(Exact) Inference in Bayesian Networks

Chapter 14

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 required for P(outcome | action; evidence)

- **Value of information:** which evidence to seek next?

- **Sensitivity analysis:** which probability values are most critical?

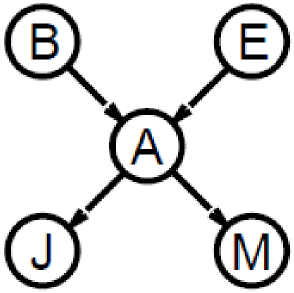
- **Explanation:** why do I need a new starter motor?

Inference by Enumeration

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

Simple query on the burglary network:

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



Rewrite full joint entries using product of CPT entries:

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

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

Enumeration Algorithm

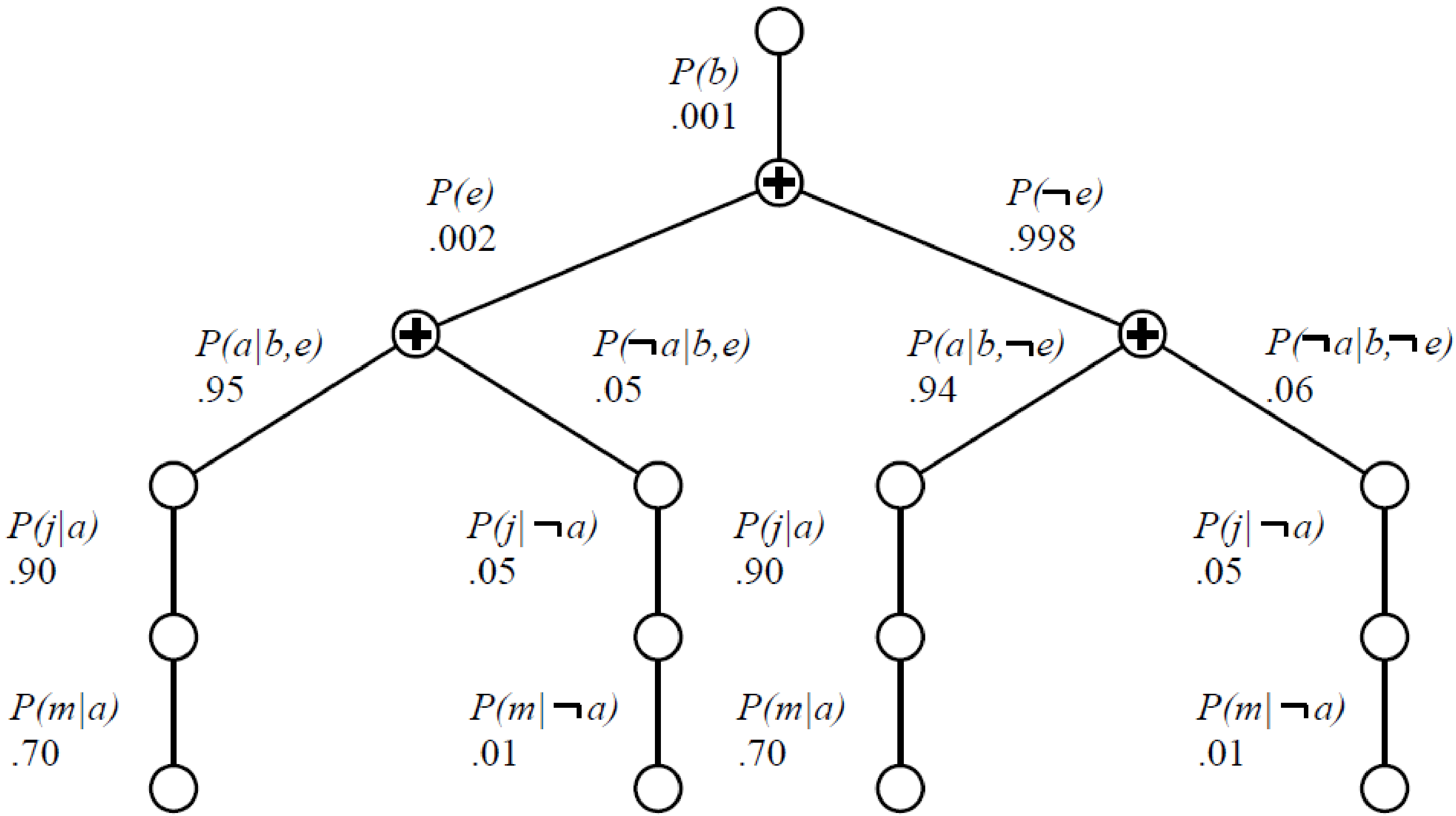
function ENUMERATION-ASK(X, e, bn) **returns** a distribution over X
  inputs: X, the query variable
    e, observed values for variables $\mathbf{E}$
    bn, a Bayesian network with variables $\{X\} \cup \mathbf{E} \cup \mathbf{Y}$

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

---

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

Evaluation Tree



Enumeration is inefficient: repeated computation
e.g., computes $P(j|a)P(m|a)$ for each value of e

Variable Elimination

Variable elimination: carry out summations right-to-left, storing intermediate results (factors) to avoid recomputation

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

Factors

A factor is a representation of a function from a tuple of random variables into a number.

We will write factor f on variables $X_1, \dots, X_j$ as $f(X_1, \dots, X_j)$.

We can assign some or all of the variables of a factor:

- $f(X_1 = v_1, X_2, \dots, X_j)$, where $v_1 \in dom(X_1)$, is a factor on $X_2, \dots, X_j$.
- $f(X_1 = v_1, X_2 = v_2, \dots, X_j = v_j)$ is a number that is the value of f when each X_i has value v_i.

The former is also written as $f(X_1, X_2, \dots, X_j)_{X_1 = v_1}$, etc.

Example of a factor

$r(X, Y, Z)$:

X	Y	Z	val
t	t	t	0.1
t	t	f	0.9
t	f	t	0.2
t	f	f	0.8
f	t	t	0.4
f	t	f	0.6
f	f	t	0.3
f	f	f	0.7

$r(X{=}t, Y, Z)$:

Y	Z	val
t	t	0.1
t	f	0.9
f	t	0.2
f	f	0.8

$r(X{=}t, Y, Z{=}f)$:

Y	val
t	0.9
f	0.8

$$r(X{=}t, Y{=}f, Z{=}f) = 0.8$$

Multiplying factors

The product of factor $f_1(\overline{X}, \overline{Y})$ and $f_2(\overline{Y}, \overline{Z})$, where $\overline{Y}$ are the variables in common, is the factor $(f_1 \times f_2)(\overline{X}, \overline{Y}, \overline{Z})$ defined by:

$$(f_1 \times f_2)(\overline{X}, \overline{Y}, \overline{Z}) \quad = \quad f_1(\overline{X}, \overline{Y}) f_2(\overline{Y}, \overline{Z}).$$

Multiplying factors example

f_1:

A	B	val
t	t	0.1
t	f	0.9
f	t	0.2
f	f	0.8

f_2:

B	C	val
t	t	0.3
t	f	0.7
f	t	0.6
f	f	0.4

$f_1 \times f_2$:

A	B	C	val
t	t	t	0.03
t	t	f	0.07
t	f	t	0.54
t	f	f	0.36
f	t	t	0.06
f	t	f	0.14
f	f	t	0.48
f	f	f	0.32

Sum out a variable

We can sum out a variable, say X_1 with domain $\{v_1, \ldots, v_k\}$, from factor $f(X_1, \ldots, X_j)$, resulting in a factor on $X_2, \ldots, X_j$ defined by:

$$
\begin{aligned}
& (\sum_{X_1} f)(X_2, \ldots, X_j) \\
= \; & f(X_1 = v_1, \ldots, X_j) + \cdots + f(X_1 = v_k, \ldots, X_j)
\end{aligned}
$$

Sum out a variable - Example

f_3:

A	B	C	val
t	t	t	0.03
t	t	f	0.07
t	f	t	0.54
t	f	f	0.36
f	t	t	0.06
f	t	f	0.14
f	f	t	0.48
f	f	f	0.32

$\sum_B f_3$:

A	C	val
t	t	0.57
t	f	0.43
f	t	0.54
f	f	0.46

Evidence

If we want to compute the posterior probability of Z given evidence $Y_1 = v_1 \wedge \ldots \wedge Y_j = v_j$:

$$
\begin{aligned}
& P(Z|Y_1 = v_1, \ldots, Y_j = v_j) \\
& \quad = \frac{P(Z, Y_1 = v_1, \ldots, Y_j = v_j)}{P(Y_1 = v_1, \ldots, Y_j = v_j)} \\
& \quad = \frac{P(Z, Y_1 = v_1, \ldots, Y_j = v_j)}{\sum_Z P(Z, Y_1 = v_1, \ldots, Y_j = v_j)}.
\end{aligned}
$$

So the computation reduces to the probability of $P(Z, Y_1 = v_1, \ldots, Y_j = v_j)$.

We normalize at the end.

Probability of a conjunction

Suppose the variables of the belief network are $X_1, \ldots, X_n$.
To compute $P(Z, Y_1 = v_1, \ldots, Y_j = v_j)$, we sum out the other variables, $Z_1, \ldots, Z_k = \{X_1, \ldots, X_n\} - \{Z\} - \{Y_1, \ldots, Y_j\}$.
We order the Z_i into an elimination ordering.

$$
\begin{aligned}
& P(Z, Y_1 = v_1, \ldots, Y_j = v_j) \\
& \quad = \sum_{Z_k} \cdots \sum_{Z_1} P(X_1, \ldots, X_n)_{Y_1 = v_1, \ldots, Y_j = v_j}. \\
& \quad = \sum_{Z_k} \cdots \sum_{Z_1} \prod_{i=1}^{n} P(X_i | parents(X_i))_{Y_1 = v_1, \ldots, Y_j = v_j}.
\end{aligned}
$$

Computing Sums of products

Computation in belief networks reduces to computing the sums of products.

How can we compute $ab + ac$ efficiently?

Distribute out the a giving $a(b + c)$

How can we compute $\sum_{Z_1} \prod_{i=1}^{n} P(X_i | parents(X_i))$ efficiently?

Distribute out those factors that don't involve Z_1.

Variable Elimination Algorithm

To compute $P(Z|Y_1 = v_1 \wedge \ldots \wedge Y_j = v_j)$:

Construct a factor for each conditional probability.

Set the observed variables to their observed values.

Sum out each of the other variables (the $\{Z_1, \ldots, Z_k\}$) according to some elimination ordering.

Multiply the remaining factors. Normalize by dividing the resulting factor $f(Z)$ by $\sum_Z f(Z)$.

Sum out Evidence

To sum out a variable Z_j from a product $f_1, \dots, f_k$ of factors:

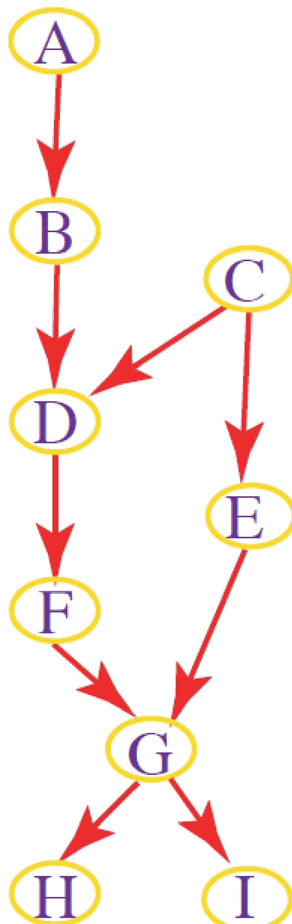
- Partition the factors into
  - those that don't contain Z_j, say $f_1, \dots, f_i$,
  - those that contain Z_j, say $f_{i+1}, \dots, f_k$

We know:

$$\sum_{Z_j} f_1 \times \cdots \times f_k = f_1 \times \cdots \times f_i \times \left(\sum_{Z_j} f_{i+1} \times \cdots \times f_k \right).$$

- Explicitly construct a representation of the rightmost factor. Replace the factors $f_{i+1}, \dots, f_k$ by the new factor.

Variable Elimination – Another Example



$$\left.\begin{array}{l} P(A)\\ P(B|A)\end{array}\right\} \xrightarrow{\text{elim } A} f_1(B)$$

$$\left.\begin{array}{l} P(C)\\ P(D|BC)\\ P(E|C)\end{array}\right\} \xrightarrow{\text{elim } C} f_2(BDE)$$

$$P(F|D)$$

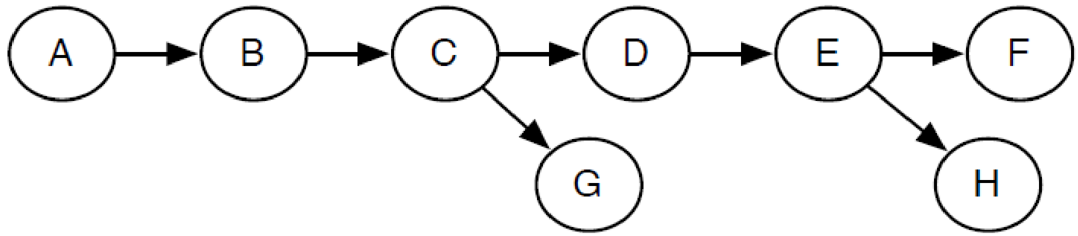
$$P(G|FE)$$

$$P(H|G)\ \} \xrightarrow{\text{obs } H} f_3(G)$$

$$P(I|G)\ \} \xrightarrow{\text{elim } I} f_4(G)$$

$$P(D,h) = \ldots\left(\sum_A P(A)P(B|A)\right)\left(\sum_I P(I|G)\right)$$

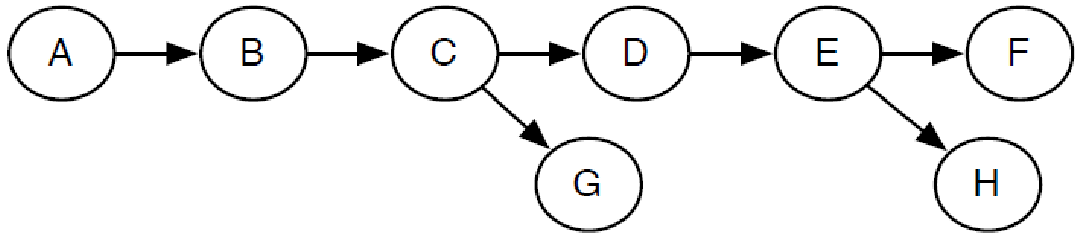
Variable Elimination – (Yet) Another Example



Query: $P(G|f)$; elimination ordering: A, H, E, D, B, C

$$P(G|f) \propto \sum_C \sum_B \sum_D \sum_E \sum_H \sum_A P(A)P(B|A)P(C|B)(D|C)$$
$$P(E|D)P(f|E)P(G|C)P(H|E)$$

Variable Elimination – (Yet) Another Example



Query: $P(G|f)$; elimination ordering: A, H, E, D, B, C

$$P(G|f) \propto \sum_C \sum_B \sum_D \sum_E \sum_H \sum_A P(A)P(B|A)P(C|B)(D|C) P(E|D)P(f|E)P(G|C)P(H|E)$$

$$= \sum_C \left(\sum_B \left(\sum_A P(A)P(B|A) \right) P(C|B) \right) P(G|C) \left(\sum_D P(D|C) \left(\sum_E P(E|D)P(f|E) \sum_H P(H|E) \right) \right)$$

Complexity of Exact Inference

Singly connected networks (or polytrees):

– any two nodes are connected by at most one (undirected) path

– time and space cost of variable elimination are $O(d^k n)$

Multiply connected networks:

– can reduce 3SAT to exact inference $\Rightarrow$ NP-hard

– equivalent to **counting** 3SAT models $\Rightarrow$ #P-complete

1. A v B v C
2. C v D v ¬A
3. B v C v ¬D