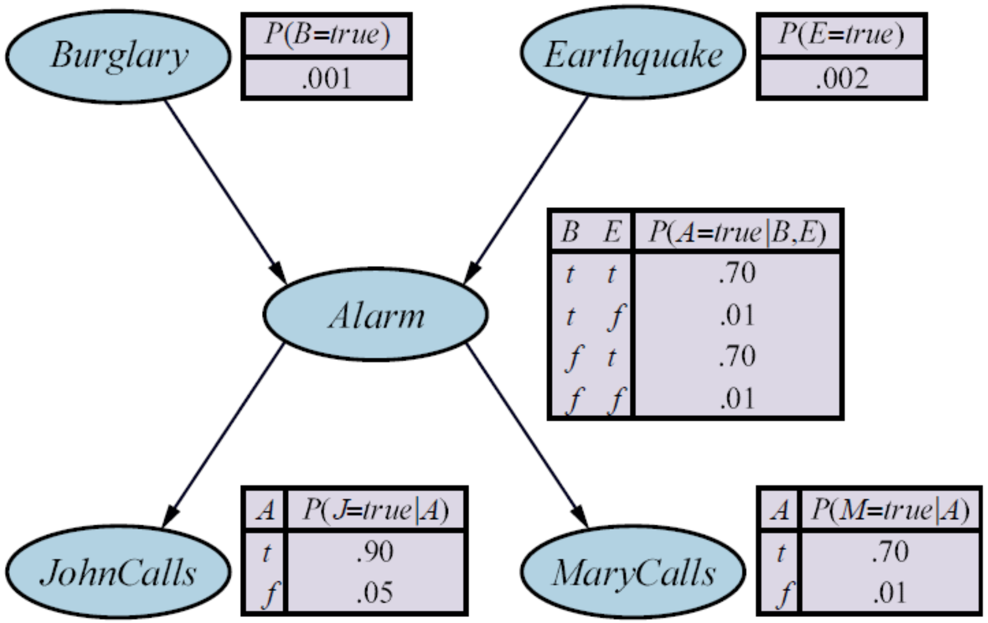
- Y will generally include *ancestors* of X and E but not descendants

Example: Alarm Network

- Y will generally include ancestors of X and E but not descendants

$$P(+b,-e,+a) = P(+b)P(-e)P(+a|+b,-e)$$
$$= (.001)(.998)(.01) = 9.98 \times 10^{-6}$$

$$P(+a) = \sum_{b,e} P(b,e,+a) = \sum_{b,e} P(b)P(e)P(+a|b,e)$$
$$= (.001)(.002)(.7) + (.001)(.998)(.01)$$
$$+ (.999)(.002)(.7) + (.999)(.998)(.01) = .01138$$



$$P(-b|+a) = \frac{\sum_e P(-b,e,+a)}{P(+a)} = \frac{\sum_e P(-b)P(e)P(+a|-b,e)}{P(+a)}$$
$$= \big((.999)(.002)(.7) + (.999)(.998)(.01)\big)/.01138 = .999$$

Example: Alarm Network

- Local independence properties can help simplify expressions before computation

$$P(+j|-a,+e,+b,+m) = P(+j|-a) = 0.05$$

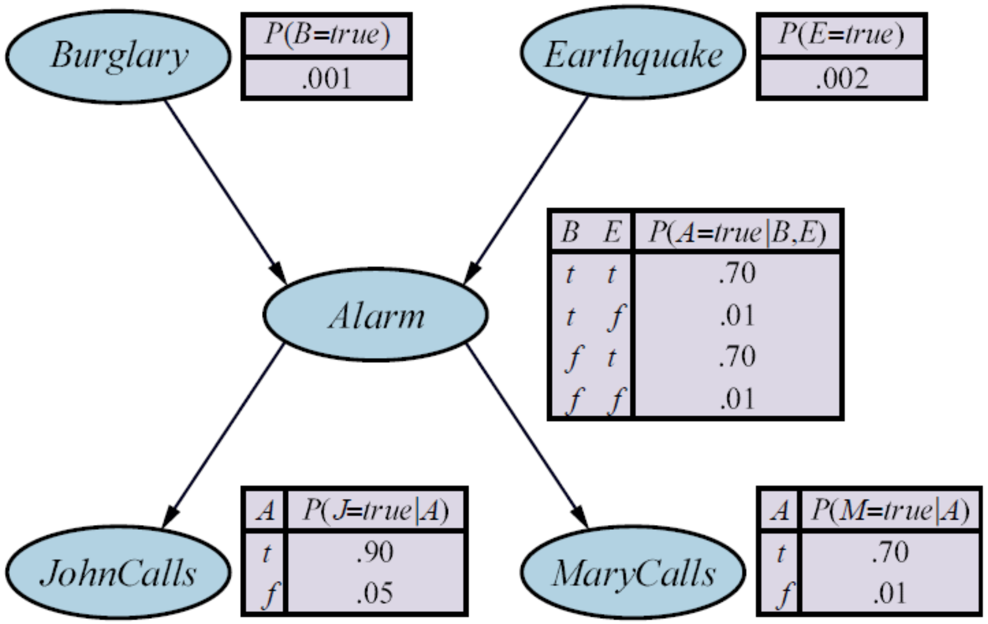
$$P(+j,+m|-a,+e,+b) = P(+j,+m|-a)$$

$$= P(+j|-a)P(+m|-a) = (0.05)(0.01) = 0.0005$$

$$P(+j,+e|-a,+b,+m) = P(+j|-a)P(+e|-a,+b)$$

$$= P(+j|-a)P(+e,-a,+b)/P(-a,+b)$$

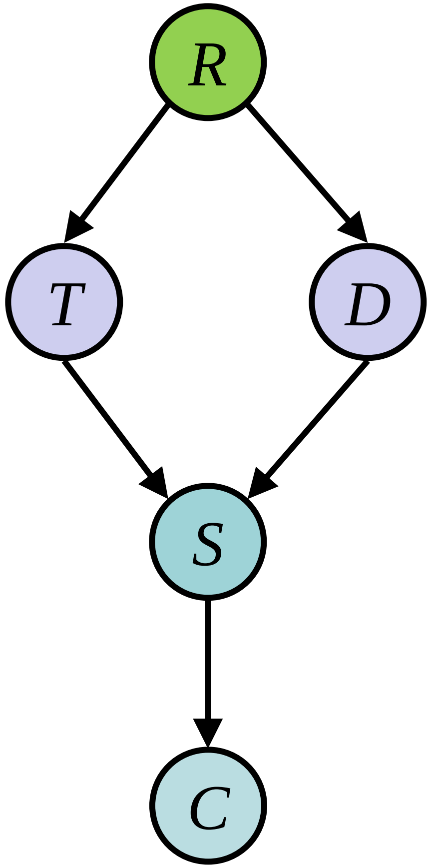
$$= \frac{P(+j|-a)P(+b)P(+e)P(-a|+b,+e)}{\sum_e P(+b)P(e)P(-a|+b,e)} = \frac{(.05)(.001)(.002)(.3)}{(.001)(.002)(.3)+(.001)(.998)(.99)} = 3.05 \times 10^{-5}$$



Querying Distributions

- We can also query entire distributions all at once
- Computational complexity will generally be exponential in number of query and hidden variables
- Ex: $P(R|+s) \propto P(R,+s) = \sum_{t,d} P(R,t,d,+s)$

  $= \sum_{t,d} P(R)P(t|R)P(d|R)P(+s|t,d)$
- Compute joint probabilities over all relevant variables by *multiplying* CPTs, and sum out the hidden variables



Factor Representation

- CPTs may represent marginal distributions, conditional distributions, or neither
- In any case, they are just tables or **factors** over which we are multiplying or adding
- Each factor f_i is a CPT indexed by the values of its input variables

$$P(R|+s) \propto \sum_{t,d} P(R)P(t|R)P(d|R)P(+s|t,d) = \sum_{t,d} f_1(R)f_2(R,t)f_3(R,d)f_4(t,d)$$

- *Multiplying* factors: *Pointwise multiplication* over the common variables, new factor depends on the *union* of dependencies

$$f_1(X,Y) \times f_2(Y,Z) = f_3(X,Y,Z)$$

- *Summing* over a factor: Same as marginalization of a joint distribution

$$\sum_{y} f_3(X,y,Z) = f_4(X,Z)$$

Example

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

R	T	D	P(R,T,D,+s)
+r	+t	+d	(0.5)(0.7)(0.7)(0.1) = .0245
+r	+t	-d	(0.5)(0.7)(0.3)(0.4) = .042
+r	-t	+d	(0.5)(0.3)(0.7)(0.2) = .021
+r	-t	-d	(0.5)(0.3)(0.3)(0.9) = .0405
-r	+t	+d	(0.5)(0.6)(0.6)(0.1) = .018
-r	+t	-d	(0.5)(0.6)(0.4)(0.4) = .048
-r	-t	+d	(0.5)(0.4)(0.6)(0.2) = .024
-r	-t	-d	(0.5)(0.4)(0.4)(0.9) = .072

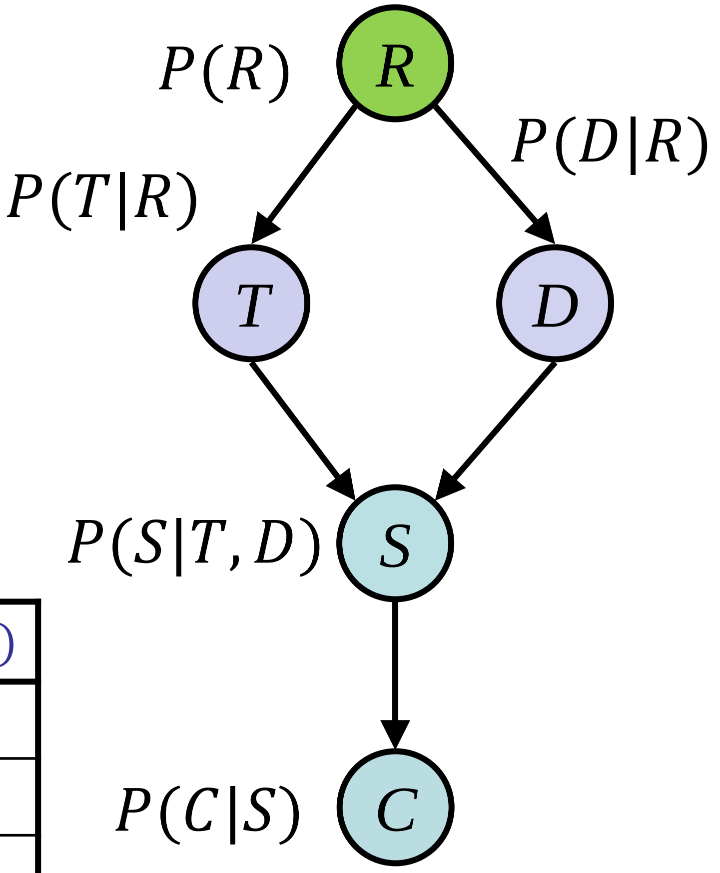
Joint distribution size: $2^3 = 8$ rows

R	$f_1(R)$
+r	0.5
-r	0.5

T	R	$f_2(T,R)$
+t	+r	0.7
+t	-r	0.6
-t	+r	0.3
-t	-r	0.4

D	R	$f_3(D,R)$
+d	+r	0.7
+d	-r	0.6
-d	+r	0.3
-d	-r	0.4

T	D	$f_4(T,D)$
+t	+d	0.1
+t	-d	0.4
-t	+d	0.2
-t	-d	0.9

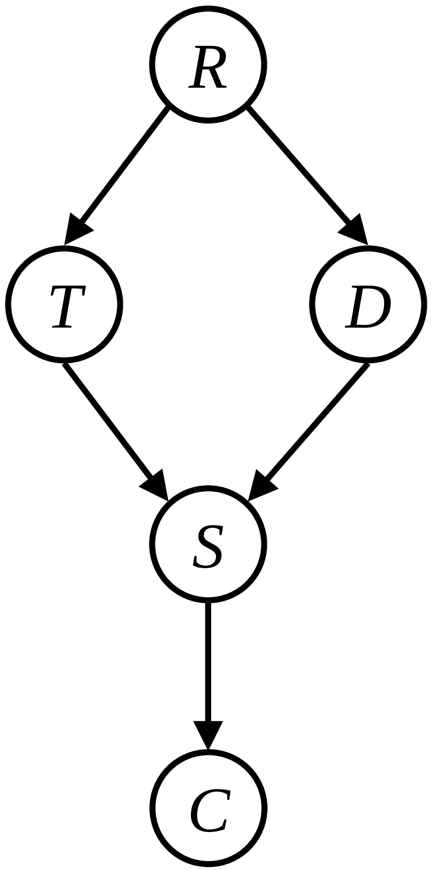


Inference Complexity

- Inference complexity will solely depend on the size of the joint distribution, or number of query and hidden variables
- But we do not have to wait to sum over all variables at the end!

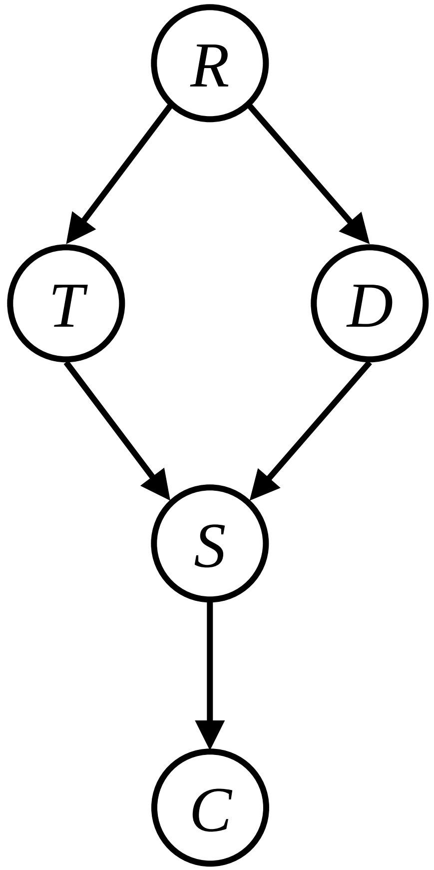
- Better idea: Perform summation over each variable independently
- Factors not dependent on X can be *taken out* of a summation over X

- Ex: $uwy + uwz + uxy + uxz + vwy + vwz + vxy + vxz$ has 16 multiplies and 7 adds
- $(u + v)(wy + wz + xy + xz)$ has 5 multiplies and 4 adds
- $(u + v)(w + x)(y + z)$ has 2 multiplies and 3 adds



Variable Elimination

- Idea: Move summations as far *inwards* as possible
- Marginalization is done starting inside and moving outward



$$P(S|r) \propto P(S,r) = \sum_{t,d} P(r)P(t|r)P(d|r)P(S|t,d)$$
$$= P(r)\sum_{t} P(t|r)\sum_{d} P(d|r)P(S|t,d)$$

$$P(S|c) \propto P(S,c) = \sum_{r,t,d} P(r)P(t|r)P(d|r)P(S|t,d)P(c|S)$$
$$= P(c|S)\sum_{r} P(r)\sum_{t} P(t|r)\sum_{d} P(d|r)P(S|t,d)$$

Example: Variable Elimination

$$P(R|+c,+d) \propto P(R)P(+d|R)\sum_t P(t|R)\sum_s P(s|t,+d)P(+c|s) = f_1(R)f_2(R)\sum_t f_3(R,T)\sum_s f_4(T,S)f_5(S)$$

T	S	$f_6(T,S)$
+t	+s	(0.1)(0.8)
+t	-s	(0.9)(0.3)
-t	+s	(0.2)(0.8)
-t	-s	(0.8)(0.3)

$\sum_s$

T	$f_7(T)$
+t	0.35
-t	0.4

$\times f_3(R,T)$

R	T	$f_8(R,T)$
+r	+t	(0.7)(0.35)
+r	-t	(0.3)(0.4)
-r	+t	(0.6)(0.35)
-r	-t	(0.4)(0.4)

$\sum_t$

R	$f_9(R)$
+r	0.365
-r	0.37

$\times f_1(R) \times f_2(R)$

R	$f_{10}(R)$
+r	(0.5)(0.7)(0.365)
-r	(0.5)(0.6)(0.37)

Max table size is 2^2 rows instead of 2^3

Total operations:
12 multiplies, 4 adds

Variable Ordering

- Elimination ordering does not affect correctness of inference, but *does* greatly affect computational efficiency!

$$P(S, c) = \sum_{r,t,d} f_1(R) f_2(T, R) f_3(D, R) f_4(S, T, D) f_5(S)$$

- R then T then D: $f_5(S) \sum_d \sum_t f_4(S, T, D) \sum_r \underbrace{f_1(R) f_2(T, R) f_3(D, R)}_{\text{8 rows}}$
- 22 multiplies, 10 adds

- T then D then R: $f_5(S) \sum_r f_1(R) \sum_d f_3(D, R) \sum_t \underbrace{f_2(T, R)\, f_4(S, T, D)}_{\text{16 rows}}$
- 30 multiplies, 14 adds

Improving Complexity

- Elimination complexity depends on size of the largest constructed CPT
- NP-hard in the worst case, as this can reduce to a satisfiability problem

- *Greedy* variable ordering can be a good heuristic: Select the next variable that minimizes the size of the constructed CPT
- Still no guarantee of optimal variable ordering

- If Bayes net is a **polytree** (replace all directed edges with undirected edges), elimination can be *linear* if we eliminate *leaves first, then root*

Summary

- Bayesian networks graphically encode independence assumptions about joint distributions in a compact way
- D-separation rules can help infer local independences given evidence

- Inference in Bayesian networks: Computing distributions over query variables given evidence variables (and marginalizing hidden variables)

- Inference by enumeration: Compute full joint distribution of all relevant variables using chain rule, then marginalize hidden variables