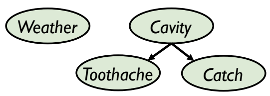
Bayes Nets

AI Class 10 (Ch. 14.1–14.4.2; skim 14.3)



Based on slides by Dr. Marie desJardin. Some material also adapted from slides by Matt E. Taylor @ WSU, Lise Getoor @ UCSC, Dr. P. Matuszek @ Villanova University, and Weng-Keen Wong at OSU. Based in part on www.csc.calpoly.edu/~fkurfess/Courses/CSC-481/W02/Slides/Uncertainty.ppt .

Today's Class

- Bayesian networks
  - Network structure
  - Conditional probability tables
  - Conditional independence
- Inference in Bayesian networks
  - Exact inference
  - Approximate inference

Review: Independence

What does it mean for A and B to be **independent**?

- P(A) ⫫ P(B)
- A and B do not affect each other's probability
- $P(A \wedge B) = P(A)\,P(B)$

Review: Conditioning

What does it mean for A and B to be **conditionally independent given C?**

- A and B don't affect each other **if C is known**
- $P(A \wedge B \mid C) = P(A \mid C)\,P(B \mid C)$

Review: Bayes' Rule

What is **Bayes' Rule**?

$$P(H_i \mid E_j) = \frac{P(E_j \mid H_i)P(H_i)}{P(E_j)}$$

What's it useful for?

- Diagnosis
- Effect is perceived, want to know (probability of) cause

$$P(cause \mid effect) = \frac{P(effect \mid cause)P(cause)}{P(effect)}$$

R&N, 495–496

Review: Joint Probability

- What is the **joint probability** of A and B?
  - P(A,B) (also known as $P(A \wedge B)$)
- The probability of any pair of legal assignments.
  - Generalizing to > 2, of course
- Booleans: expressed as a matrix/table

	alarm	**¬alarm**
burglary	0.09	0.01
¬burglary	0.1	0.8

≡

A	**B**	
T	T	0.09
T	F	0.1
F	T	0.01
F	F	0.8

- Continuous domains: probability functions

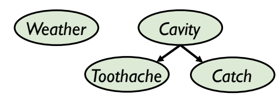
Bayes' Nets: Big Picture

- Problems with full joint distribution tables as our probabilistic models:
  - Joint gets **way** too big to represent explicitly
    - Unless there are only a few variables
  - Hard to learn (estimate) anything empirically about more than a few variables at a time
    - Why?

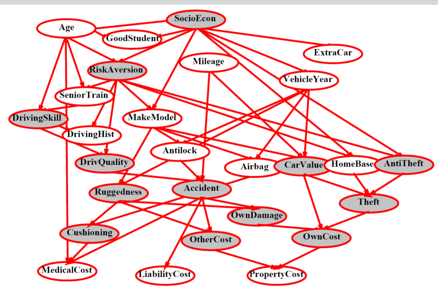
	A		¬A	
	E	¬E	E	¬E
B	0.01	0.08	0.001	0.009
¬B	0.01	0.09	0.01	0.79

Bayes' Nets: Big Picture

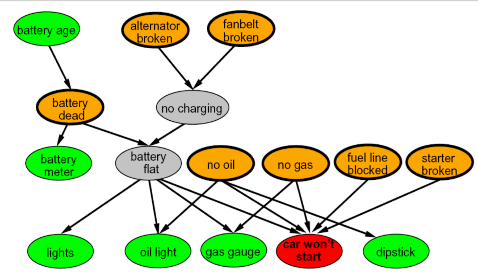
- **Bayes' nets:** a technique for describing complex joint distributions (models) using simple, local distributions (conditional probabilities)
  - A type of graphical models
- We describe how variables interact **locally**
  - Local interactions chain together to give global, indirect interactions



Example: Insurance

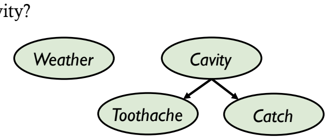


Example: Car



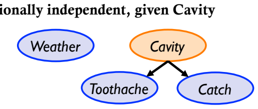
Example: Toothache

- Random variables:
  - How's the weather?
  - Do you have a toothache?
  - Does the dentist's probe catch when she pokes your tooth?
  - Do you have a cavity?



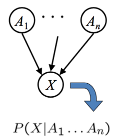
Graphical Model Notation

- Nodes: variables (with domains)
- Can be assigned (observed) or unassigned (unobserved)
- Arcs: interactions
  - Indicate "direct influence" between
  - Formally: **encode conditional independence**
    - Toothache and Catch are **conditionally independent, given Cavity**
- For now: imagine that arrows mean *causation*
  - (in general, they don't!)



Bayesian Belief Networks (BNs)

- Let's formalize the semantics of a BN
  - A set of nodes, one per variable *X*
- A directed arc between each con-influential node
  - *X* → *Y* means *X* **has an influence on** *Y*
- A directed, acyclic graph

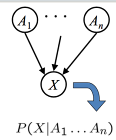


Bayesian Belief Networks (BNs)

- Each node *X* has a **conditional probability distribution**:

  $P(X_i \mid Parents(X_i))$

  - A collection of distributions over *X*
  - One for each combination of parents' values
  - Quantifies the effects of the parents on a node
- CPT: conditional probability table
  - Description of a noisy "causal" process



Conditional Probability Tables

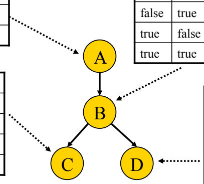
- For X_i, CPD $P(X_i \mid Parents(X_i))$ quantifies effect of parents on X_i
- **Parameters** are probabilities in conditional probability tables (CPTs):

A	P(A)
false	0.6
true	0.4

A	B	P(B|A)
false	false	0.01
false	true	0.99
true	false	0.7
true	true	0.3

B	C	P(C|B)
false	false	0.4
false	true	0.6
true	false	0.9
true	true	0.1

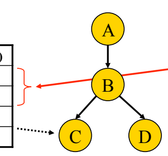
B	D	P(D|B)
false	false	0.02
false	true	0.98
true	false	0.05
true	true	0.95



CPTs cont'd

- Conditional Probability Distribution for C given B
- If you have a Boolean variable with k Boolean parents, this table has 2^{k+1} probabilities

B	C	P(C|B)
false	false	0.4
false	true	0.6
true	false	0.9
true	true	0.1



For a given combination of values of the parents (B in this example), the entries for P(C=true | B) and P(C=false | B) must sum to 1
Example:
P(C=true | B=false) +
P(C=false |B=false) = 1

Bayesian Belief Networks (BNs)

- Definition: **BN = (DAG, CPD)**
  - **DAG**: directed acyclic graph (BN's **structure**)
    - **Nodes**: random variables
      - Typically binary or discrete
      - Methods exist for continuous variables
    - **Arcs**: indicate probabilistic dependencies between nodes
      - *Lack* of link signifies conditional independence
  - **CPD**: conditional probability distribution (BN's **parameters**)
    - Conditional probabilities at each node, usually stored as a table (conditional probability table, or **CPT**)

Bayesian Belief Networks (BNs)

- Definition: **BN = (DAG, CPD)**
  - **DAG**: directed acyclic graph (BN's structure)
    - **Nodes**: random variables
      - Typically binary or discrete
      - Methods exist for continuous variables
- **Arcs**: indicate probabilistic dependencies between nodes
  - *Lack* of link signifies conditional independence!
- **CPD**: conditional probability distribution (the BN's parameters)
  - Conditional probabilities at each node, usually stored as a table (conditional probability table, or **CPT**)

Bayesian Belief Networks (BNs)

- Definition: **BN = (DAG, CPD)**
  - **DAG**: directed acyclic graph (BN's **structure**)
  - **CPD**: conditional probability distribution (BN's **parameters**)
    - Conditional probabilities at each node, usually stored as a table (conditional probability table, or **CPT**)

$$P(x_i \mid \pi_i) \text{ where } \pi_i \text{ is the set of all parent nodes of } x_i$$

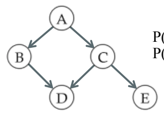
  - Root nodes are a special case
    - No parents, so use priors in CPD:

$$\pi_i = \varnothing, \text{ so } P(x_i \mid \pi_i) = P(x_i)$$

Example BN

P(A) = 0.001

P(B|A) = 0.3
P(B|¬A) = 0.001



P(C|A) = 0.2
P(C|¬A) = 0.005

P(D|B,C) = 0.1
P(D|B,¬C) = 0.01
P(D|¬B,C) = 0.01
P(D|¬B,¬C) = 0.00001

P(E|C) = 0.4
P(E|¬C) = 0.002

We only specify P(A) etc., not P(¬A), since they have to sum to one

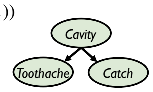
Probabilities in BNs

- Bayes' nets implicitly **encode joint distributions** as a **product of local conditional distributions**.
- To see probability of a **full assignment**, multiply all the relevant conditionals together:

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

  - Example:

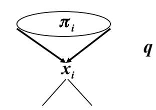
$$P(+\text{cavity}, +\text{catch}, \neg\text{toothache}) = ?$$



- This lets us reconstruct any entry of the full joint

Conditional Independence and Chaining

- Conditional independence assumption: $P(x_i \mid \pi_i, q) = P(x_i \mid \pi_i)$
  - $\boldsymbol{q}$ is any set of variables (nodes) other than $\boldsymbol{x_i}$ and its successors
  - π_i **blocks influence** of other nodes on x_i and its successors
    - That is, $\boldsymbol{q}$ influences x_i only through variables in π_i)



  - With this assumption, complete joint probability distribution of all variables in the network can be represented by (recovered from) local CPDs by chaining these CPDs:

$$P(x_1, \ldots, x_n) = \Pi_{i=1}^{n} P(x_i \mid \pi_i)$$

The Chain Rule

- $P(\alpha_1 \wedge \alpha_2 \wedge \ldots \wedge \alpha_n) = P(\alpha_1) \times P(\alpha_2 \mid \alpha_1) \times P(\alpha_3 \mid \alpha_1 \wedge \alpha_2) \times \ldots \times P(\alpha_n \mid \alpha_1 \wedge \cdots \wedge \alpha_{n-1})$

$$= \Pi_{i=1..n} P(\alpha_i \mid \alpha_1 \wedge \cdots \wedge \alpha_{i-1})$$

$$= P(x_1, \ldots, x_n) = \Pi_{i=1}^{n} P(x_i \mid \pi_i)$$

The Chain Rule

$P(x_1, \ldots, x_n) = \Pi_{i=1}^{n} P(x_i \mid \pi_i)$

e.g, $P(x_1, \ldots, x_n) = P(x_1) P(x_2 \mid x_1) P(x_3 \mid x_1, x2) \ldots$

- Decomposition:

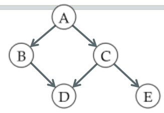
$P(\text{Traffic, Rain, Umbrella}) = P(\text{Rain})\, P(\text{Traffic} \mid \text{Rain})\, P(\text{Umbrella} \mid \text{Rain, Traffic})$

- With assumption of conditional independence:

$P(\text{Traffic, Rain, Umbrella}) = P(\text{Rain})\, P(\text{Traffic} \mid \text{Rain})\, P(\text{Umbrella} \mid \text{Rain})$

- Bayes' nets express conditional independence assumptions

Chaining: Example



Computing the joint probability for all variables is easy:

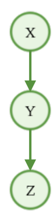
P(a, b, c, d, e)
= P(e | a, b, ***c***, d) P(a, b, c, d) *by the product rule*
= P(e | c) P(a, b, c, d) *by cond. indep. assumption*
= P(e | c) P(d | a, ***b, c***) P(a, b, c)
= P(e | c) P(d | b, c) P(c | ***a,*** b) P(a, b)
= P(e | c) P(d | b, c) P(c | a) P(b | a) P(a)

Topological Semantics

- A node is **conditionally independent** of its **non-descendants** given its **parents**
- A node is **conditionally independent** of all other nodes in the network given its parents, children, and children's parents (also known as its **Markov blanket**)
- A method called **d-separation** can be applied to decide whether a set of nodes X is independent of another set Y, given a third set Z

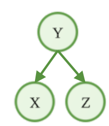
Independence and Causal Chains

- Important question about a BN:
  - Are two nodes independent given certain evidence?
  - If yes, can prove using algebra (tedious in general)
  - If no, can prove with a counter-example
- Question: are X and Z necessarily independent?
  - No. (E.g., low pressure causes rain, which causes traffic)
  - X can influence Z, Z can influence X (via Y)
- This configuration is a "causal chain"

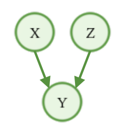


Two More Main Patterns

- Common Cause:
  - Y cause X *and* Y causes Z
  - Are X and Z independent?
  - Are X and Z independent given Y?



- Common Effect:
  - Two causes of one effect
  - Are X and Z independent? (yes)
  - Are X and Z independent given Y?
  - →**No**!
  - Observing an effect "*activates*" influence between possible causes.



Inference in Bayesian Networks

Chapter 14.4.1-14.4.2

Inference Tasks

- **Simple queries:** Compute posterior marginal P(X_i | E=e)
  - E.g., P(NoGas | Gauge=empty, Lights=on, Starts=false)
- **Conjunctive queries:**
  - P(X_i, X_j | E=e) = P(X_i | e=e) P(X_j | X_i, E=e)
- **Optimal decisions:**
  - *Decision networks* include utility information
  - Probabilistic inference gives P(outcome | action, evidence)
- **Value of information:** Which evidence should we seek next?
- **Sensitivity analysis:** Which probability values are most critical?
- **Explanation:** Why do I need a new starter motor?

Approaches to Inference

- Exact inference
  - **Enumeration**
  - Belief propagation in polytrees
  - **Variable elimination**
  - Clustering / join tree algorithms
- Approximate inference
  - Stochastic simulation / sampling methods
  - Markov chain Monte Carlo methods
  - Genetic algorithms
  - Neural networks
  - Simulated annealing
  - Mean field theory

Direct Inference with BNs

- Instead of computing the joint, suppose we just want the probability for *one* variable
- Exact methods of computation:
  - **Enumeration**
  - **Variable elimination**
  - **Join trees: get the probabilities associated with every query variable**

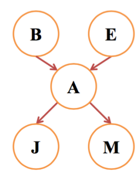
Inference by Enumeration

- Add all of the terms (atomic event probabilities) from the full joint distribution
- If **E** are the evidence (observed) variables and **Y** are the other (unobserved) variables, then:
  $P(X \mid \mathbf{e}) = \alpha P(X, \mathbf{E}) = \alpha \sum P(X, \mathbf{E}, \mathbf{Y})$
- Each P(X, **E**, **Y**) term can be computed using the chain rule
- Computationally expensive!

Example 1: Enumeration

- Recipe:
  - State the marginal probabilities you need
  - Figure out ALL the atomic probabilities you need
  - Calculate and combine them
- Example:
  - P(+b | +j, +m) =

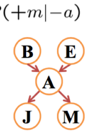
$$\frac{P(+b, +j, +m)}{P(+j, +m)}$$



Example 1 cont'd

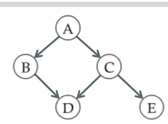
$$P(+b,+j,+m) =$$
$$P(+b)P(+e)P(+a|+b,+e)P(+j|+a)P(+m|+a)+$$
$$P(+b)P(+e)P(-a|+b,+e)P(+j|-a)P(+m|-a)+$$
$$P(+b)P(-e)P(+a|+b,-e)P(+j|+a)P(+m|+a)+$$
$$P(+b)P(-e)P(-a|+b,-e)P(+j|-a)P(+m|-a)$$

P(+m | +b, +e)?



Example 2: Enumeration

- $P(x_i) = \Sigma_{\pi i} P(x_i \mid \pi_i) P(\pi_i)$
- Suppose we want *P*(D=true)
- Only E is *given* as true
- $P(d \mid e) = \alpha \Sigma_{ABC} P(a, b, c, d, e)$ *(where α = 1/P(e))*
  $= \alpha \Sigma_{ABC} P(a) P(b \mid a) P(c \mid a) P(d \mid b,c) P(e \mid c)$
- With simple iteration, that's a lot of repetition!
  - P(e|c) has to be recomputed every time we iterate over C=true



Variable Elimination

- Basically just enumeration, but with caching of local calculations
- Linear for polytrees (singly connected BNs)
- Potentially exponential for multiply connected BNs
  $\Rightarrow$ **Exact inference in Bayesian networks is NP-hard!**
- Join tree algorithms are an extension of variable elimination methods that compute posterior probabilities for **all** nodes in a BN simultaneously

Variable Elimination Approach

General idea:

- Write query in the form

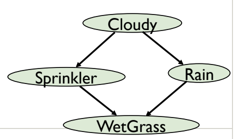
$$P(X_n, e) = \sum_{x_k} \cdots \sum_{x_3} \sum_{x_2} \prod_i P(x_i \mid pa_i)$$

  - Note that there is no α term here
  - It's a conjunctive probability, not a conditional probability…
- Iteratively
  - Move all irrelevant terms outside of innermost sum
  - Perform innermost sum, getting a new term
  - Insert the new term into the product

Variable Elimination: Example

$$P(w) = \sum_{r,s,c} P(w \mid r,s) P(r \mid c) P(s \mid c) P(c)$$
$$= \sum_{r,s} P(w \mid r,s) \sum_{c} P(r \mid c) P(s \mid c) P(c)$$
$$= \sum_{r,s} P(w \mid r,s) f_1(r,s)$$

"factors"



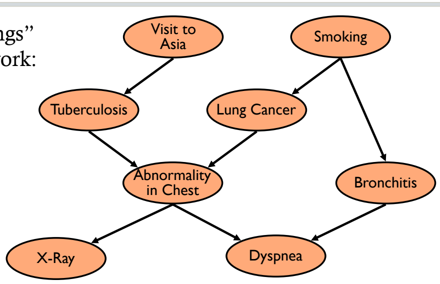
Computing Factors

R	S	C	P(R|C)	P(S|C)	P(C)	P(R|C) P(S|C) P(C)
T	T	T				
T	T	F				
T	F	T				
T	F	F				
F	T	T				
F	T	F				
F	F	T				
F	F	F				

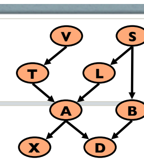
R	S	$f_1(R,S) = \sum_c P(R\|S)\ P(S\|C)\ P(C)$
T	T	
T	F	
F	T	
F	F	

A More Complex Example

- "Lungs" network:



Lungs 1

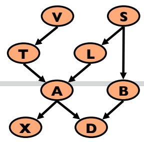


- We want to compute *P(d)*
- Need to eliminate: *v,s,x,t,l,a,b*

Initial factors:

$$P(v)P(s)P(t \mid v)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

Lungs 2



- We want to compute *P(d)*
- Need to eliminate: *v,s,x,t,l,a,b*

Initial factors:

$$P(v)P(s)P(t|v)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$$

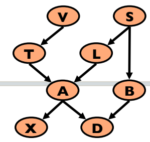
Eliminate: *v*

Compute: $f_v(t) = \sum_v P(v)P(t|v)$

$$\Rightarrow f_v(t)P(s)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$$

- Note: $f_v(t) = P(t)$
- In general, result of elimination is not necessarily a probability term

Lungs 3



- We want to compute *P(d)*
- Need to eliminate: *s,x,t,l,a,b*

Initial factors:

$$P(v)P(s)P(t|v)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$$

$$\Rightarrow f_v(t)P(s)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$$

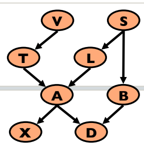
Eliminate: *s*

Compute: $f_s(b,l) = \sum_s P(s)P(b|s)P(l|s)$

$$\Rightarrow f_v(t)f_s(b,l)P(a|t,l)P(x|a)P(d|a,b)$$

- Summing on *s* results in a factor with two arguments $f_s(b,l)$
- In general, result of elimination may be a function of several variables

Lungs 4



- We want to compute *P(d)*
- Need to eliminate: *x,t,l,a,b*

Initial factors

$$P(v)P(s)P(t|v)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$$

$$\Rightarrow f_v(t)P(s)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$$

$$\Rightarrow f_v(t)f_s(b,l)P(a|t,l)P(x|a)P(d|a,b)$$

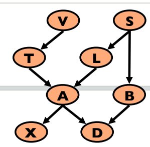
Eliminate: *x*

Compute: $f_x(a) = \sum_x P(x|a)$

$$\Rightarrow f_v(t)f_s(b,l)f_x(a)P(a|t,l)P(d|a,b)$$

Note: $f_x(a) = 1$ for all values of *a !!*

Lungs 5



- We want to compute *P(d)*
- Need to eliminate: *t,l,a,b*

Initial factors $P(v)P(s)P(t|v)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$

$$\Rightarrow f_v(t)P(s)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$$

$$\Rightarrow f_v(t)f_s(b,l)P(a|t,l)P(x|a)P(d|a,b)$$

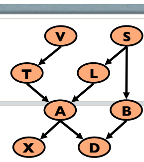
$$\Rightarrow f_v(t)f_s(b,l)f_x(a)P(a|t,l)P(d|a,b)$$

Eliminate: *t*

Compute: $f_t(a,l) = \sum_t f_v(t)P(a|t,l)$

$$\Rightarrow f_s(b,l)f_x(a)f_t(a,l)P(d|a,b)$$

Lungs 6



- We want to compute *P(d)*
- Need to eliminate: *l,a,b*

Initial factors $P(v)P(s)P(t|v)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$

$$\Rightarrow f_v(t)P(s)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$$

$$\Rightarrow f_v(t)f_s(b,l)P(a|t,l)P(x|a)P(d|a,b)$$

$$\Rightarrow f_v(t)f_s(b,l)f_x(a)P(a|t,l)P(d|a,b)$$

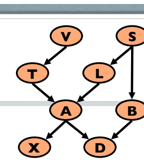
$$\Rightarrow f_s(b,l)f_x(a)f_t(a,l)P(d|a,b)$$

Eliminate: *l*

Compute: $f_l(a,b) = \sum_l f_s(b,l)f_t(a,l)$

$$\Rightarrow f_l(a,b)f_x(a)P(d|a,b)$$

Lungs Finale



- We want to compute *P(d)*
- Need to eliminate: *b*

Initial factors $P(v)P(s)P(t|v)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$

$$\Rightarrow f_v(t)P(s)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$$

$$\Rightarrow f_v(t)f_s(b,l)P(a|t,l)P(x|a)P(d|a,b)$$

$$\Rightarrow f_v(t)f_s(b,l)f_x(a)P(a|t,l)P(d|a,b)$$

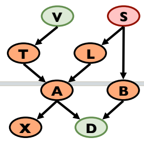
$$\Rightarrow f_s(b,l)f_x(a)f_t(a,l)P(d|a,b)$$

$$\Rightarrow f_l(a,b)f_x(a)P(d|a,b) \Rightarrow f_a(b,d) \Rightarrow f_b(d)$$

Eliminate: *a,b*

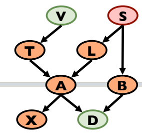
Compute: $f_a(b,d) = \sum_a f_l(a,b)f_x(a)p(d|a,b)$ $f_b(d) = \sum_b f_a(b,d)$

Dealing with Evidence



- How do we deal with evidence?
  - And what is "evidence?"
  - Variables whose value has been observed
- Suppose we are given evidence: $V = t,\ S = f,\ D = t$
- We want to compute $P(L,\ V = t,\ S = f,\ D = t)$

Dealing with Evidence



- We start by writing the factors:

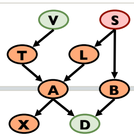
$$P(v)P(s)P(t \mid v)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

- Since we know that $V = t$, we don't need to eliminate V
- Instead, we can replace the factors $P(V)$ and $P(T/V)$ with

$$f_{P(V)} = P(V = t) \quad f_{p(T|V)}(T) = P(T \mid V = t)$$

- These "select" appropriate parts of original factors given evidence
- Note that $f_{P(V)}$ is a constant, so **does not appear** in elimination of other variables

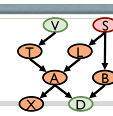
Dealing with Evidence



- So now…
  - Given evidence $V = t,\ S = f,\ D = t$
  - Compute $P(L,\ V = t,\ S = f,\ D = t)$
  - Initial factors, after setting evidence:

$$f_{P(v)}f_{P(s)}f_{P(t|v)}(t)f_{P(l|s)}(l)f_{P(b|s)}(b)P(a \mid t,l)P(x \mid a)f_{P(d|a,b)}(a,b)$$

Dealing with Evidence



- Given evidence $V = t,\ S = f,\ D = t$, we want to compute $P(L,\ V = t,\ S = f,\ D = t)$
- Initial factors, after setting evidence:

$$f_{P(v)}f_{P(s)}f_{P(t|v)}(t)f_{P(l|s)}(l)f_{P(b|s)}(b)P(a \mid t,l)P(x \mid a)f_{P(d|a,b)}(a,b)$$

- Eliminating x, we get

$$f_{P(v)}f_{P(s)}f_{P(t|v)}(t)f_{P(l|s)}(l)f_{P(b|s)}(b)P(a \mid t,l)f_x(a)f_{P(d|a,b)}(a,b)$$

- Eliminating t, we get

$$f_{P(v)}f_{P(s)}f_{P(l|s)}(l)f_{P(b|s)}(b)f_t(a,l)f_x(a)f_{P(d|a,b)}(a,b)$$

- Eliminating a, we get

$$f_{P(v)}f_{P(s)}f_{P(l|s)}(l)f_{P(b|s)}(b)f_a(b,l)$$

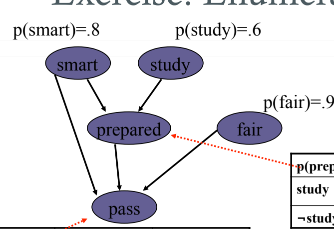
- Eliminating b, we get

$$f_{P(v)}f_{P(s)}f_{P(l|s)}(l)f_b(l)$$

Variable Elimination Algorithm

- Let $X_1, \ldots, X_m$ be an ordering on the non-query variables
- For i = m, …, 1 $\sum_{X_1}\sum_{X_2}\ldots\sum_{X_m}\prod_j P(X_j \mid Parents(X_j))$
  - In the summation for X_i, leave only factors mentioning X_i
  - Multiply the factors, getting a factor that contains a number for each value of the variables mentioned, including X_i
  - Sum out X_i, getting a factor f that contains a number for each value of the variables mentioned, not including X_i
  - Replace the multiplied factor in the summation

Exercise: Enumeration

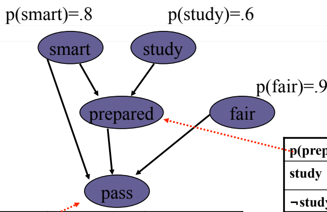


p(prep|…)	smart	¬smart
study	.9	.7
¬study	.5	.1

p(pass|…)	smart		¬smart	
	prep	¬prep	prep	¬prep
fair	.9	.7	.7	.2
¬fair	.1	.1	.1	.1

Query: What is the probability that a student **studied**, given that they pass the exam?

Exercise: Variable Elimination



p(prep|…)	smart	¬smart
study	.9	.7
¬study	.5	.1

p(pass|…)	smart		¬smart	
	prep	¬prep	prep	¬prep
fair	.9	.7	.7	.2
¬fair	.1	.1	.1	.1

Query: What is the probability that a student **is smart**, given that they pass the exam?

Summary

- **Bayes nets**
  - **Structure**
  - **Parameters**
  - **Conditional independence**
  - **Chaining**
- **BN inference**
  - **Enumeration**
  - **Variable elimination**
  - Sampling methods