Chap. 14

Probabilistic Reasoning: Bayesian Networks Model

How to build network models to reason under uncertainty according to the laws of probability theory?

Outline

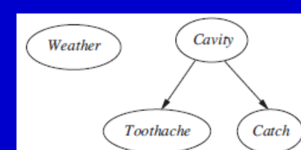
- Syntax
- Semantics
- Parameterized Distributions
- Exact inference by enumeration
- Exact inference by variable elimination
- Approximate inference
  - by stochastic simulation
  - by Markov Chain Monte Carlo

Bayesian networks

- A simple, *graphical notation for conditional independence assertions* and hence for compact specification of full joint distributions: called, *Belief Network, Probabilistic Network, Causal Network, Knowledge Map.*
  - Diagnostic Rule: Observed *Effect* ⇒ Hidden *Causes*
  - Causal Rule: Hidden *Causes*(*Property*) ⇒ *Effect* (*Percept*)
- Syntax:
  - a set of nodes, one per variable
  - a directed, acyclic graph (link ≈ directly "influences")
  - a conditional distribution for each node given its parents:
    $P(X_i \mid Parents(X_i))$: a quantity that an effect of the parents on 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

Example

- Topology of network encodes conditional independence assertions:

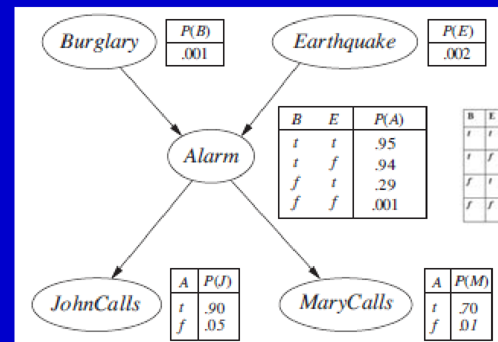


- *Weather* is independent of the other variables
- *Toothache* and *Catch* are conditionally independent given *Cavitiy*

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

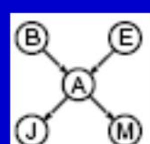
Example contd.



B	E	P(a | B, E)	P(¬a | B, E)
t	t	P(a | b, e) = .95	P(¬a | b, e) = 1 - P(a | b, e) = .05
t	f	P(a | b, ¬e) = .94	P(¬a | b, ¬e) = 1 - P(a | b, ¬e) = .06
f	t	P(a | ¬b, e) = .29	P(¬a | ¬b, e) = 1 - P(a | ¬b, e) = .71
f	f	P(a | ¬b, ¬e) = .001	P(¬a | ¬b, ¬e) = 1 - P(a | ¬b, ¬e) = .999

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, $2^0 + 2^0 + 2^2 + 2^1 + 2^1 = 10$ numbers (vs. $2^5 - 1 = 31$)

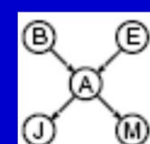


Global semantics

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

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

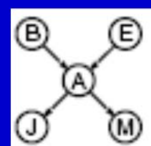
- E.g.) $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e) =$



Global semantics

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

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



- E.g.) $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$
  $= P(j|a)P(m|a)P(a|\neg b, \neg e)P(\neg b)P(\neg e)$
  $= 0.9 \times 0.7 \times 0.001 \times 0.999 \times 0.998$
  ≈ 0.00063

Local semantics

- **Local semantics –** descendants:
  each node is *conditionally independent* ($\perp$) of its *non-descendants(ancestors+, siblings, cousins, uncles, etc.)* given its parents.
  - E.g.)
    - $\forall i, j,\ X \perp Z_j | U_j$

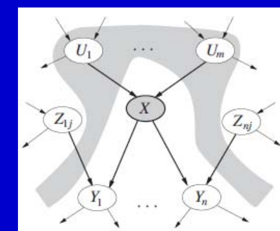
    i.e. $P(X | Z_j, U_j) = P(X | U_i)$
    - $Y_1 \perp U_k | X, Z_{1j}$

    i.e. $P(Y_1 | X, Z_{1j}, U_k) = P(Y_1 | X, Z_{1j})$

    $J \perp B, E, M | A$

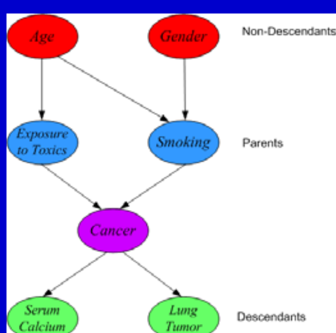
    i.e. $P(J | B, E, A, M) = P(J | A)$

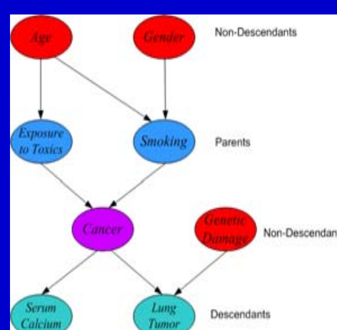
    $P(J, M | B, E, A) = P(J | A) \cdot P(M | A)$



- Theorem: Local semantics (topological semantics)
  $\Leftrightarrow$ Global semantics (numerical semantics)

Continued..





Markov blanket

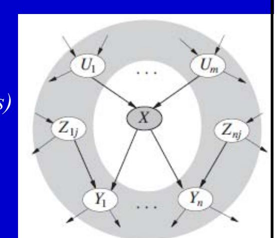
- **Local semantics –Markov blanket**:
  Each node is *conditionally independent* of others given its **Markov blanket**: parents + children + *children's parents*
  - E.g)

    $X \perp Others | Shades$

    i.e. $P(X | Others, Shades) = P(X | Shades)$

    $B \perp M, J | A, E$

    i.e. $P(B | M, J, A, E) = P(B | A, E)$



Constructing Bayesian Networks

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

1. Choose an ordering of Variables $X_1, \ldots, X_n$
2. For i = 1 to n
    1. Add X_i to the network
    2. Select parents from $X_1, \ldots, X_{i-1}$ such that
       $P(X_i \mid Parents(X_i)) = P(X_i \mid X_1, \ldots, X_{i-1})$

- This choice of parents guarantees the global semantics:
  $P(X_1, \ldots, X_n) = \prod_{i=1..n} P(X_i \mid X_1, \ldots, X_{i-1})$ (chain rule)
  $= \prod_{i=1..n} P(X_i \mid Parents(X_i))$ (by construction)

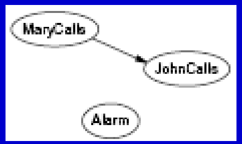
Example

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

$P(J \mid M) = P(J)$?

Example contd.

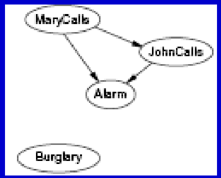
- Suppose we choose the ordering M, J, A, B, E



- $P(J \mid M) = P(J)$? No.
  If Mary calls, that probably means that alarm has gone off, which would make it more likely that John calls; therefore, *JohnCalls* needs *MaryCalls* as a parent.
  $P(A \mid J, M) = P(A \mid J)$? $P(A \mid J, M) = P(A)$?

Example contd.

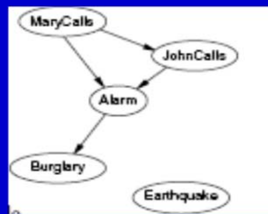
- Suppose we choose the ordering M, J, A, B, E



$P(J \mid M) = P(J)$? No.
$P(A \mid J, M) = P(A \mid J)$? $P(A \mid J, M) = P(A)$? No.
If both call, it's more likely that the alarm has gone off than if just one or neither call, so we need both *MaryCalls* and *JohnCalls* as parents.
$P(B \mid A, J, M) = P(B \mid A)$?
$P(B \mid A, J, M) = P(B)$?

Example contd.

- 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.
If we know the alarm state, then the call from John or Mary might give us information about our phone ringing or Mary's music, but not about burglary.
Thus, we need just Alarm as parent.
P(E | B, A, J, M) = P(E | A) ? P(E | B, A, J, M) = P(E | A, B) ?

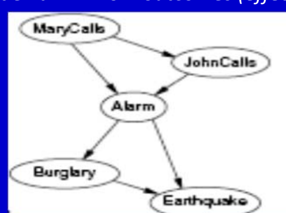
Example contd.

- 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.
If the alarm is on, it's more likely that there has been an earth quake.
But, if we know that there has been a burglary, then that explains the alarm, and the probability of an earthquake would be only slightly above normal. Hence, we need both Alarm and Burglary as parents.

Example contd.

- Diagnostic Model: a link from *outcomes (effects)* to *causes*

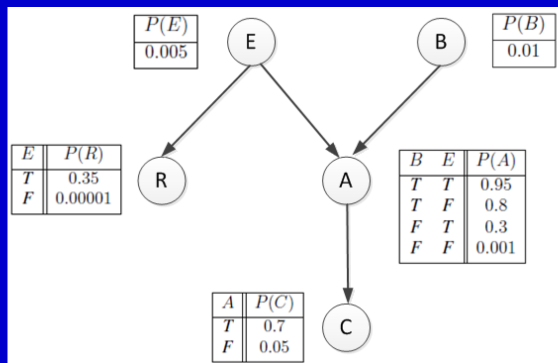


- Deciding conditional independence is hard in noncausal directions
  (Causal models and conditional independence seem hardwired for humans)
- Assessing conditional probabilities is hard in noncausal directions
- Network is less compact: *1+2+4+2+4=13* numbers needed.

Example 2

- I'm at work, my neighbor calls to say my alarm is ringing.
  Sometimes it's set off by minor earthquakes.
  But, the radio didn't report an earthquake.
  Is there a burglar?
- Variables: *Burglar, Earthquake, Alarm, NeighborCall, Radio*
- Network topology reflects "causal" knowledge:
  - A burglar can set the alarm off
  - An earthquake can set the alarm off
  - The alarm can cause your neighbor to call
  - It might be reported on the radio if there is an earthquake.

Example 2: contd.

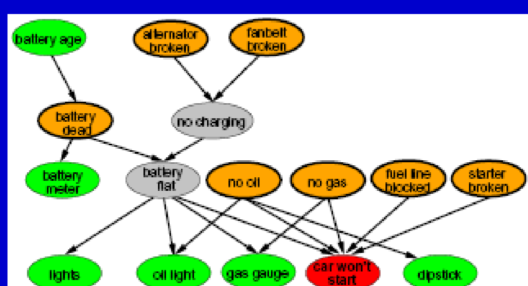


Example 2: cont.

- Suppose we choose the ordering C, A, B, E, R
- Assignment 1:
  Construct Bayes Network with the variables in the above order.
  Justify a (un)conditional (in)dependence of variables
  whenever you add a variable.
  Refer to the previous Example.

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



Compact conditional distributions

- CPT grows exponentially with number of parents.
- CPT becomes infinite with continuous-valued parent or child.
- Solution: canonical distributions that are defined compactly
  for a relationship b/t the parents and the child.
- Deterministic nodes are the simplest case: w/o uncertainty
  $X = f(Parents(X))$ for some function f
  - E.g. Boolean functions
    $NorthAmerican \Leftrightarrow Canadian \vee US \vee Mexican$
  - E.g. numerical relationships among continuous variables
    $$\frac{\delta Level}{\delta t} = inflow + precipitation - outflow - evaporation$$

Compact conditional distributions: contd.

- **Noisy-OR** distributions model multiple non-interacting causes
  - It allows for uncertainty a/b the ability of each parent to cause the child to be true.
  - The causal relationship b/t parent & child may be inhibited (e.g. $\neg fever, cold$)

1. Parents include all causes (can add leak node that covers miscellaneous causes.)
2. Independent failure probability q_i for each cause alone

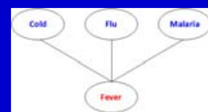
$\Rightarrow P(X \mid U_1, .., U_j, \neg U_{j+1}, ..., \neg U_k) = P(X_i \mid parent(X_i)) = 1 - \prod_{\{j:X j=true\}} q_i$

inhibition probability (or failure probability):

$q_{cold} = P(\neg fever \mid cold, \neg flu, \neg malaria) = 0.6$

$q_{flu} = P(\neg fever \mid \neg cold, flu, \neg malaria) = 0.2$

$q_{malaria} = P(\neg fever \mid \neg cold, \neg flu, malaria) = 0.1$



$Cold$	Flu	$Malaria$	$P(Fever)$	$P(\neg Fever)$
F	F	F	0.0	1.0
F	F	T	0.9	0.1
F	T	F	0.8	0.2
F	T	T	0.98	$0.02 = 0.2 \times 0.1$
T	F	F	0.4	0.6
T	F	T	0.94	$0.06 = 0.6 \times 0.1$
T	T	F	0.88	$0.12 = 0.6 \times 0.2$
T	T	T	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

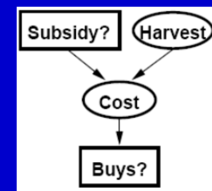
$P(fever \mid Cold, Flu, Malaria)$

$P(\neg fever \mid Cold, Flu, Malaria)$

- Number of parameters is linear in number of parents: $O(k)$, not $O(2^k)$

Hybrid Bayesian Network

- Discrete (*Subsidy?* and *Buys?*); continuous (*Harvest* and *Cost*)



- Option 1: discretization – possible large error, large CPTs
  - E.g.) (< 0° C), (0 ° C – 100 C), (> 100 ° C)
- Option 2: define finitely parameterized canonical families, e.g.) $N(\mu, \sigma^2)$

1. Continuous child variable, discrete + continuous parents (e.g. *Cost*)
2. Discrete child variable, continuous parents (e.g. *Buys?*)

Continuous child variables

- Need one **conditional density** function for child variable (e.g. *Cost*) given continuous parents (e.g. *Harvest*) and each possible assignment to discrete parents (e.g. *Subsidy*)
- How the distribution over the cost depends on the continuous value of *Harvest*? -- Most common is the **linear Gaussian** model, e.g.,:

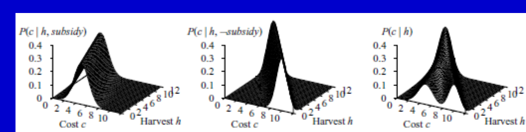
$$P(Cost = c \mid Harvest = h, Subsidy? = true) = N(a_t h + b_t, \sigma_t)(c) = \frac{1}{\sigma_t \sqrt{2\pi}} exp\left(-\frac{1}{2}\left(\frac{c - (a_t h + b_t)}{\sigma_t}\right)^2\right)$$

$$P(Cost = c \mid Harvest = h, Subsidy? = false) = N(a_f h + b_f, \sigma_f)(c) = \frac{1}{\sigma_f \sqrt{2\pi}} exp(-\frac{1}{2}(\frac{c - (a_f h + b_f)}{\sigma_f})^2)$$

- Mean *Cost* varies linearly with *Harvest*, standard variation is fixed
- Linear variation is unreasonable over the full range

  but works OK if the likely range of *Harvest* is narrow

Continuous child variables



(a) (b) (c)

- All-continuous network with Linear Gaussian distributions

  $\Rightarrow$ full joint distribution is a multivariate Gaussian:

$$P(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} e^{-\frac{1}{2}((\mathbf{x}-\mu)^T \Sigma^{-1} (\mathbf{x}-\mu))}$$

where μ is the mean vector

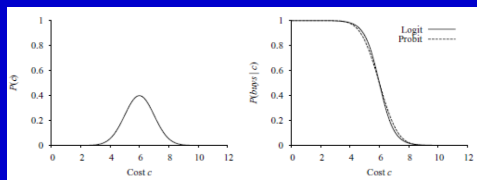
Σ is the covariance matrix.

$\mathrm{cov}(X, Y) = \mathrm{E}((X - \mu_X)(Y - \mu_Y))$

$\Sigma_{ij} = \mathrm{cov}(X_i, X_j) = \mathrm{E}((X_i - \mu_i)(Y_j - \mu_j))$

Discrete variables w/ continuous parents

- Probability of *Buys?* given *Cost* should be a "soft" threshold:



μ=6, σ=1.0

- Probit distribution uses integral of Gaussian:
  $\Phi(x) = \int_{(-\infty,\, x)} N(0,\, 1)(u)\, du = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} exp(-\frac{u^2}{2}) du$
  $P\ (Buys? = true \mid Cost{=}c) = \Phi\ ((\ {-c}+\mu) / \sigma)$
- Sigmoid (or logit) distribution also used: similar shape but much longer tails $P(Buys? = true \mid Cost = c) = \frac{1}{1 + exp(-2\frac{-c+\mu}{\sigma})}$

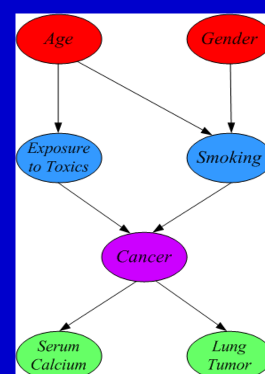
Summary

- Bayes nets provide a natural representation for (causally induced) conditional independence
- Topology + CPTs = compact representation of joint distribution
- Generally easy for (non)experts to construct
- Canonical distributions (e.g., noisy-OR)
  = compact representation of CPTs
- Continuous variables
  $\Rightarrow$ parameterized distributions (e.g. linear Gaussian)

Inference tasks

- Simple queries: compute posterior marginal $P(X_i \mid E{=}e)$
  e.g., ***P****(NoGas | Gauge=empty, Lights=on, Starts=false)*
- Conjunctive queries: $P(X_i, X_j \mid E{=}e) = P(X_i \mid E{=}e) P(X_j \mid X_i, E{=}e)$
- 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?

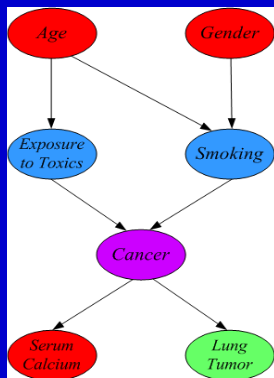
Example: Predictive Inference



How likely are *elderly males* to get *malignant cancers*?

P(C=malignant | Age > 60, Gender=male)

Example: Combined Inference



How likely is an *elderly male* patient with high *Serum Calcium* to have *malignant cancers*?

P(C=malignant | Age > 60, Gender=male, Serum Calcium=high)

Inference in Belief Networks

- Find $P(Q=q \mid E=e)$
  - Q: the query variable
  - E: a set of evidence variables

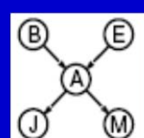
$$P(q|e) = \frac{P(q,e)}{P(e)} = \alpha \cdot P(q,e)$$

- $X_1, \ldots X_n$ are network variables except Q, E, *i.e.* hidden vars.

$$P(q,e) = \sum_{x_1,\ldots,x_n} P(q,e,x_1,\ldots,x_n)$$

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:



$P(B \mid j, m)$
$= P(B, j, m) / P(j, m)$
$= \alpha P(B, j, m)$
$= \alpha \sum_e \sum_a P(B, e, a, j, m)$

- Rewrite full joint entries using product of CPT entries:

$P(B \mid j, m)$
$= \alpha \sum_e \sum_a P(B)\, P(e)\, P(a \mid B, e)\, P(j|a)\, P(m|a)$
$= \alpha P(B) \sum_e P(e) \sum_a P(a \mid B, e) \cdot P(j|a) \cdot P(m|a)$
$\Rightarrow$ *Normalize* $< P(b \mid j, m)\ P(\neg b \mid j, m) >$

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

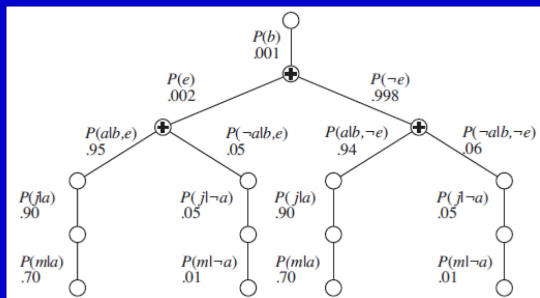
Enumeration algorithm

```
function ENUMERATION-ASK(X, e, bn) returns a distribution over X
  inputs: X, the query variable
          e, observed values for variables E
          bn, a Bayes net with variables {X} ∪ E ∪ Y   /* Y = hidden variables */

  Q(X) ← a distribution over X, initially empty
  for each value x_i of X do
     Q(x_i) ← ENUMERATE-ALL(bn.VARS, e_{x_i})
        where e_{x_i} is e extended with X = x_i
  return NORMALIZE(Q(X))

function ENUMERATE-ALL(vars, e) returns a real number
  if EMPTY?(vars) then return 1.0
  Y ← FIRST(vars)
  if Y has value y in e
     then return P(y | parents(Y)) × ENUMERATE-ALL(REST(vars), e)
     else return Σ_y P(y | parents(Y)) × ENUMERATE-ALL(REST(vars), e_y)
        where e_y is e extended with Y = y
```

Evaluation tree



- Structure of the expression: top-down
- Enumeration is inefficient: repeated computation
  e.g.) computes $P(j|a)P(m|a)$ for each value of e.

Inference by variable elimination

- Variable elimination:
  carry out summations right-to-left (i.e. bottom-up in ET),
  storing intermediate results (factors) to *avoid repeated computation*

$$\begin{aligned}
\mathbf{P}(B|j,m) &= \alpha \underbrace{\mathbf{P}(B)}_{B} \Sigma_e \underbrace{P(e)}_{E} \Sigma_a \underbrace{\mathbf{P}(a|B,e)}_{A} \underbrace{P(j|a)}_{J} \underbrace{P(m|a)}_{M} \\
&= \alpha \mathbf{P}(B) \Sigma_e P(e) \Sigma_a \mathbf{P}(a|B,e) P(j|a) f_M(a) \\
&= \alpha \mathbf{P}(B) \Sigma_e P(e) \Sigma_a \mathbf{P}(a|B,e) f_J(a) f_M(a) \\
&= \alpha \mathbf{P}(B) \Sigma_e P(e) \Sigma_a f_A(a,b,e) f_J(a) f_M(a) \\
&= \alpha \mathbf{P}(B) \Sigma_e P(e) f_{\bar{A}JM}(b,e) \text{ (sum out } A) \\
&= \alpha \mathbf{P}(B) f_{\bar{E}\bar{A}JM}(b) \text{ (sum out } E) \\
&= \alpha f_B(b) \times f_{\bar{E}\bar{A}JM}(b)
\end{aligned}$$

Example: variable elimination

$$\mathbf{P}(B|j,m) = \alpha \underbrace{\mathbf{P}(B)}_{\mathbf{f}_1(B)} \sum_e \underbrace{P(e)}_{\mathbf{f}_2(E)} \sum_a \underbrace{\mathbf{P}(a|B,e)}_{\mathbf{f}_3(A,B,E)} \underbrace{P(j|a)}_{\mathbf{f}_4(A)} \underbrace{P(m|a)}_{\mathbf{f}_5(A)}.$$

$$\mathbf{f}_4(A) = \begin{pmatrix} P(j|a) \\ P(j|\neg a) \end{pmatrix} = \begin{pmatrix} 0.90 \\ 0.05 \end{pmatrix} \quad \mathbf{f}_5(A) = \begin{pmatrix} P(m|a) \\ P(m|\neg a) \end{pmatrix} = \begin{pmatrix} 0.70 \\ 0.01 \end{pmatrix}$$

$$f_3(A,B,E) = P(a|B,e) = (P(a|B,E), P(\neg a|B,E))$$

$$= \left(\begin{pmatrix} P(a|b,e) & P(a|b,\neg e) \\ P(a|\neg b,e) & P(a|\neg b,\neg e) \end{pmatrix}, \begin{pmatrix} P(\neg a|b,e) & P(\neg a|b,\neg e) \\ P(\neg a|\neg b,e) & P(a|\neg b,\neg e) \end{pmatrix} \right)$$

$$= \alpha \mathbf{f}_1(B) \times \sum_e \mathbf{f}_2(E) \times \sum_a \mathbf{f}_3(A,B,E) \times \mathbf{f}_4(A) \times \mathbf{f}_5(A)$$

$$\begin{aligned}
\mathbf{f}_6(B,E) &= \sum_a \mathbf{f}_3(A,B,E) \times \mathbf{f}_4(A) \times \mathbf{f}_5(A) \\
&= (\mathbf{f}_3(a,B,E) \times \mathbf{f}_4(a) \times \mathbf{f}_5(a)) + (\mathbf{f}_3(\neg a,B,E) \times \mathbf{f}_4(\neg a) \times \mathbf{f}_5(\neg a)).
\end{aligned}$$

$$= \alpha \mathbf{f}_1(B) \times \sum_e \mathbf{f}_2(E) \times \mathbf{f}_6(B,E)$$

$$\mathbf{f}_7(B) = \sum_e \mathbf{f}_2(E) \times \mathbf{f}_6(B,E) = \mathbf{f}_2(e) \times \mathbf{f}_6(B,e) + \mathbf{f}_2(\neg e) \times \mathbf{f}_6(B,\neg e)$$

$$= \alpha \mathbf{f}_1(B) \times \mathbf{f}_7(B)$$

Example: continued.

$$\mathbf{f}_4(A) = \begin{pmatrix} P(j|a) \\ P(j|\neg a) \end{pmatrix} = \begin{pmatrix} 0.90 \\ 0.05 \end{pmatrix} \quad \mathbf{f}_5(A) = \begin{pmatrix} P(m|a) \\ P(m|\neg a) \end{pmatrix} = \begin{pmatrix} 0.70 \\ 0.01 \end{pmatrix}$$

$$\begin{aligned}
\mathbf{f}_6(B,E) &= \sum_a \mathbf{f}_3(A,B,E) \times \mathbf{f}_4(A) \times \mathbf{f}_5(A) \\
&= (\mathbf{f}_3(a,B,E) \times \mathbf{f}_4(a) \times \mathbf{f}_5(a)) + (\mathbf{f}_3(\neg a,B,E) \times \mathbf{f}_4(\neg a) \times \mathbf{f}_5(\neg a)). \\
&= \left[\begin{pmatrix} P(a|b,e) & P(a|b,\neg e) \\ P(a|\neg b,e) & P(a|\sim b,\neg e) \end{pmatrix} \cdot P(j|a) \cdot P(m|a) \right. \\
&\quad \left. + \begin{pmatrix} P(\neg a|b,e) & P(\neg a|b,\neg e) \\ P(\neg a|\neg b,e) & P(\neg a|\neg b,\neg e) \end{pmatrix} \cdot P(j|\neg a) \cdot P(m|\neg a) \right] \\
&= \begin{pmatrix} P(j,m|b,e) & P(j,m|b,\neg e) \\ P(j,m|\neg b,e) & P(j,m|\neg b,\neg e) \end{pmatrix}
\end{aligned}$$

$$\mathbf{f}_7(B) = \sum_e \mathbf{f}_2(E) \times \mathbf{f}_6(B,E) = \mathbf{f}_2(e) \times \mathbf{f}_6(B,e) + \mathbf{f}_2(\neg e) \times \mathbf{f}_6(B,\neg e)$$

$$= P(e) \begin{pmatrix} P(j,m|b,e) \\ P(j,m|\neg b,e) \end{pmatrix} + P(\neg e) \begin{pmatrix} P(j,m|b,\neg e) \\ P(j,m|\sim b,\neg e) \end{pmatrix} = \begin{pmatrix} P(j,m|b) \\ P(j,m|\neg b) \end{pmatrix}$$

$$\begin{aligned}
\boldsymbol{P(B|j,m)} &= \alpha f_1(B) \times f_7(B) = \alpha (P(b), P(\neg b)) \cdot \begin{pmatrix} P(j,m|b) \\ P(j,m|\neg b) \end{pmatrix} \\
&= \alpha \cdot (P(j,m,b),\ P(j,m,\neg b)) \\
&= (P(b|j,m),\ P(\neg b|j,m))
\end{aligned}$$

Variable elimination: Basic operations

- Summing out a variable from a product of factors:
  - move any constant factors outside the summation
  - add up submatrices in pointwise product of remaining factors
- $\sum_x f_1 \times \cdots \times f_k = f_1 \times \cdots \times f_i \sum_x f_{i+1} \times \cdots \times f_k = f_1 \times \cdots \times f_i \times f_{\bar{X}}$

  assuming $f_1, \ldots, f_i$ do not depend on X
- Pointwise product of factors f1 and f2:

  $f_1(x_1,\ldots, x_j, y_1,\ldots, y_k) \times f_2(y_1, \ldots, y_k, z_1, \ldots, z_l) = f(x_1, \ldots, x_j, y_1, \ldots, y_k, z_1, \ldots, z_l)$
- E.g.) $f_1(a, b) \times f_2(b,c) = f(a, b, c)$

Cont. : Basic operations

A	B	$\mathbf{f}_1(A,B)$	B	C	$\mathbf{f}_2(B,C)$	A	B	C	$\mathbf{f}_3(A,B,C)$
T	T	.3	T	T	.2	T	T	T	$.3 \times .2 = .06$
T	F	.7	T	F	.8	T	T	F	$.3 \times .8 = .24$
F	T	.9	F	T	.6	T	F	T	$.7 \times .6 = .42$
F	F	.1	F	F	.4	T	F	F	$.7 \times .4 = .28$
						F	T	T	$.9 \times .2 = .18$
						F	T	F	$.9 \times .8 = .72$
						F	F	T	$.1 \times .6 = .06$
						F	F	F	$.1 \times .4 = .04$

Figure 14.10 Illustrating pointwise multiplication: $\mathbf{f}_1(A,B) \times \mathbf{f}_2(B,C) = \mathbf{f}_3(A,B,C)$.

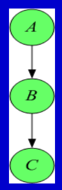
$$\mathbf{f}(B,C) = \sum_a \mathbf{f}_3(A,B,C) = \mathbf{f}_3(a,B,C) + \mathbf{f}_3(\neg a,B,C) = \begin{pmatrix} .06 & .24 \\ .42 & .28 \end{pmatrix} + \begin{pmatrix} .18 & .72 \\ .06 & .04 \end{pmatrix} = \begin{pmatrix} .24 & .96 \\ .48 & .32 \end{pmatrix}.$$

Variable elimination algorithm

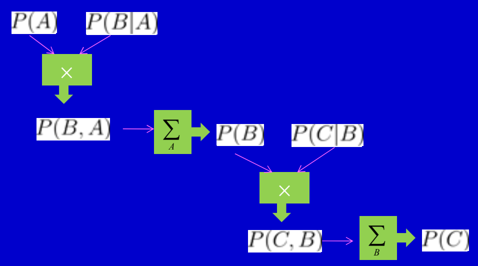
```
function ELIMINATION-ASK(X, e, bn) returns a distribution over X
  inputs: X, the query variable
          e, observed values for variables E
          bn, a Bayesian network specifying joint distribution P(X1, ..., Xn)

  factors ← []
  for each var in ORDER(bn.VARS) do
     factors ← [MAKE-FACTOR(var, e)|factors]
     if var is a hidden variable then factors ← SUM-OUT(var, factors)
  return NORMALIZE(POINTWISE-PRODUCT(factors))
```

Example: Variable Elimination



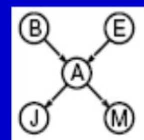
$$P(c) = \sum_b P(c|b) \underbrace{\sum_a P(b|a) P(a)}_{P(b)}$$



Irrelevant variables

- Consider the query P(JohnCalls | Burglary = true)

  $P(J \mid b) = \alpha P(b) \sum_e P(e) \sum_a P(a \mid b,e)\, P(J|a) \sum_m P(m|a)$



- Sum over m is identically 1; M is irrelevant to the query
- Thm 1: Y is irrelevant unless $Y \in Ancestors(\{X\} \cup E)$
- Here, $X = JohnCalls,\ E = \{Burglary\}$, and

  $Ancestors(\{X\}) = \{Alarm, Earthquake\}$

  so, $MaryCalls$ is irrelevant.

  (Compare this to backward chaining from the query in Horn clause KBs).
- Every variable that is not an ancestor of a query variable or evidence variable is irrelevant to the query.

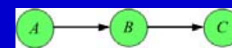
Complexity of exact inference

- *Time* & *Space Complexity* of VE

  are dominated by the *size of the largest factor constructed* and

  is determined by the *order of elimination of variables*

  and by the *structure* of the network.
- Basic Inference



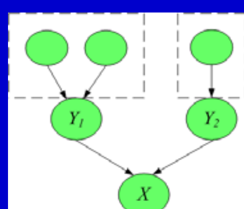
$$P(b) = \sum_a P(a,b) = \sum_a P(b|a)P(a)$$

$$P(c) = \sum_b P(c|b)\, P(b)$$

$$P(c) = \sum_{b,a} P(a,b,c) = \sum_{b,a} P(c|b)P(b|a)P(a) = \sum_b P(c|b) \underbrace{\sum_a P(b|a)P(a)}_{P(b)}$$

Continued..

- Inference in Trees



$$P(x) = \sum_{y_1, y_2} P(x|y_1, y_2)P(y_1, y_2)$$

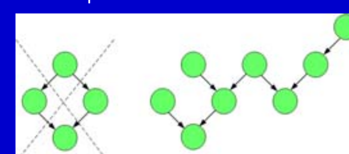
because of independence of Y_1, Y_2:

$$= \sum_{y_1, y_2} P(x|y_1, y_2)P(y_1)P(y_2)$$

Continued..

- Singly connected network (or polytree):
  - Any two nodes are connected by at most one (undirected) path

    i.e. no undirected loop



  - Inference in a singly connected network can be done in *linear in the size of network* where size of network is defined as # of CPT entries.
  - if # of parents of each node is bounded by a constant (k), linear in # of nodes.

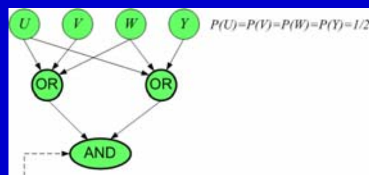
    i.e. Time and space cost of variable elimination are $O(d^k \cdot n)$

    because it needs to maintain distribution over single nodes only.

Continued..

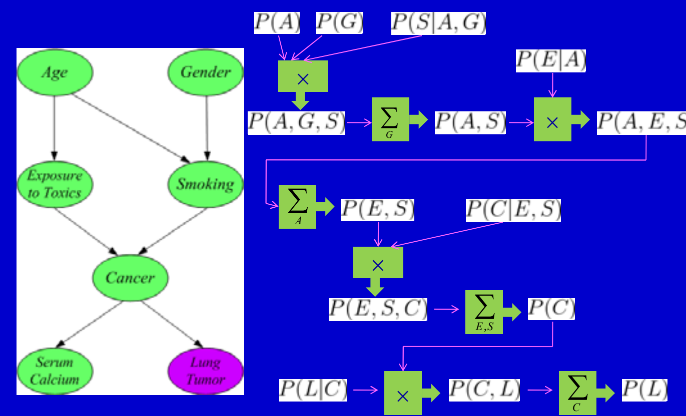
- Multiply connected networks:
  - Exponential time and space complexity of variable elimination in the worst case.
  - Can reduce 3SAT to exact inference $\Rightarrow$ NP-hard

    Inference in a multi-connected Bayesian network is NP-hard.
  - Boolean 3 CNF formula $\phi = (u \vee v \vee w) \wedge (\neg u \vee \neg w \vee y)$



- *Probability( )=$1/2^n$ · # of satisfying assignments of ϕ*
- A close connection b/t the complexity of BN inference and that of CSPs.

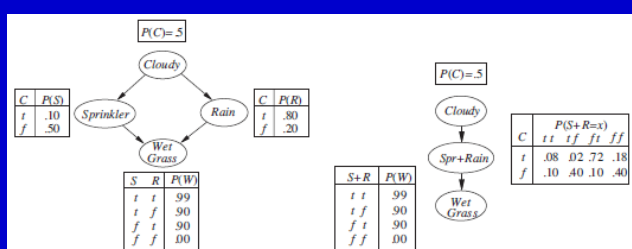
Continued..



Complexity is exponential in the size of the factors.

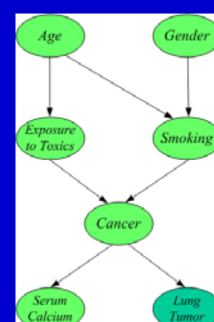
Continued..

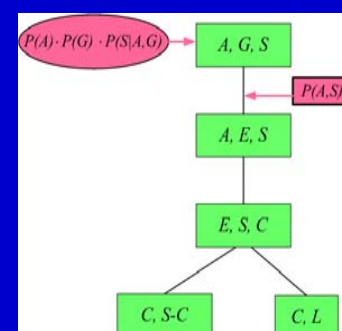
- VE is efficient for a single query, not for queries for all, n, variables, i.e. n posterior probabilities: $O(n^2)$. Using clustering (or joint tree) algorithm, $O(n)$.
- A joint tree is a partially precompiled factorization.



Continued..

- A joint tree is a partially precompiled factorization.





Other Approach to Uncertain Reasoning

- Dempster-Shafer Theory: representing ignorance
- Fuzzy Logic and Set: representing vagueness
- Rule_based methods for uncertain reasoning:
  certainty factor model in MYCIN

Summary

- Exact inference by variable elimination:
  - polytime on polytrees, NP-hard on general graphs
  - space = time, very sensitive to topology
- Approximate inference by LW, MCMC:
  - Will be covered later.
  - LW does poorly when there is lots of (downstream) evidence
  - LW, MCMC generally insensitive to topology
  - Convergence can be very slow with probabilities close to 1 or 0
  - Can handle arbitrary combinations of discrete and continuous variables