Introduction to Artificial Intelligence

COMP307

Inference in a Bayesian Network

Yi Mei

yi.mei@ecs.vuw.ac.nz

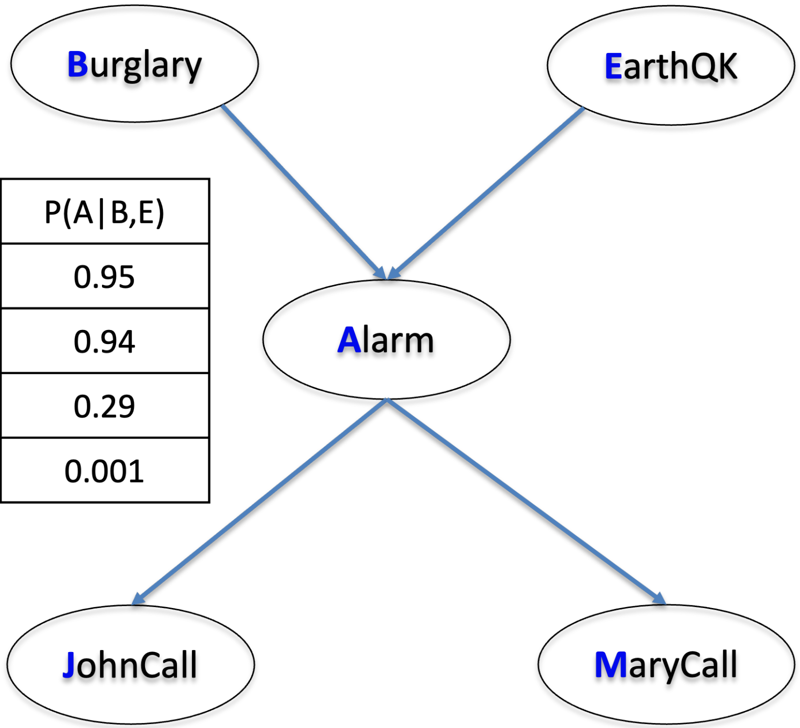
Outline

- Inference in Bayesian networks
- Exact Inference by Enumeration
- Variable Elimination Algorithm
- Examples

Inference in a BN

- If there was an earthquake, how likely Mary will call you?
- If both John and Mary called you, how likely there was a burglary?
- If Mary called you, how likely John will call you as well?

- Answering questions like above is **inference in a BN**



B	P(B)
T	0.001

E	P(E)
T	0.002

B	E	A	P(A|B,E)
T	T	T	0.95
T	F	T	0.94
F	T	T	0.29
F	F	T	0.001

A	J	P(J|A)
T	T	0.9
F	T	0.05

A	M	P(M|A)
T	T	0.7
F	T	0.01

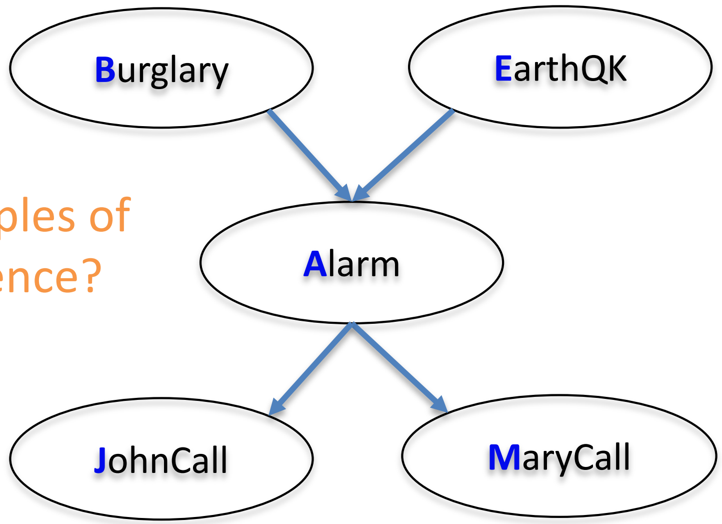
Inference in a BN

- Inference in a BN is to compute the posterior probability distribution for a set of **query nodes**, given values for some **evidence nodes**.
  - What is P(Burglary=true), if we know that (Alarm=true)? P(B = T I A = T)

- This task is called belief updating or probabilistic inference.

- Inference in Bayesian networks is very flexible, as evidence can be entered for any node while beliefs in any other nodes are updated.
  - Causal Reasoning: P(Effect I Cause)
  - Diagnostic Reasoning: P(Cause I Effect)
  - Inter-causal Reasoning: the query nodes are common causes of the evidence nodes.

Can you list some examples of the above type of inference?



Inference by Enumeration

- Problem (**capital letter = variable, lowercase = value**):
  - Given evidence nodes: $e_1, e_2, \dots, e_n$
    - **NB:** b means Burglary is true, $\neg j$ means John didn't call
  - Want to know a query node: Q
  - Other hidden nodes in the Bayesian network: $H_1, H_2, \dots, H_m$
  - $P(Q|e_1, \dots, e_n)$?

- Use the 3 probability rules
  - Product rule
  - Sum rule
  - Normalisation rule

Inference by Enumeration

$$P(Q|e_1,\dots,e_n) = \frac{P(Q,e_1,\dots,e_n)}{P(e_1,\dots,e_n)} = \alpha * P(Q,e_1,\dots,e_n)$$

$$\alpha = \frac{1}{P(e_1,\dots,e_n)}$$

[product rule]

- For the enumerator: include the hidden nodes **[sum rule]**

$$P(Q,e_1,\dots,e_n) = \sum_{H_1,\dots,H_m} P(Q,e_1,\dots,e_n,H_1,\dots,H_m)$$

- Use factorisation of the network, for each term

$$P(Q,e_1,\dots,e_n,H_1,\dots,H_m) =$$

$$P(Q|pa(Q)) * P(e_1|pa(E_1)) * \cdots * P(H_m|pa(H_m))$$

- How many calculations? Assume all binary variables
  - $\mathbf{2^{m+1}}$ joint probabilities, each with $\boldsymbol{m+n}$ multiplications

Variable Elimination Algorithm

- Directly calculating all the joint probabilities can be time consuming
  - $(n + m)2^{m+1}$ multiplications
  - Exponential to the number of hidden variables
  - Can be very slow in large Bayesian networks

- Variable Elimination
  - Many duplicate multiplications between conditional probabilities
    - e.g., for calculating $P(b|j,m) = \alpha * P(b,j,m)$

$$P(b,j,m) = P(b,j,m,e,a) + P(b,j,m,e,\neg a)$$

$$+P(b,j,m,\neg e,a) + P(b,j,m,\neg e,\neg a)$$

$$= P(b)P(e)P(a|b,e)P(j|a)p(m|a) + P(b)P(e)P(\neg a|b,e)P(j|\neg a)p(m|\neg a)$$

$$+P(b)P(\neg e)P(a|b,\neg e)P(j|a)P(m|a) + P(b)P(\neg e)P(\neg a|b,\neg e)P(j|\neg a)P(m|\neg a)$$

  - Calculate once and save for later use

Factors

- A **factor** of some random variables is a **table** of all the possible values of the random variables. Note that the **table value can be any function** involving the random variables.

- Example factors

B	P(B)
t	0.001
f	0.999

A	P(J = t | A)
t	0.9
f	0.05

M	A	P(M | A)
t	t	0.7
f	t	0.3
t	f	0.01
f	f	0.99

Join Factors

- The **join** operation between two factors $f1$ and $f2$, denoted as $f1 \otimes f2$, is a table of the *union* of the variables in $f1$ and $f2$, where each row is the multiplication of the corresponding row of $f1$ and $f2$.

A	P(J = t | A)
t	0.9
f	0.05

$\otimes$

M	A	P(M | A)
t	t	0.7
f	t	0.3
t	f	0.01
f	f	0.99

=

M	A	P(J = t | A) * P(M | A)
t	t	0.7 * 0.9 = 0.63
f	t	0.3 * 0.9 = 0.27
t	f	0.01 * 0.05 = 0.0005
f	f	0.99 * 0.05 = 0.0495

Eliminate

- The **elimination/sum-out** operation of a factor f on a variable $X \in f$, denoted as $\sum_X f$, is a table of all the variables except X, where each row is the sum of the all the rows in f with the the same value of the other variables.

$\sum_M$

M	A	P(J = t | A) * P(M | A)
t	**t**	0.63
f	**t**	0.27
t	**f**	0.0005
f	**f**	0.0495

=

A	$\sum_M$ P(J = t | A) * P(M | A)
t	0.63 + 0.27 = 0.9
f	0.0005 + 0.0495 = 0.05

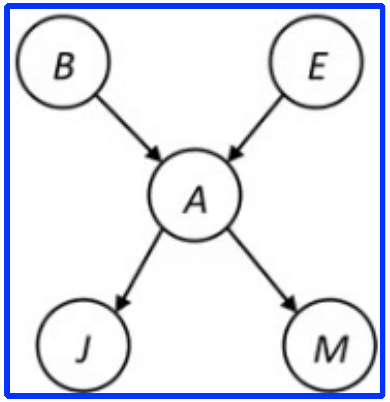
Variable Elimination Algorithm

- Input:
  - Query node Q,
  - Evidence nodes $e_1, \ldots,$
  - Factorisation $P(X_1, \ldots X_n) = P(X_1|parents(X_1)) * \cdots * P(X_n|parents(X_n))$
- Decide the order $X_1', \ldots, X_n'$
- Initialise the factors from the CPTs
- **For each** $i = 1 \rightarrow n$:
- join all the factors with X_i'
- If X_i' is a hidden node, **then** eliminate/sum-out X_i'

- Ordering can affect efficiency
- The computational and space complexity of variable elimination is determined by the largest factor, not the number of factors

Variable Elimination Algorithm

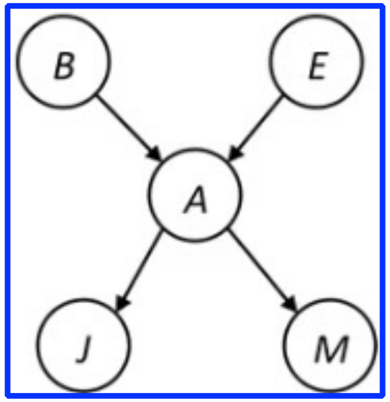
- The alarm network example
- How likely there was a burglary, if John called?
  - Evidence nodes: j
  - Query node: B
  - Hidden nodes: A, E, M



- $P(B|j) = \alpha * P(B, j)$
- $P(B|j) = \alpha * \sum_{A,E} P(B, A, E, M, j)$
- $P(B|j) = \alpha * \sum_{A,E,M} P(B)P(E)P(A|B,E)P(j|A)P(M|A)$

- **No variable elimination**, how many probability multiplications?

Variable Elimination Algorithm

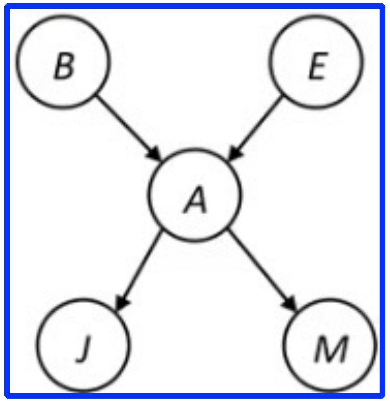


- $\sum_{A,E,M} P(B)P(E)P(A|B,E)P(J=t|A)P(M|A)$
- $B = b$ or $\neg b$
- $E = e$ or $\neg e$
- $A = a$ or $\neg a$
- $M = m$ or $\neg m$
- 2*2*2*2=16 different situations,
  - (b, e, a, m): $P(b)P(e)P(a|b,e)P(j|a)P(m|a)$
  - $(b, e, \neg a, m)$: $P(b)P(e)P(\neg a|b,e)P(j|\neg a)P(m|\neg a)$
  - …
- 4 multiplications per situation

- In total: 16*4=64 multiplications

Variable Elimination Algorithm

- If using VE algorithm
  - Evidence nodes: j
  - Query node: B
  - Hidden nodes: A, E, M
- Order of hidden variables: M -> A -> E



- Initial factors
  - $f_1(B) = P(B)$: 2-row table
  - $f_2(E) = P(E)$: 2-row table
  - $f_3(A, B, E) = P(A|B, E)$: 8-row table
  - $f_4(A) = P(J = t|A)$: 2-row table
  - $f_5(M, A) = P(M|A)$: 4-row table

- $\sum_{A,E,M} f_1(B) \otimes f_2(E) \otimes f_3(A, B, E) \otimes f_4(A) \otimes f_5(M, A)$

Variable Elimination Algorithm

- Calculate $\sum_{A,E,M} f_1(B) \otimes f_2(E) \otimes f_3(A,B,E) \otimes f_4(A) \otimes f_5(M,A)$

- Step 1: Join all factors containing **M**, and sum-out **M**
  - $f_6(A) = \sum_M f_5(M,A)$: 2-row table
  - Calculate $\sum_{A,E} f_1(B) \otimes f_2(E) \otimes f_3(A,B,E) \otimes f_4(A) \otimes f_6(A)$

- Step 2: join all factors containing **A**, and sum-out **A**
  - $f_7(B,E) = \sum_A f_3(A,B,E) \otimes f_4(A) \otimes f_6(A)$: 4-row table
  - Calculate $\sum_E f_1(B) \otimes f_2(E) \otimes f_7(B,E)$

- Step 3: join all factors containing **E**, and sum-out **E**
  - $f_8(B) = \sum_E f_2(E) \otimes f_7(B,E)$: 2-row table
  - Calculate $f_1(B) \otimes f_8(B)$

- How many probability multiplications?
- **Read the tutorial**
  https://github.com/meiyi1986/tutorials/blob/master/notebooks/bayesian-network-variable-elimination.ipynb